DB2 11 for z/OS

Messages

IBM

DB2 11 for z/OS

Messages

IBM

Contents

About this information

This information helps you to understand the messages that you could receive while working with DB2® 11 for z/OS® (DB2 for z/OS). The topics provide detailed descriptions of the messages that are issued by DB2, with explanations and user responses for the situations that are associated with these messages.

Unless the message is issued by a subcomponent operating outside the DB2 environment, messages are preceded by a *command prefix* that is defined during system installation. The command prefix is not shown in the message text.

This information assumes that your DB2 subsystem is running in Version 11 new-function mode. Generally, new functions that are described, including changes to existing functions, statements, and limits, are available only in new-function mode, unless explicitly stated otherwise. Exceptions to this general statement include optimization and virtual storage enhancements, which are also available in conversion mode unless stated otherwise.

Throughout this information, "DB2" means "DB2 11 for z/OS". Other DB2 products are given their complete names or abbreviations.

Who should read this information

DB2 Messages lists messages issued by DB2, with explanations and suggested responses. This information is intended for users and system programmers who might encounter DB2 messages.

DB2 Utilities Suite

Important: In this version of DB2 for z/OS, the DB2 Utilities Suite is available as an optional product. You must separately order and purchase a license to such utilities, and discussion of those utility functions in this publication is not intended to otherwise imply that you have a license to them.

The DB2 Utilities Suite can work with DB2 Sort and the DFSORT program. You are licensed to use DFSORT in support of the DB2 utilities even if you do not otherwise license DFSORT for general use. If your primary sort product is not DFSORT, consider the following informational APARs mandatory reading:

- II14047/II14213: USE OF DFSORT BY DB2 UTILITIES
- II13495: HOW DFSORT TAKES ADVANTAGE OF 64-BIT REAL ARCHITECTURE

These informational APARs are periodically updated.

> **Related information**
> DB2 utilities packaging (Utility Guide)

Terminology and citations

When referring to a DB2 product other than DB2 for z/OS, this information uses the product's full name to avoid ambiguity.

The following terms are used as indicated:

DB2 Represents either the DB2 licensed program or a particular DB2 subsystem.

OMEGAMON®
Refers to any of the following products:
- IBM® Tivoli® OMEGAMON XE for DB2 Performance Expert on z/OS
- IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS
- IBM DB2 Performance Expert for Multiplatforms and Workgroups
- IBM DB2 Buffer Pool Analyzer for z/OS

C, C++, and C language
Represent the C or C++ programming language.

CICS® Represents CICS Transaction Server for z/OS.

IMS™ Represents the IMS Database Manager or IMS Transaction Manager.

MVS™ Represents the MVS element of the z/OS operating system, which is equivalent to the Base Control Program (BCP) component of the z/OS operating system.

RACF®
Represents the functions that are provided by the RACF component of the z/OS Security Server.

Accessibility features for DB2 11 for z/OS

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in z/OS products, including DB2 11 for z/OS. These features support:
- Keyboard-only operation.
- Interfaces that are commonly used by screen readers and screen magnifiers.
- Customization of display attributes such as color, contrast, and font size

Tip: The Information Management Software for z/OS Solutions Information Center (which includes information for DB2 11 for z/OS) and its related publications are accessibility-enabled for the IBM Home Page Reader. You can operate all features using the keyboard instead of the mouse.

Keyboard navigation

You can access DB2 11 for z/OS ISPF panel functions by using a keyboard or keyboard shortcut keys.

For information about navigating the DB2 11 for z/OS ISPF panels using TSO/E or ISPF, refer to the *z/OS TSO/E Primer*, the *z/OS TSO/E User's Guide*, and the *z/OS ISPF User's Guide*. These guides describe how to navigate each interface, including the use of keyboard shortcuts or function keys (PF keys). Each guide includes the default settings for the PF keys and explains how to modify their functions.

Related accessibility information

Online documentation for DB2 11 for z/OS is available in the Information Management Software for z/OS Solutions Information Center, which is available at the following website: http://pic.dhe.ibm.com/infocenter/dzichelp/v2r2/index.jsp

IBM and accessibility

See the *IBM Accessibility Center* at http://www.ibm.com/able for more information about the commitment that IBM has to accessibility.

How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this book or other DB2 for z/OS documentation. You can use the following methods to provide comments:

- Send your comments by email to db2zinfo@us.ibm.com and include the name of the product, the version number of the product, and the number of the book. If you are commenting on specific text, please list the location of the text (for example, a chapter and section title or a help topic title).

- You can also send comments by using the **Feedback** link at the footer of each page in the Information Management Software for z/OS Solutions Information Center at http://pic.dhe.ibm.com/infocenter/dzichelp/v2r2/index.jsp.

How to interpret message numbers

DB2 issues messages to communicate system status and processing results. Messages are identifiable by unique message numbers. The format of a message number can help you understand what the the message applies to and, in some cases, its severity.

Messages are identifiable by unique message numbers, which are eight to ten characters long. Message identifiers have the following format:

DSNB209I

DSN The first three characters are DSN, which is the DB2 component prefix.

B The fourth character is the *subcomponent identifier*, which identifies the subcomponent of DB2 that issued the message. Each subcomponent has an associated *hexadecimal identifier* (hex ID), which is the hexadecimal representation of the subcomponent identifier. In this example, B identifies the DB2 buffer manager.

209 The *numeric identifier* is three to five characters long and begins at the fifth character in the message. The numeric identifier is unique within the subcomponent.

I The last character of the message identifier is the *message type code*. For DB2 messages in general, the I type code is used to indicate that this message is for information only. However, some older messages use other type codes. The type code does not necessarily reflect the severity of these messages.

Table 1. DB2 message type codes

Message type code	Meaning
A	Action is required immediately. The associated task does not continue until the requested action is taken.

Table 1. DB2 message type codes (continued)

Message type code	Meaning
D	Decision or action is required immediately. The associated task does not continue until the requested decision is made or action is taken.
E	Action is required at some point; however, the associated task continues. All RC=8 error messages that are not host related (SOFT errors) have the E changed to a W in precompiler output when SQL(ALL) is in effect. All RC=8 errors (HARD errors) look the same as they have in the past.
I	No action is required. The message is informational only.

Messages in some subcomponents are associated with particular return codes. The return codes can vary by subcomponent. The introduction to each set of messages describes the return codes used for that subcomponent.

Messages are organized by subcomponent identifier and numeric identifier, and sorted numerically within each group.

Subcomponent identifiers

The numbers that are assigned to messages and reason codes in DB2 are associated with certain DB2 subcomponents or functional areas.

The following table lists the DB2 subcomponents and functional areas and shows the identifiers associated with each one.

Table 2. Subcomponent identifiers used in messages and reason codes

Subcomponent or functional area	Associated identifier for messages	Associated identifier for reason codes
Call attachment facility	A	X'C1'
Buffer manager	B	X'C2'
TSO attachment facility	E	X'C5'
Message generator	F	X'C6'
Database descriptor manager	G	n/a
Precompiler, DSNH CLIST	H	X'C8'
Data manager	I	X'C9'
Recovery log manager	J	X'D1'
Distributed data facility	L	X'D3'
IMS attachment facility	M	X'D4'
XML manager	n/a	X'D5'
Data space manager	P	X'D7'
MQListener	Q	n/a
Recovery manager	R	X'D9'
Storage manager	S	X'E2'

Table 2. Subcomponent identifiers used in messages and reason codes (continued)

Subcomponent or functional area	Associated identifier for messages	Associated identifier for reason codes
Service controller, install	T	X'E3'
Utilities	U	X'E4'
Agent services manager	V	X'E5'
Instrumentation facility	W	X'E6'
Relational data system	X	X'E7'
Initialization procedures	Y	X'E8'
System parameter manager	Z	X'E9'
Service facilities	1	X'F1'
Subsystem support subcomponent	3	X'F3'
Group manager	7	X'F7'
Sample applications	8	X'F8'
General command processor	9	X'F9'

Messages with `DSNC` and `DSN2` identifiers and codes in the X'C3' range were previously used to display CICS messages. For information about CICS messages, see the information in the CICS Library.

How to find additional information about messages

When you are working with DB2, you might encounter messages issued by other IBM products, such as IMS, CICS, or z/OS. Information about messages issued by these and other IBM products is available in a variety of formats.

CICS Information centers for multiple versions of CICS are available as part of the CICS Library.

IMS IMS messages are available in IMS messages and codes.

IRLM Messages and codes related to the IRLM, which were previously part of *DB2 Codes*, are now provided in *IRLM Messages and Codes for IMS and DB2 for z/OS*. IRLM messages have the prefix dxr.
- IRLM return and reason codes (IRLM)
- IRLM abend codes (IRLM)
- DXR messages (IRLM)

z/OS Information for multiple versions of z/OS is available as part of the z/OS Library.

Many IBM message references are also available from the IBM Publications Center and some IBM product messages are enabled for the LookAt facility.

Chapter 1. DSNA messages

Message identifiers that begin with DSNA communicate information about the call attachment facility (CAF).

DSNA messages fall into two categories:

- Messages that are written to a trace file. These are seen only if the trace is turned on (that is, ddname DSNTRACE was allocated to SYSOUT or the TSO terminal).
- Messages that are returned by a DB2 stored procedure. These are seen only as a parameter that is returned by the stored procedure.

DSNA messages use common elements and share the same basic structure:

- Message number (DSNA*nnna*).
- The name of the CSECT responsible for writing the message.
- TCB address of the task currently executing.
- A series of tokens and token-explanations unique to each message.

The intent of these messages is to aid IBM development and the IBM Support Center. They contain information that might not be meaningful except to IBM.

Messages in the DSNA8 range are intended for diagnosis and are documented in *Diagnosis Guide and Reference*.

Related information:

➡ DB2 Diagnosis Guide and Reference

DSNA002E   DSNACA00 TCB=address SYSTEM ERROR - FOUND UNEXPECTED THREAD

Explanation: The DB2 call attachment facility control block, the CAB, indicated that a thread was present when none should have been present. This is a CAF system error.

System action: This message immediately precedes an 04E abend, with reason code 00C10002.

System programmer response: Collect a SYSUDUMP and DSNTRACE trace stream.

Programmer response: Refer the problem to your system programmer.

Problem determination: Contact your IBM Support Center.

DSNA201E   DSNACA00 TCB=address CANNOT USE 'CONNECT', ALREADY CONNECTED TO DB2

Explanation: A CONNECT request was received but a connection to DB2 already existed for the TCB identified by 'address' in the message. This is a CAF error (application program error). The message is accompanied by return code (R15) 200 and code (R0) 00C10201.

System action: No new connection is made to DB2 and an error return and reason code are issued. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: You must DISCONNECT the previous connection before you can re-CONNECT. Correct the logic in your code.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA202E   DSNACA00 TCB=address CANNOT USE 'OPEN', PRIOR OPEN OR SQL ALREADY CONNECTED APPLICATION TO DB2

Explanation: An OPEN request was received but a DB2 thread had already been created for the specified TCB. This is a user error. The message is accompanied by return code (R15) 200 and code (R0) 00C10202.

System action: No new DB2 thread is created and an error return and reason code are issued. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Your application program had already (explicitly or implicitly) established an OPEN level connection to DB2. You cannot do this again until the original OPEN is terminated. 'Explicit' OPENs occur when an application issues the call attachment facility OPEN request. 'Implicit' OPENs occur when an application issues SQL statements before a DB2 connection was established. Correct the logic in your application program.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA203E   DSNACA00 TCB=address CANNOT USE 'CLOSE', THERE WAS NO 'OPEN' IN EFFECT

Explanation: A CLOSE request was received but no OPEN level connection (implicit or explicit) existed for this task. This is a user error. The message is accompanied by return code (R15) 200 and code (R0) 00C10203.

System action: No CLOSE is performed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Correct the logic in your application. Do not issue CLOSE unless an explicit OPEN has been performed. 'Explicit' OPENs occur when an application issues the call attachment facility OPEN request.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA204E   DSNACA00 TCB=address CANNOT USE 'DISCONNECT', THERE WAS NO 'CONNECT' IN EFFECT

Explanation: A DISCONNECT request was received but no DB2 connection had been established. This is a user error. The message is accompanied by return code (R15) 200 and code (R0) 00C10204.

System action: No DISCONNECT has been performed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Correct the logic in your application. Do not issue DISCONNECT unless a CONNECT has been performed.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA205E   DSNACA00 TCB=address CANNOT USE 'TRANSLATE', THERE WAS NO CONNECTION TO DB2 OR NOTHING TO TRANSLATE

Explanation: A TRANSLATE request was received but no data was available for translation. You must have valid DB2 call attachment facility control structures before you can make a TRANSLATE request. Control structures are built when you issue explicit or implicit connection requests. This is a user error. The message is accompanied by return code (R15) 200 and code (R0) 00C10205.

System action: No TRANSLATE is performed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Correct the logic in your application. Do not issue TRANSLATE unless a DB2 connection has been attempted.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA206E   **DSNACA00 TCB=address WRONG NUMBER OF PARMS OR VL BIT OFF**

Explanation: An erroneous DB2 call attachment facility request was received. The request had too many or too few parameters and could not be processed. The message is accompanied by return code (R15) 200 and code (R0) 00C10206.

System action: The previous request was not processed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Correct the logic in your application. Be sure that any parameter list passed to the DB2 call attachment facility has the correct number of parameters for the individual call, and that the high order (VL) bit is turned on to end the parameter list.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA207E   **DSNACA00 TCB=**_tcb-address_ **DID NOT RECOGNIZE ACTION PARAMETER -** _parameter-name_

Explanation: An unrecognized action request type or parameter was received by the DB2 call attachment facility and could not be processed. The allowable action request types are CONNECT, OPEN, CLOSE, DISCONNECT, and TRANSLATE. This message is accompanied by return code (R15) 200 and code (R0) 00C10207.

System action: Control returns to the caller. Your request was not processed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Correct the logic in your application. Check the spelling of the action parameter type or keyword on the previous request passed to the DB2 call attachment facility. The action parameter type you passed is listed in the parm field of the DSNA800I trace message. If the error is not the action parameter, the parameter in error is listed in the parm field of this message.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

Related tasks:

➡ Invoking the call attachment facility (DB2 Application programming and SQL)

Related reference:

➡ CAF return codes and reason codes (DB2 Application programming and SQL)

Related information:

➡ 00C10207 (DB2 Codes)

DSNA208E   **DSNACA00 TCB=address CANNOT HAVE 2 SSIDS (ssid1 AND ssid2) FROM 1 TCB**

Explanation: An OPEN request was received specifying a different subsystem identifier (ssid2) than the previous CONNECT request (ssid1). Your task can only be connected to a single DB2 subsystem. The message is accompanied by return code (R15) 200 and code (R0) 00C10208.

System action: Control returns to the caller. Your request was not processed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Specify the correct DB2 subsystem identifier on both the OPEN and CONNECT requests.

Problem determination: See a DSNTRACE trace stream to understand the context of this message.

DSNA209E   **DSNACA00 A STORED PROCEDURE ISSUED AN EXPLICIT CALL TO DSNALI**

Explanation: A stored procedure issued an explicit call to DSNALI. This is not allowed. Return code 200 in register 15 and code 00C10209 in register 0 accompany this message.

System action: Control returns to the caller. The request was not processed. The DB2 call attachment facility is ready for more input.

System programmer response: Refer the problem to the programmer using the DB2 call attachment facility.

Programmer response: Remove explicit calls to DSNALI from the application program.

Problem determination: See a DSNTRACE trace job stream to understand the context of this message.

DSNA301I   **THE VALUE** *parm-value* **FOR PARAMETER** *parm-name* **IS NOT VALID**

Explanation: The stored procedure DSNACICS could not run because one of the input parameters contains an incorrect value.

System action: The stored procedure DSNACICS terminates without calling the CICS transaction.

System programmer response: Ensure that the user exit DSNACICX sets the parameter to a valid value.

Programmer response: Correct the input parameter value specified on the SQL CALL statement.

Severity: 12

DSNA302I   **THE REQUIRED INPUT PARAMETER** *parm-name* **CONTAINS NULLS**

Explanation: The stored procedure DSNACICS could not run because one or more of the required input parameters contains a null value.

System action: The stored procedure DSNACICS terminates without calling the CICS transaction.

System programmer response: Ensure that the user exit DSNACICX sets the parameter to a non-null value.

Programmer response: Correct the input parameter value in the SQL CALL statement, or ensure that the user exit DSNACICX sets the parameter to a non-null value.

Severity: 12

DSNA303I   **THE PARAMETER NETNAME WAS NOT SPECIFIED WHEN THE CONNECT TYPE OF SPECIFIC WAS SPECIFIED**

Explanation: The input parameter NETNAME must contain a non-null value when the CONNECT TYPE parameter is set to SPECIFIC.

System action: The stored procedure DSNACICS terminates without calling the CICS transaction.

System programmer response: Ensure that the user exit DSNACICX sets the parameter to a non-null value.

Programmer response: Correct the input parameter value in the SQL CALL statement, or ensure the the user exit DSNACICX sets the parameter to a non-null value.

Severity: 12

DSNA304I   **THE USER EXIT DSNACICX RETURNED RETURN CODE = X'hhhhhhhh' AND THE FOLLOWING ERROR MESSAGE:**

Explanation: The user exit DSNACICX returned a non-zero return code and a message to stored procedure DSNACICS. The message that DSNACICX returns immediately follows message DSNA304I.

System action: The stored procedure DSNACICS terminates without calling the CICS transaction.

System programmer response: Correct the cause of the error message that the user exit generated.

Programmer response: Contact the system programmer.

Severity: 12

DSNA305I   **THE CICS EXCI** *func* **REQUEST FAILED. CICS RESPONSE CODE = X'hhhh' CICS REASON CODE = X'hhhhhhhh'**

Explanation: DSNACICS uses the CICS EXCI Call interface to request the execution of a CICS server program. CICS EXCI interface call *func* resulted in an unexpected CICS response code and reason code. Possible values of *func* are:

Initialize_User
       Initializes the EXCI user environment.

Allocate_Pipe
>Allocates a session to a CICS region.

Open_Pipe
>Connects an allocated pipe to a receive session of the CICS region.

DPL_Request
>Issues a request to execute the CICS server program.

Close_Pipe
>Disconnects an open pipe from a CICS region.

Deallocate_Pipe
>Deallocates a pipe from a CICS region.

The failure can occur because the EXCI interface is not properly enabled for the CICS subsystem. For explanations of the CICS response code and reason code, see *CICS Transaction Server for z/OS External Interfaces Guide*.

System action: The stored procedure DSNACICS terminates without calling the CICS transaction.

System programmer response: Ensure that the EXCI interface is enabled to the CICS region that the stored procedure calls. If you suspect an error in the DB2 stored procedure, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Add a DSNDUMP DD statement to the JCL startup procedure for the stored procedure address space in which DSNACICS runs. Then rerun the application that calls DSNACICS. This causes DB2 to generate an SVC dump whenever it issues an error message for DSNACICS.

If you suspect an error in the DB2 stored procedure, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA306I   **THE CICS SERVER PROGRAM** *program* **ABENDED. CICS ABEND CODE=C'cccc'**

Explanation: The CICS server program that DSNACICS called abnormally terminated. CICS returns a four-character abend code to indicate the error. These error codes are described in *CICS Transaction Server for z/OS Problem Determination Guide*.

System action: Stored procedure DSNACICS processes the next request. CICS does a backout to the last syncpoint and forces the SYNCONRETURN option.

System programmer response: Correct the cause of the error in the server program. Ensure that the mirror transaction is still enabled after this error. The mirror transaction must be enabled so that subsequent requests to run a server program are accepted.

Programmer response: Correct the error in the server program.

Severity: 12

DSNA307I   **z/OS** *token-service* **REQUEST FAILED. SYSTEM RETURN CODE=X'hhhh'**

Explanation: DSNACICS uses the z/OS token services to save task-level information. The z/OS service returned an unexpected return code.

token-service is the token service function that DSNACICS requested. Possible values are:

Create_token
>Create a token using the IEANTCR service.

Retrieve_token
    Retrieve a token using the IEANTRT service.

See the description of IEANTCR or IEANTRT in *z/OS MVS Programming: Assembler Services Reference, Volumes 1 and 2* for a full description of the return code.

System action: Stored procedure DSNACICS processes the next request.

System programmer response: Add a DSNDUMP DD statement to the JCL startup procedure for the stored procedure address space in which DSNACICS runs. Then rerun the application that calls DSNACICS. This causes DB2 to generate an SVC dump whenever it issues an error message for DSNACICS.

If you suspect an error in the DB2 stored procedure, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
* CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA308I   THE CICS EXCI DPL_REQUEST REQUEST FAILED. CICS RESPONSE CODE=0. DPL RETAREA CONTAINS FIELD1=X'hhhhhhhh', FIELD2=X'hhhhhhhh', FIELD3=C'cccc'

Explanation: DSNACICS uses the CICS EXCI Call interface to request the execution of a CICS server program. The DPL_REQUEST command returned an unexpected value in the DPL_RETAREA. See *CICS Transaction Server for z/OS External Interfaces Guide* and *CICS Application Programming Guide* for a full description of the meanings of FIELD1, FIELD2, and FIELD3.

System action: Stored procedure DSNACICS processes the next request.

System programmer response: Add a DSNDUMP DD statement to the JCL startup procedure for the stored procedure address space in which DSNACICS runs. Then rerun the application that calls DSNACICS. This causes DB2 to generate an SVC dump whenever it issues an error message for DSNACICS.

If you suspect an error in the DB2 stored procedure, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
* CICS trace listing.

Programmer response: Correct any errors in the server program. Contact the system programmer.

Severity: 12

DSNA309I   UNABLE TO LOAD USER EXIT DSNACICX. SYSTEM REASON CODE=X'hhhh' SYSTEM COMPLETION CODE=X'hhhh'

Explanation: DSNACICS was unable to load the user exit routine DSNACICX.

System action: The stored procedure DSNACICS does not call the CICS transaction. DSNACICS processes the next request.

System programmer response: Ensure that the STEPLIB statement in the startup procedure for the stored procedure address space specifies the library that contains DSNACICX. A default user exit is shipped in the SDSNLOAD DB2 library.

Add a DSNDUMP DD statement to the JCL startup procedure for the stored procedure address space in which DSNACICS runs. Then rerun the application that calls DSNACICS. This causes DB2 to generate an SVC dump whenever it issues an error message for DSNACICS.

If you suspect an error in the DB2 stored procedure, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA310I   STORED PROCEDURE TASK CANCELLED BY OPERATOR

Explanation: The task that called the DSNACICS stored procedure was canceled by the operator.

System action: DSNACICS terminates immediately. The CICS server program might have been called.

System programmer response: Determine why the operator canceled the stored procedure task.

Programmer response: Contact the system programmer.

Severity: 12

DSNA311I   DSNACICS ABNORMALLY TERMINATED

Explanation: The task that called the DSNACICS stored procedure abnormally terminated.

System action: DSNACICS terminates immediately. The CICS server program might have been called. An SVC dump or a SYSMDUMP is generated for this error. If a CICS service encountered an error, a SYSMDUMP might be generated. For an abend in the stored procedure, an SVC dump is generated.

System programmer response: Review the JES messages for the stored procedure address space in which DSNACICS runs. Look for CICS or system messages that describe the source of the problem.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA312I   USER EXIT DSNACICX ABNORMALLY TERMINATED

Explanation: User exit program DSNACICX abnormally terminated.

System action: DSNACICS terminates immediately. The CICS server program is not called. An SVC dump is generated for this error.

System programmer response: Determine why the user exit abended. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.. Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA313I   **DSNACICS TERMINATED BY DB2 CANCEL THREAD COMMAND**

Explanation: The stored procedure DSNACICS terminated because the operator issued the DB2 CANCEL THREAD command.

System action: DSNACICS terminates immediately. The CICS server program might have been called.

System programmer response: Determine why the operator issued the DB2 CANCEL THREAD command.

Programmer response: Contact the system programmer.

Severity: 12

DSNA314I   **UNABLE TO LOAD CICS STUB ROUTINE DFHXCSTB. SYSTEM REASON CODE=X'hhhh'
SYSTEM COMPLETION CODE=X'hhhh'**

Explanation: DSNACICS was unable to load the CICS stub routine DFHXCSTB.

System action: The stored procedure DSNACICS does not call the CICS transaction. DSNACICS processes the next request.

System programmer response: Ensure that the STEPLIB statement in the startup procedure for the stored procedure address space specifies the CICS library that contains DFHXCSTB. DFHXCSTB is in the CICS EXCI library (SDFHEXCI). This library must be APF authorized.

Add a DSNDUMP DD statement to the JCL startup procedure for the stored procedure address space in which DSNACICS runs. Then rerun the application that calls DSNACICS. This causes DB2 to generate an SVC dump whenever it issues an error message for DSNACICS.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. Collect the following diagnostic items:

• Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
• Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
• CICS trace listing.

Programmer response: Contact the system programmer.

Severity: 12

DSNA315I   *stored-procedure* **FUNCTION** *func-name* **HAS COMPLETED SUCCESSFULLY.**

Explanation: The stored procedure, *stored-procedure*, executed successfully.

System action: None.

Programmer response: None.

DSNA316I   *stored-procedure* **INPUT PARAMETER ERROR. CALLER MUST PROVIDE INPUT FOR PARAMETER**
parm-name **WITH FUNCTION** *func-name***.**

Explanation: A value for parameter *parm-name* is required for function *func-name*.

System action: The execution of the stored procedure, *stored-procedure*, terminated before invoking IMS.

Programmer response: Provide an appropriate value for the parameter, and re-run the function.

DSNA317I   *stored-procedure* **ERROR IN** *otmaci-api* **API. RC=**_return-code_**, RSN1=**_reason-code1_**, RSN2=**_reason-code2_**,**
RSN3=_reason-code3_**, RSN4=**_reason-code4_**.**

Explanation: OTMA Callable Interface API *otmaci-api* encountered an error.

System action: The execution of the stored procedure, *stored-procedure*, terminated before invoking IMS.

Programmer response: Refer to the *IMS OTMA Guide and Reference* for explanations of the return and reason codes.

DSNA318I   *stored-procedure* **ERROR IN RRS CTXRCC API. RC=***return-code*

Explanation: An error was encountered in RRS when processing the two-phase commit.

System action: The execution of the stored procedure, *stored-procedure*, terminated before invoking IMS.

Programmer response: Refer to the *MVS Programming Resource Recovery* manual for an explanation of the return code.

DSNA319I   *stored-procedure* **RECEIVED UNKNOWN FUNCTION** *func-name*

Explanation: The stored procedure, *stored-procedure*, received an unknown function name in parameter 1 of the stored procedure call.

System action: The stored procedure terminated before invoking IMS.

Programmer response: Specify a function name known to the stored procedure in parameter 1.

DSNA320I   **DSNAIMS ERROR IN CTXSWCH DURING THE SWITCH AWAY FROM PRIVATE CONTEXT. RC=***return-code*

Explanation: An error was encountered in CTXSWCH API during the switch away from private context of DSNAIMS.

System action: DSNAIMS is terminated after executing CTXSWCH.

Programmer response: Contact IBM service for the return code.

DSNA321I   **DSNAIMS ERROR IN CTXSWCH DURING THE SWITCH TO PRIVATE CONTEXT. RC=***return-code*

Explanation: An error was encountered in CTXSWCH API during the switch to private context of DSNAIMS.

System action: DSNAIMS is terminated after executing CTXSWCH.

Programmer response: Contact IBM service for the return code.

DSNA601I   *csect-name* **THE PARAMETER** *parm-name* **IS NOT VALID, INVALID REASON CODE =** *xx*

Explanation: The parameter specified in the message is not valid. The cause of the invalid error is identified by the INVALID REASON CODE value in the message text. The specified parameter is not valid for the indicated reason:

1. Value is not an acceptable value.
2. Value is not unique.
3. Value is null.
4. Value is blank.
5. Value is not null.
6. Value is too long.
7. Named parameter is not known.
8. Named parameter is missing.

The *csect-name* in the message identifies the CSECT that issued the message.

System action: The stored procedure or the started task terminates.

Programmer response: Correct the input parameter value specified on the SQL CALL statement or on the start command.

Severity: 12 (severe)

DSNA602I   *csect-name* **INCORRECT DSN SUBCOMMAND** *subcommand* **RECEIVED**

Explanation: The DSN subcommand specified is not an acceptable value. The only supported DSN subcommands are: BIND, REBIND and FREE.

System action: Processing terminates.

Programmer response: Correct the invalid DSN subcommand.

Severity: 12 (severe)

DSNA603I   *csect-name* **THE COLUMN** *column-name* **IN TABLE** *table-name* **IS NOT VALID.** *row-identifier=row-id*.
          INVALID REASON CODE= *reason-code*

Explanation: An invalid value was supplied for *column-name* column.

csect-name
> The name of the control section that issued the message.

column-name
> The name of the column.

table-name
> The name of the table.

row-identifier=row-id
> Indicates the row that contains the invalid column.

reason-code
> A numeric value that indicates that reason that the specified column is not valid:
>
> **1**   The value is not an acceptable value.
>
> **2**   The length of the value is invalid.
>
> **3**   The value is null.

System action: Processing terminates.

User response: Specify a valid value for the specified column.

Severity: 12 (severe error)

DSNA604I   *csect-name* **DUPLICATE OBJECT-ID** *object-id* **IN TABLE** *table-name*

Explanation: A duplicate value was supplied for the object ID column in *table-name* table. The duplicate value was in the row with object ID *object-id* .

System action: Processing terminates.

User response: Specify a unique object-id for the object ID column in the named table.

Severity: 12 (severe)

DSNA605I   *csect-name* **THERE ARE** *object-num* **OBJECTS NOT FOUND IN THE DB2 CATALOG**

Explanation: There are *object-num* objects not found in the DB2 catalog.

System action: Objects not found in the DB2 catalog are not processed.

User response: Verify the name of the objects that were not processed.

Severity: 12 (severe)

DSNA606I   *csect-name* **UNABLE TO** *data-set-action* **DATA SET DSNAME=***dsname*, **MVS** *macname* **MACRO FAILED**
          WITH RETURN CODE=*xx*

Explanation: An MVS non-VSAM macro returned an unexpected return code when it was invoked.

The *csect-name* in the message identifies the CSECT that issued the message.

System action: Processing of the specified data set terminates.

Programmer response: Specify valid credentials in the SQL CALL statement. If the problem persists, ask your security administrator to check the security configuration for the specified user.

Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the MVS RETURN CODE associated with the failed MVS non-VSAM macro.

Severity: 12 (severe)

DSNA607I   *csect-name* **UNABLE TO** *member-action* **MEMBER=** *member* **OF DATA SET DSNAME=** *dsname*, **MEMBER NOT FOUND**

Explanation: The action on the library member failed because the library member was not found.

System action: Processing terminates.

User response: Specify a valid library for *member-name*.

Severity: 12 (severe)

DSNA608I   *csect-name* **UNABLE TO** *member-action* **MEMBER=***member* **OF DATA SET DSNAME=***dsname*, **MVS** *macname* **MACRO FAILED WITH RETURN CODE=***xx*

Explanation: An MVS non-VSAM macro returned an unexpected return code when it was invoked. The 'csect-name' in the message identifies the CSECT that issued the message.

System action: Processing of the specified library member terminates.

User response: Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the MVS RETURN CODE associated with the failed MVS non-VSAM macro.

Severity: 12 (severe)

DSNA609I   *csect-name* **UNABLE TO SUBMIT A JOB FOR EXECUTION, MVS** *macname* **MACRO FAILED WITH RETURN CODE=***xx*

Explanation: An error occurred when an MVS non-VSAM macro to open, write, or close the internal reader was called to write a complete job to the internal reader. The 'csect-name' in the message identifies the CSECT that issued the message.

System action: Processing terminates.

User response: Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the MVS RETURN CODE associated with the failed MVS non-VSAM macro.

Severity: 12 (severe)

DSNA610I   *csect-name* **INSUFFICIENT RACF AUTHORITY FOR DATA SET**

Explanation: A RACF error was encountered while processing a data set.

System action: Processing the data set terminates.

User response: Contact the system programmer, or system administrator to arrange for the required RACF or equivalent security authorization. If more information is needed, look at the messages in the console log.

Severity: 12 (severe)

DSNA611I   *csect-name* **THE** *data-set-attribute* **OF THE DATA SET DSNAME=***dsname* **IS NOT** *yy*

Explanation: The data set indicated in the message cannot be processed because the data set attribute is not supported.

System action: No action is taken with the named data set.

Programmer response: Specify a data set with the correct data set attribute.

Severity: 12 (severe)

DSNA612I   *csect-name* **INSUFFICIENT SPACE FOR DATA SET**

Explanation: Insufficient disk space was encountered while writing to a data set.

System action: Processing of the data set terminates.

Programmer response: Allocate more space for the data set. If more information is needed, look at the messages in the console log.

Severity: 12 (severe)

DSNA613I   *csect-name* **THE DATA SET NAME DSNAME=***dsname* **MUST NOT BEGIN WITH A PATTERN MATCHING CHARACTER**

Explanation: The specified data set name begins with an '*' or '%'. This is not allowed for filtering, because it will cause a massive search of the catalog and could tie up a lot of system resources.

System action: No action is taken with the named data set.

Programmer response: Correct the invalid data set name.

Severity: 12 (severe)

DSNA614I   *csect-name* **UNABLE TO OBTAIN** *storage-usage* **STORAGE, SIZE = ** *size* **BYTES**

Explanation: An error occurred when the stored procedure attempted to obtain *size* bytes of storage for the *storage-usage* indicated. The *csect-name* in the message identifies the CSECT that issued the message.

System action: The stored procedure terminates.

Programmer response: Notify the system programmer before attempting to reissue the SQL CALL statement.

Severity: 12 (severe)

DSNA615I   *csect-name* **UNABLE TO LOAD** *module-name* **MODULE, LOAD R15=***rc*

Explanation: An error resulted when an attempt was made to load the module indicated in the message.

System action: Processing terminates.

System programmer response: Verify that the named load module exists, and that DB2 can access it.

Programmer response: Notify the system programmer.

Severity: 12 (severe)

DSNA616I   *csect-name* **CATALOG SEARCH INTERFACE ERROR, R15=***xx***, REASON CODE= ** *yy* **RETURN CODE=***zz*

Explanation: An error occurred when the Catalog Search Interface (CSI) was called.

System action: Processing terminates.

Programmer response: Analyze the R15 reason code and return code values, and take the appropriate action.

Severity: 12 (severe)

Related information:

➟ http://publibfp.dhe.ibm.com/cgi-bin/bookmgr/BOOKS/EDCLB1A0/CCONTENTS

DSNA617I   *csect-name* **IFCA ERROR, IFCARC1=***ifcarc1-value***, IFCARC2=***ifcarc2-value***, IFCABNM=***ifcabnm-value***, IFCAGRSN=***ifcagrsn-value***, IFCAGBNM=***ifcagbnm-value*

Explanation: An error occurred when submitting a DB2 command through the instrumentation facility interface (IFI). Tokens in the message indicate values of fields in the IFI Communication Area (IFCA):

ifcarc1-value
        The X'E6......' return code value, from the IFCARC1 field.

ifcarc2-value
>   The X'E6......' reason code value, from the IFCARC2 field.

ifcabnm-value
>   The number of bytes that did not fit in the return area, from the value of the IFCABNM field.

ifcagrsn-value
>   Reason code for the situation in which an IFI calls requests data from members of a data sharing group, and not all the data is returned from group members, from the IFCAGRSN field.

ifcagbnm-value
>   Total length of data that was returned from other data sharing group members and did not fit in the return area, form the IFCAGBNM field.

System action: Processing terminates.

Programmer response: To diagnose the error, refer to the reason codes, and other IFCA field values that accompany the message to determine the cause of the error.

Severity: 12 (severe error)

Related concepts:

⯈ X'E6......' codes (DB2 Codes)

Related reference:

⯈ Instrumentation facility communications area (IFCA) (DB2 Performance)

DSNA618I   *csect-name* **SQL ERROR DURING SQL STATEMENT** *sql-statement*, *object-type* **=** *object-name*

Explanation: An SQL error resulted when the stored procedure issued an SQL statement for the indicated DB2 object.

csect-name
>   The name of the control section that issued the message.

sql-statement
>   The SQL statement that encountered the error.

object-type
>   The type of DB2 object that the failed SQL statement is processing.

object-name
>   The name of the DB2 object that the failed SQL statement is processing

System action: The stored procedure that issued the SQL statement terminates.

Information is written to the SQLCA. The information can be either:
- A formatted form of the SQLCA and a text message that is based on the SQLCODE field of the SQLCA.
- An unformatted form of the SQLCA.

Programmer response: Locate the additional information that is written to the SQLCA and take the appropriate action or actions.

Related concepts:

⯈ SQL error codes (DB2 Codes)

DSNA619I   *csect-name* **SUBSYSTEM INTERFACE ERROR, FUNCTION CODE=**n, **RETURN CODE=**xx,
              SSOBRETN=yy, *additional-info*

Explanation: An error occurred when a subsystem interface (SSI) function was called. The 'csect-name' in the message identifies the CSECT that issued the message. The SSI function codes are:
- 54 - Request subsystem version information call
- 79- SYSOUT Application Program Interface
- 80 - Extended status function call

System action: Processing terminates.

Programmer response: Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the MVS RETURN CODE associated with the failed MVS non-VSAM macro.

Severity: 12 (severe)

DSNA620I   *csect-name* **ERROR ALLOCATING DATA SET DSNAME=***dsname*, **RETURN CODE=***xx*, **ERROR REASON CODE=***yy*, **INFO REASON CODE=***zz*

Explanation: An error occurred when the DYNALLOC macro was invoked to dynamically allocate the named data set.

System action: Processing terminates.

User response: Analyze the DYNALLOC RETURN CODE, error reason code, and the informational reason code values, and take the appropriate action.

Severity: 12 (severe)

Related reference:

➦ Interpreting reason and return codes issued by DYNALLOC (MVS Programming Authorized Assembler Services Reference)

DSNA621I   *csect-name* **ATTEMPT TO PERFORM RRS ATTACH FUNCTION** *funcname* **FAILED, RRS ATTACH RC=***rc*, **RETCODE=***return-code* **REASCODE=***reason-code*

Explanation: RRSAF returned an unexpected return code when it was called to connect to DB2.

System action: Processing terminates.

User response: Take the appropriate action for the specified RRSAF attach, return, and reason codes.

Severity: 12 (severe)

Related concepts:

➦ RRSAF return codes and reason codes (DB2 Application programming and SQL)

Related tasks:

➦ Invoking the Resource Recovery Services attachment facility (DB2 Application programming and SQL)

➦ Controlling RRS connections (DB2 Administration Guide)

DSNA622I   *csect-name* **THE** *function-name* **FUNCTION CALL IS UNSUCCESSFUL,** *reason*

Explanation: A z/OS C/C++ function returned an error.

csect-name
        The name of the control section that issued the message.

function-name
        The name of the z/OS C/C++ function.

reason   The run time error message string that corresponds to the error code *errno*, followed by the *errno2* value. (*errno* and *errno2* are z/OS C/C++ function error code values.)

System action: Processing terminates.

User response: See the information about z/OS XL C/C++ run time library functions for descriptions of the error codes and take the appropriate action.

Severity: 12 (severe error)

Related reference:

➦ z/OS XL C/C++ runtime library functions (C/C++ Run-Time Library Reference)

DSNA623I   *csect-name* **THE PROCEDURE** *procedure-name* **DID NOT COMPLETE SUCCESSFULLY, RETURN CODE=***xx*

Explanation: The procedure *procedure-name* did not complete successfully when it was internally invoked, returning the indicated return code value. This message is followed by the error message returned by the named procedure, if applicable.

System action: Processing terminates.

User response: Review the error message returned by the named stored procedure.

Severity: 12 (severe)

DSNA624I   *csect-name* **THE STORED PROCEDURE ABENDED**

Explanation: The stored procedure abended and did not complete successfully.

System action: Processing terminates.

User response: Report the problem to the IBM Support Center.

Severity: 12 (severe)

DSNA625I   *csect-name* **THE STORED PROCEDURE TASK WAS TERMINATED BY THE OPERATOR**

Explanation: The stored procedure did not complete successfully, because it was terminated by the operator.

System action: Processing terminates.

User response: Issue the SQL CALL statement again.

Severity: 12 (severe)

DSNA626I   *csect-name* **THE** *function-name* **FUNCTION CALL IS UNSUCCESSFUL,** *parameter-name***=***parameter-value***,** *reason*

Explanation: A z/OS C/C++ function or a TSO/E function returned an error when the specified parameter value was passed to the function.

csect-name
> The name of the control section that issued the message.

function-name
> The name of the z/OS C/C++ or TSO/E function.

parameter-name
> The name of the function parameter.

parameter-value
> The value of the function parameter.

reason   For z/OS C/C++ functions, *reason* is the run time error message string that corresponds to the error code *errno*, followed by the *errno2* value. (*errno* and *errno2* are z/OS C/C++ function error code values.)

> For TSO/E functions, *reason* can be one of the following values:
> - If *function-name* is LISTDSI, *reason* is the LISTDSI reason code (SYSREASON).
> - If *function-name* is SYSDSN, *reason* is the message that is returned by SYSDSN.

System action: Processing terminates.

User response: For z/OS C/C++ functions, see the information about z/OS XL C/C++ run time library functions for descriptions of the error codes and take the appropriate action.

For TSO/E functions, see the information about LISTDSI and SYSDSN for descriptions of the TSO/E function errors and take the appropriate action.

Severity: 12 (severe error)

Related reference:

⮕ z/OS XL C/C++ runtime library functions (C/C++ Run-Time Library Reference)

⮕ LISTDSI (TSO/E REXX Reference)

⮕ SYSDSN (TSO/E REXX Reference)

DSNA627I   *csect-name* **THE** *funcname* **FUNCTION CALL IS UNSUCCESSFUL, RETURN CODE=** *xx*, **ERROR REASON CODE=***yy*, **INFO REASON CODE=***zz*

Explanation: A z/OS C/C++ function returned an unexpected return code. The 'csect-name' in the message identifies the CSECT that issued the message. The possible names of the z/OS C/C++ function are:
- dynalloc(): Allocate a data set
- dynfree(): Deallocate a data set

An error occurred when a subsystem interface (SSI) function was called. The 'csect-name' in the message identifies the CSECT that issued the message. The SSI function codes are:
- 54 - Request subsystem version information call
- 79- SYSOUT Application Program Interface
- 80 - Extended status function call

System action: If the function name is dynalloc(), then processing terminates.

Programmer response: Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for information on the DYNALLOC RETURN CODE, ERROR REASON CODE, and INFO REASON CODE values.

Severity: 12 (severe) if the function name is dynalloc(). If the function name is dynfree(), then the Severity is the dynfree() return code.

DSNA628I   *csect-name* **THE STORED PROCEDURE** *procedure-name* **ENCOUNTERED AN ERROR WHILE USING THE EXTENDED MCS CONSOLE TO ISSUE THE MVS SYSTEM COMMAND** *command-name*. *additional-information*

Explanation: The stored procedure encountered an error when it attempted to activate the extended MCS console or retrieve messages that are queued to the extended MCS console.

csect-name
> The name of the control section that issued the message.

procedure-name
> The name of the stored procedure.

command-name
> The system command that was issued.

additional-information
> A string that provides more details about the reason for the message.

System action: Processing terminates.

Programmer response: For each of the macros that are described in *additional-information*, look up the associated return and reason codes in the MVS Programming Authorized Assembler Services information.

Related reference:

⮕ MVS Programming Authorized Assembler Services Reference, Volume 4 (SET-WTO)

DSNA629I   *csect-name* **THE PARAMETER** *parm-name* **RETURNED BY THE STORED PROCEDURE** *proc-name* **IS NOT VALID. INVALID REASON CODE=***xx*

Explanation: An invalid value was returned by the indicated procedure for the specified parameter. The cause of the invalid error is identified by the invalid reason code value in the message text. The specified parameter is not valid for the indicated reason:

3   Value is null.

System action: Processing terminates.

User response: Report the problem to the IBM Support Center.

Severity: 12 (severe)

DSNA631I   *csect-name* **THE** *jes-version* **COMMAND** *command* **WAS ISSUED BUT NOT ACCEPTED**

Explanation: The command response received does not indicate whether the command was accepted or completed. The 'csect-name' in the message identifies the CSECT that issued the message. The possible commands are:
- CANCEL - Cancel a job
- PURGE - Purge a job

System action: Processing completes. The RETURN_CODE output parameter is 8 if the command was issued by a stored procedure.

User response: Issue the JES2 command $D JOB or the JES3 command *INQUIRY,J to display the current status of the job.

Severity: 8 (error)

DSNA632I   *csect-name* **JOB SUBMISSION FAILED**

Explanation: An error occurred while submitting a job for execution. A job identifier was not returned when the ENDREQ macro was issued after a complete job was written to the internal reader. The 'csect-name' in the message identifies the CSECT that issued the message.

System action: Processing terminates.

User response: Review the submitted JCL records to ensure that it has a valid JOB statement.

Severity: 12 (severe)

DSNA634I   *csect-name* **THE JES-MANAGED SYSOUT DATA SET IS EMPTY, JOBID=***jobid*

Explanation: A request was made to process SYSOUT data sets residing on JES spool for the indicated job identifier specified. However, the indicated job does not have any SYSOUT data sets.

System action: No action is taken with the specified job identifier.

User response: If the job action was requested by a stored procedure, correct the invalid job identifier on the SQL Call statement.

Severity: 12 (severe)

DSNA635I   *csect-name* **AN INTERNAL ERROR OCCURRED.** *additional-info*

Explanation: An error has occurred, as indicated by additional information in the message:

additional-info
        A string that provides additional details about the reason for the message.

System action: Processing terminates.

User response: When the value of *additional-info* refers to the SMS INTERFACE, see the information about SMS reason codes to diagnose the problem. Otherwise, contact IBM Software Support.

Severity: 12 (severe error)

Related reference:

Storage Management Subsystem Reason Codes(z/OS DFSMSdfp Diagnosis)

DSNA636I   *csect-name* **DB2 THREAD WAS CANCELED**

Explanation: The thread running the stored procedure was cancelled.

System action: Processing terminates.

User response: If necessary, issue the SQL CALL statement again.

Severity: 12 (severe)

DSNA637I   *csect-name* **I/O ERROR ON DATA SET**

Explanation: An I/O error occurred while processing a data set or library member.

System action: Processing terminates.

User response: Look at the messages in the console log for more information. If necessary, report the problem to the IBM Support Center.

Severity: 12 (severe)

DSNA638I   *csect-name* **THE SEQUENCE NUMBER** *seq* **IN ROW** *rownum* **OF TABLE** *table-name* **IS OUT OF SEQUENCE**

Explanation: The sequence number of the rows in the indicated table is not in ascending sequence, or the row sequence number does not start with 1.

System action: Processing terminates.

User response: Correct the sequence number of the rows in the indicated table.

Severity: 12 (severe)

DSNA639I   *csect-name* **THE NUMBER OF ROWS** *table-size* **IN THE INPUT TABLE** *table-name* **AND THE NUMBER OF OBJECTS** *objnum* **SPECIFIED IN THE PARAMETER** *parm-name* **DO NOT MATCH**

Explanation: The number of rows in the indicated table did not match the number of objects specified in the named parameter.

System action: Processing terminates.

User response: Correct the number of objects specified in the indicated parameter, or repopulate the named table with the correct number of rows.

Severity: 12 (severe)

DSNA640I   *csect-name* **SHUTDOWN DURATION** *xxxx* **REACHED, SHUTDOWN INITIATED**

Explanation: When the overall execution time reached the indicated shutdown duration, shutdown was initiated. When shutdown is initiated, current utility executions run to completion but no new utility will be started.

System action: Current utility executions are allowed to run to completion, but no new utility will be started.

User response: Determine which objects have not been processed due to the shutdown. Call SYSPROC.ADMIN_UTL_SCHEDULE again to process the unprocessed objects. If necessary, increase the SHUTDOWN_DURATION input parameter on the SQL CALL statement.

Severity: 4 (warning)

DSNA643I   *csect-name* **INVALID CREDENTIALS, ERRNO=***errno*

Explanation: The stored procedure is unable to log in because the USERID or PASSWORD parameter in the SQL CALL statement is invalid.

The *csect-name* in the message identifies the CSECT that issued the message.

System action: The stored procedure terminates.

Programmer response: Specify valid credentials in the SQL CALL statement. If the problem persists, ask your

security administrator to check the security configuration for the specified user.

Severity: 12 (severe)

DSNA644I   **THE STORED PROCEDURE** *procedure-name* **CAN ONLY BE EXECUTED IN** *mode* **MODE.**

Explanation: A stored procedure has been called in a DB2 mode that is not allowed.

procedure-name
        The name of the stored procedure.

mode   The mode that is required for this stored procedure.

System action: Execution of the stored procedure stops.

User response: Change DB2 to the mode that is identified by *mode* in the message and run the stored procedure again.

Severity: 12 (severe error)

DSNA645I   *csect-name* **THE PARAMETER MAX_INVOCATIONS IS NOT 1 ALTHOUGH NO SCHEDULE IS DEFINED**

Explanation: This input parameter must be set to 1 when parameters INTERVAL, POINT_IN_TIME and TRIGGER_TASK_ID are null.

The *csect-name* in the message identifies the CSECT that issued the message.

System action: The stored procedure terminates.

Programmer response: Specify 1 for MAX_INVOCATIONS in the SQL CALL statement.

Severity: 12 (severe)

DSNA646I   *csect-name* **THE PARAMETER** *parm-name* **HAS AN INVALID FORMAT, POS =** *xx*

Explanation: The specified parameter in the message has an invalid format. The error was detected at position *xx* in the parameter value.

The *csect-name* in the message identifies the CSECT that issued the message.

System action: The stored procedure terminates.

Programmer response: Correct the value for the specified input parameter in the SQL CALL statement.

Severity: 12 (severe)

DSNA648I   **THE ACTION** *action-name* **IN ALERT** *alert-id* **IS NOT SUPPORTED.**

Explanation: The stored procedure ADMIN_UTL_EXECUTE did not recognize the action to be taken to process the alert.

action-name
        The action that the stored procedure could not recognize.

alert-id   The alert identifier from the catalog table SYSIBM.SYSAUTOALERTS.

System action: The stored procedure continues to execute but the alert cannot be processed. The alert is ignored.

User response: Verify that the action is valid and the latest version of the stored procedure ADMIN_UTL_EXECUTE is installed.

Severity: 4 (warning)

Related reference:

▶ ADMIN_UTL_EXECUTE stored procedure (DB2 Performance)

DSNA649I   **ALERTS COULD NOT BE RESOLVED BY PROCEDURE. REASON** *reason-code*.

Explanation: The stored procedure ADMIN_UTL_EXECUTE could not resolve any alerts.

reason-code
> A numeric value that indicates the reason for the failure. Possible values are:

> **1**   ADMIN_UTL_EXECUTE was not executed within a maintenance time window.

> **2**   ADMIN_UTL_EXECUTE did not have enough time in the maintenance time window to resolve alerts.

System action: The stored procedure continues to execute, but the alerts cannot be processed. Alerts are ignored.

User response: Correct the condition that is indicated by the *reason-code* value.

Severity: 4 (warning)

Related reference:

▐➤ ADMIN_UTL_EXECUTE stored procedure (DB2 Performance)

DSNA650I   *csect-name* **CANNOT CONNECT TO ADMIN SCHEDULER** *scheduler-name*

Explanation: The stored procedure or user-defined function cannot connect to the specified administrative task scheduler. Either the name of the administrative task scheduler is incorrect or the administrative task scheduler is currently unavailable.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

System action: The stored procedure terminates.

System programmer response: Verify the name of the administrative task scheduler and change the ADMTPROC subsystem parameter, if necessary. Verify that the specified administrative task scheduler is up and running before calling this stored procedure again.

Severity: 12 (severe error)

Related reference:

▐➤ ADMIN SCHEDULER field (ADMTPROC subsystem parameter) (DB2 Installation and Migration)

DSNA651I   *csect-name* **DB2** *db2-ssid* **IS NOT ALLOWED TO CONNECT TO ADMIN SCHEDULER** *scheduler-name*

Explanation: The specified administrative task scheduler does not accept requests from the specified DB2 subsystem.

csect-name
> The name of the control section that issued the message.

db2-ssid The DB2 subsystem identifier. The administrative task scheduler parameter DB2SSID must be set to this value in order to accept requests from this DB2 subsystem.

scheduler-name
> The name of the administrative task scheduler.

System action: The stored procedure terminates.

System programmer response: Verify the name of the administrative task scheduler and change the ADMTPROC subsystem parameter, if necessary. Verify that the specified administrative task scheduler is set up to accept requests from the specified DB2 subsystem.

Severity: 12 (severe error)

Related reference:

➡ ADMIN SCHEDULER field (ADMTPROC subsystem parameter) (DB2 Installation and Migration)

DSNA652I   *csect-name* **THE USER** *user-name* **IS NOT ALLOWED TO ACCESS TASK** *task-id*

Explanation: The specified user, which is the value to which the special register CURRENT SQLID is set, is not allowed to update or delete the specified task. This user did not create the task and does not have SYSOPR, SYSCTRL, or SYSADM authority.

The *csect-name* in the message identifies the CSECT that issues the message.

System action: The stored procedure terminates.

Programmer response: Ensure that the DB2 special register (CURRENT SQLID) is set to the same user that added this task, or to a user that has SYSOPR, SYSCTRL, or SYSADM authority, before calling this stored procedure again.

Severity: 12 (severe)

DSNA653I   *csect-name* **CANNOT FIND ZPARM** *parm-value*

Explanation: The specified ZPARM is not defined.

The *csect-name* in the message identifies the CSECT that issues the message.

System action: The stored procedure terminates.

System programmer response: Define the specified ZPARM value.

Severity: 12 (severe)

DSNA654I   *csect-name* **THE TASK** *task-name* **DOES NOT EXIST**

Explanation: The specified task cannot be updated or removed because it does not exist in the administrative task scheduler task list.

csect-name
        The name of the control section that issued the message.

task-name
        The unique name of the scheduled administrative task.

System action: The stored procedure terminates.

Programmer response: Verify the task ID. Correct the task ID, if necessary, and resubmit the request.

Severity: 12 (severe error)

DSNA655I   *csect-name* **THE TASK** *task-name* **HAS AN EXECUTION STATUS THAT IS NOT COMPATIBLE WITH THE REQUESTED ACTION.**

Explanation: The specified task cannot be removed because it is currently executing, or it cannot be cancelled because it is not executing.

csect-name
        The name of the control section that issued the message.

task-name
        The unique name of the task to be removed or cancelled.

System action: The stored procedure terminates.

Programmer response: Wait for the completion of this task before trying to remove it. Use the DSNADM.ADMIN_TASK_STATUS user-defined function to check the execution status of the task.

Severity: 12 (severe)

DSNA656I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **TASK LIST IS FULL**

Explanation: The task list for the specified administrative task scheduler has reached its maximum size. No tasks can be added until other tasks are removed from the task list.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

System action: The stored procedure terminates.

Programmer response: Delete one or more tasks that have already been executed and whose status is no longer required.

Severity: 12 (severe error)

Related tasks:

↪ Removing a scheduled task (DB2 Administration Guide)

DSNA657I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT ACCESS ITS TASK LISTS**

Explanation: The specified administrative task scheduler cannot access any of its task lists to make the requested changes. Tasks cannot be added, removed, or updated.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

System action: The stored procedure and the administrative task scheduler both terminate.

System programmer response: Check the operator console messages for other messages about the administrative task scheduler, and take the appropriate action or actions.

At least one of the redundant administrative task scheduler task lists must be available to start the administrative task scheduler.

Severity: 12 (severe error)

Related concepts:

↪ Task lists of the administrative task scheduler (DB2 Administration Guide)

DSNA658I   *csect-name* **THE** *parameter-name* **PARAMETER WAS MODIFIED** *change-type* **IN MODULE** *module-name*
> **BY USER** *authorization-name*. **THE OLD SETTING WAS** *old-setting*. **THE NEW SETTING IS**
> *new-setting*.

Explanation: A user modified the value of the specified DB2 subsystem parameter by using the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure. This message is an informational message that is issued once for every parameter that was modified by this stored procedure.

csect-name
> The name of the control section that issued the message.

parameter-name
> The subsystem parameter for which the value was modified.

change-type
> The type of change, which can be one of the following values:

> **ONLINE**
> > SYSPROC.ADMIN_UPDATE_SYSPARM used the SET SYSPARM command to load module *module-name* and change the value of the subsystem parameter immediately.

OFFLINE
The new value of the subsystem parameter does not become effective until the next time that DB2
is started with module *module-name*.

module-name
The name of the subsystem parameter load module.

authorization-name
The authorization ID of the user who requested the change.

old-setting
The value of the subsystem parameter before the change was requested.

new-setting
The new value of the subsystem parameter.

System action: Processing continues.

User response: No action is required.

Related reference:

▶ ADMIN_UPDATE_SYSPARM stored procedure (DB2 Administration Guide)

▶ -SET SYSPARM (DB2) (DB2 Commands)

DSNA659I   *csect-name* **THE TASK** *task-name* **TRIGGERS DEPENDENT TASKS**

Explanation: The execution of the specified task triggers the execution of one or more tasks defined in the
administrative task scheduler task list. This task cannot be removed until the dependencies have been removed.

csect-name
The name of the control section that issued the message.

task-name
The unique name of the scheduled administrative task.

System action: The stored procedure terminates.

Programmer response: Remove the dependent tasks.

Severity: 12 (severe error)

DSNA660I   *csect-name* **AN INTERNAL ERROR HAS OCCURRED**

Explanation: The administrative task scheduler could not access internal resource that should be available.

csect-name
The name of the control section that issued the message.

System action: The stored procedure terminates.

System programmer response: Gather the administrative task scheduler trace, and contact IBM support.

Severity: 12 (severe error)

DSNA661I   *csect-name* **THE MACRO** *macro-name* **FAILED WITH RETURN CODE=***return-code*

Explanation: A DB2 stored procedure attempted to use a z/OS macro to identify attributes of a data set, and the
macro failed.

csect-name
The name of the control section that issued the message.

macro-name
The name of the z/OS macro.

return-code
A numeric value that indicates the reason for the failure. This return code is issued by z/OS.

System action: Processing continues.

Programmer response: Review the return codes for the macro that is identified by the *macro-name* value and take the indicated action.

Related reference:

⇨ DFSMS Macro Instructions for Data Sets

⇨ MVS Programming Authorized Assembler Services Reference, Volume 4 (SET-WTO)

Related information:

⇨ DFSMSdfp Advanced Services

DSNA662I *csect-name* **CATALOG MANAGEMENT HAS DETECTED AN ERROR FOR** *entry-name*. **RETURN CODE=***return-code***, REASON CODE=***reason-code*

Explanation: A catalog error occurred when a DB2 stored procedure attempted to use the Catalog Search Interface to obtain information about data sets.

csect-name
> The name of the control section that issued the message.

entry-name
> The entry name returned by the Catalog Search Interface.

return-code
> A numeric value that indicates the reason for the failure. This return code is issued by catalog management.

reason-code
> A numeric value that indicates the reason for the failure. This reason code is issued by catalog management.

System action: Processing continues.

Programmer response: Use the information provided for the return and reason codes for message IDC3009I to correct the catalog error.

Related reference:

⇨ Message IDC3009I (MVS System Messages)

DSNA663I *csect-name* **THE ATTRIBUTES OF** *dataset-count* **DATA SETS HAVE NOT BEEN SUCCESSFULLY RETRIEVED.**

Explanation: The DB2 stored procedure ADMIN_DS_LIST could not retrieve all attributes of one or more data sets.

csect-name
> The name of the control section that issued the message.

dataset-count
> The number of rows in the result set where the ERRMSG column is not blank. If a data set spans more than one volume, and ADMIN_DS_LIST could not retrieve all the attributes of a volume where this data set resides, this volume counts as one row in the count.

System action: Processing continues.

Programmer response: Scan the ERRMSG column of the result set to identify the data sets whose attributes have not all been successfully retrieved. Use the information provided in the ERRMSG column to correct the data set error.

DSNA664I *csect-name* **THE VOLUME SPACE STATISTICS OF***object-count* **OBJECTS HAVE NOT BEEN SUCCESSFULLY RETRIEVED.**

Explanation: The ADMIN_INFO_SMS stored procedure was unable to retrieve the volume space statistics for one or more objects.

csect-name
> The name of the control section that issued the message.

object-count
> The number of rows in the result set that contain non-blank values in the ERRMSG column.

System action: Processing continues.

Programmer response: Scan the ERRMSG column of the result set to identify the objects whose volume space statistics have not been successfully retrieved. Use the information provided in the ERRMSG column to correct the error.

Related reference:

ADMIN_INFO_SMS stored procedure (DB2 Administration Guide)

DSNA665I   *csect-name* **THE ALERT** *skipped-alert* **CANNOT BE EXECUTED. REASON=***reason-text*

Explanation: The stored procedure ADMIN_UTL_EXECUTE did not execute the indicated alert because it executed a different alert instead.

csect-name
> The name of the control section that issued the message.

skipped-alert **and** *alternate-alert*
> Unique alert IDs. These values represent the primary key (ALERT_ID column) of a row in the SYSIBM.SYSAUTOALERTS table.
>
> This message always includes two alert IDs. The first alert ID in the message (*skipped-alert*) is not executed. The second alert ID in the message (*alternate-alert*, which is part of the *reason-text*), is executed instead of the skipped alert. The two alert IDs refer to the same object, such as a table space.

reason-text
> A brief explanation of why the skipped alert was not executed, including the ID of the alert that was executed in its place.
>
> **ACTION OVERRULED BY ALERT** *alternate-alert*
> > The action to solve the skipped alert is overruled by the action to solve the indicated one.
>
> **ALREADY EXECUTING SIMILAR ALERT** *alternate-alert*
> > The action to solve the skipped alert is the same as the action to solve the indicated one, and the action is currently executing.

System action: The skipped alert (*skipped-alert*) is not executed and is marked as completed. The alternate alert (*alternate-alert*) is executed instead.

User response: No action is required.

Severity: 0

Related reference:

ADMIN_UTL_EXECUTE stored procedure (DB2 Performance)

DSNA666I   *csect-name* **A REQUIRED ACTION WAS NOT COMPLETED AFTER THE SUBSYSTEM PARAMETERS WERE MODIFIED. ACTION-TYPE=** *action-type***. REASON CODE=***reason-code***.** *additional-info*

Explanation: A user modified the value of at least one DB2 subsystem parameter by using the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure. However, DB2 did not complete a required or requested action after the subsystem parameter values were changed.

Message DSNA666I is returned in the result table of the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure.

csect-name
> The name of the control section that issued the message.

action-type
> A numeric value that identifies the action that failed. *action-type* can be one of the following values:
>
> **01**   The SET SYSPARM command with the LOAD option was not invoked.
>
> **02**   The message DSNA658I was not displayed on the JES SYSLOG.

reason-code
> A numeric value that indicates the reason for the failure. *reason-code* can be one of the following values:

DSNA667I

	01	The meaning depends on the *action-type* value as follows:

01   The meaning depends on the *action-type* value as follows:
- If the *action-type* is 01, at least one modified subsystem parameter cannot be changed online.
- If the *action-type* is 02, the stored procedure encountered an error while displaying a DSNA658I message on the JES SYSLOG.

02   Synchronous processing of the SET SYSPARM command is not supported.

additional-info
> Additional descriptive text that explains the reason that the specified action failed.
>
> Although *additional-info* is part of message DSNA666I, it is not returned in the same result set row. *additional-info* is returned in a separate row after DSNA666I.

System action: Processing continues.

User response: Depending on the value of *action-type* and *reason-code* take one of the following actions:

action-type **value**	*reason-code* **value**	Corrective action
01	01	In the input table SYSIBM.SYSPARM_SETTINGS, specify only those subsystem parameters that can be changed online.
01	02	Ensure that APAR PM40501 is installed.
02	01	Check for the return code from module DSNADMCC in the *additional-info* field and report the problem to IBM Software Support.

Related reference:

➡ ADMIN_UPDATE_SYSPARM stored procedure (DB2 Administration Guide)

➡ -SET SYSPARM (DB2) (DB2 Commands)

Related information:

"DSNA658I" on page 22

DSNA667I   *csect-name* **AN ERROR WAS ENCOUNTERED WHEN ADDING OR DELETING THE** *environment-name* **HOST COMMAND ENVIRONMENT USING** *function-name*. **ACTION=**ance*action-type*. **RC=***return-code*.

Explanation: A DB2 stored procedure attempted to add or delete a host command environment, but the host command environment add or delete failed.

csect-name
> The name of the control section that issued the message.

environment-name
> The name of the host command environment: DSNREXX or SDSF.

function-name
> The function that was invoked: RXSUBCOM or ISFCALLS.

action-type
> The action that was specified: ADD or DELETE.

return-code
> A numeric value that provides information about the failure. This return code is issued by the host command environment.
>
> If *function-name* is RXSUBCOM and *return-code* is 40, RXSUBCOM found an error in the parameter list passed to it, such as too many or too few parameters.

System action: Processing terminates.

Programmer response: Review the return code information for the host command environment and take the indicated action.

Related reference:

➡ Return codes for ISFEXEC (z/OS SDSF Operation and Customization)

➡ Return codes for IRXSUBCM (TSO/E REXX Reference)

DSNA668I   *csect-name* **AN ERROR WAS ENCOUNTERED WHEN INVOKING THE HOST ENVIRONMENT COMMAND** *host-command* **WITH OPERAND** *operand-name*. **RC=***return-code*.

Explanation: A DB2 stored procedure invoked a host environment command and the command failed.

csect-name
> The name of the control section that issued the message.

host-command
> The name of the host environment command: ISFEXEC or ISFLOG.

operand-name
> The name of the operand that was specified for the host command.

return-code
> A numeric value that provides information about the failure. This return code is issued by the host command environment.

If *return-code* is 12 and message DSNA668I is followed by the following SDSF messages, the stored procedure must be executed on a system that has z/OS V1R12 or later installed.

```
ISF302E string was seen in command position
1 where one of the following was expected:
END-OF-STMT, ISFACT, ISFEXEC, ISFGET.
ISF767I Request completed
```

System action: Processing terminates. This message might be followed by one or more messages from the host environment command.

Programmer response: Check for additional messages from the host environment command. Review the return code information for the host environment command and take the indicated action.

Related reference:

➡ Return codes for ISFEXEC (z/OS SDSF Operation and Customization)

➡ Return codes for ISFLOG (z/OS SDSF Operation and Customization)

DSNA669I   *csect-name* **THE SUBSYSTEM PARAMETERS WERE NOT MODIFIED. REASON CODE=***reason-code*.
additional-info

Explanation: A user requested a change to the value of at least one DB2 subsystem parameter by using the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure. However, the subsystem parameters were not modified for the indicated reason.

Message DSNA669I is returned in the result table of the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure, unless the *reason-code* value is 01, 04, 05, 06, or 07. In these special cases, DSNA669I is displayed in the WLM job log.

csect-name
> The name of the control section that issued the message.

reason-code
> A numeric value that indicates the reason for the failure. *reason-code* can be one of the following values:
>
> **01**   A REXX programming violation occurred.
>
> **02**   The created global temporary table SYSIBM.SYSPARM_SETTINGS does not exist.

03	The user does not have the authority to use the created global temporary table SYSIBM.SYSPARM_SETTINGS.
04	The created global temporary table SYSIBM.UPDSYSPARM_MSG does not exist.
05	The user does not have authority to use the created global temporary table SYSIBM.UPDSYSPARM_MSG.
06	The DSNREXX package was not found.
07	DB2 cannot access a DSNREXX environment. DB2 REXX Language Support is not available.
08	All libraries in the WLM STEPLIB concatenation are APF-authorized.
09	An error occurred when validating an input parameter for the stored procedure.
10	The WLM startup procedure is missing the ZPMDFLTS DD statement.
11	A data set that is required by the stored procedure is not available for use.
12	A validation error occurred for input table SYSIBM.SYSPARM_SETTINGS for one of the following reasons: • The table is empty. • The table has a duplicate ROWNUM value. • The table has a negative ROWNUM value. • A macro name is invalid. • A parameter is invalid.
13	An SQL error occurred when retrieving new subsystem parameter values from the input table SYSIBM.SYSPARM_SETTINGS.
14	The specified *source_library* parameter for the ADMIN_UPDATE_SYSPARM stored procedure is not a partitioned data set (PDS) or a partitioned data set extended (PDSE).
15	The record format (RECFM) of the specified *source_library* parameter for the ADMIN_UPDATE_SYSPARM stored procedure is not F or FB.
16	The record length (LRECL) of the specified *source_library* parameter for the ADMIN_UPDATE_SYSPARM stored procedure is not 80.
17	The ZPMDFLTS DD statement in the WLM startup procedure points to a data set that contains a SRCLIB_NAME library data set. That SRCLIB_NAME data set is not a partitioned data set (PDS) or a partitioned data set extended (PDSE).
18	The ZPMDFLTS DD statement in the WLM startup procedure points to a data set that contains a SRCLIB_NAME library data set. The record format (RECFM) of that SRCLIB_NAME data set is not F or FB.
19	The ZPMDFLTS DD statement in the WLM startup procedure points to a data set that contains a SRCLIB_NAME library data set. The record length (LRECL) of that SRCLIB_NAME data set is not 80.
20	The backup of the member that the stored procedure is to replace with the new JCL to assemble and link-edit the new subsystem parameters load module did not complete successfully. The failure occurred for one of the following reasons: • A backup already exists. • An I/O error occurred.
21	One of the following errors occurred when retrieving the current values of the subsystem parameters: • The SYSPROC.ADMIN_INFO_SYSPARM stored procedure did not complete successfully. Either the return code was greater than 0 or the SQL code was not +466, as expected. • An SQL error occurred when processing the result set from the SYSPROC.ADMIN_INFO_SYSPARM stored procedure.
22	The required program ASMA90 was not found.
23	The required program HEWL (which executes the linkage editor) was not found.

24   The process of assembling the subsystem parameter macros did not complete successfully. The return code was greater than 0.

25   The backup of the subsystem parameters load module that is being replaced did not complete successfully for one of the following reasons:

- The backup already exists.
- A SYSDSN error occurred for a backup member.
- IEBCOPY failed to make a backup copy.

26   The process of link-editing the subsystem parameters load module did not complete successfully. The return code was greater than 4.

27   The SET SYSPARM command with the LOAD option did not complete successfully for one of the following reasons:

- The SYSPROC.ADMIN_COMMAND_DB2 stored procedure did not complete successfully. Either an IFI error occurred, the return code was greater than 0, or the SQL code was not +466, as expected.
- The SET SYSPARM command returned messages that indicate that the command was not successful.

28   The result of the SET SYSPARM command with the LOAD option is unknown. An SQL error occurred when processing the result set of the ADMIN_COMMAND_DB2 stored procedure.

32   An I/O error occurred when accessing assembler and link-edit files.

33   The JCL to assemble and link-edit the new subsystem parameters load module was not saved for one of the following reasons:

- A data set allocation error occurred.
- A data set I/O error occurred.

34   The SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure could not retrieve the subsystem name of the connected DB2 subsystem.

35   The SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure could not retrieve the subsystem name and command prefix of the DB2 subsystem whose subsystem parameters are to be modified. This situation occurred for one of the following reasons:

- The SYSPROC.ADMIN_COMMAND_DB2 stored procedure did not complete successfully. Either an IFI error occurred, the return code was greater than 0, or the SQL code was not +466, as expected.
- An SQL error occurred when processing the result set from the ADMIN_COMMAND_DB2 stored procedure.
- The version of DB2 is not supported.

additional-info
Additional descriptive text or another message that explains the reason that the specified action failed.

Although *additional-info* is part of message DSNA669I, it is not returned in the same result set row. *additional-info* is returned in a separate row after DSNA669I. The exception is if the *reason-code* value is 01, 04, 05, 06, or 07. In these special cases, DSNA669I and the additional text is displayed in the WLM job log.

System action: The SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure terminates, unless the error is encountered during cleanup or restore processing.

User response: Check the *additional-info* field for a more detailed explanation of the error and take the appropriate action.

- If a DSN message is specified, see the information for the specific DB2 for z/OS message.
- If an EXECIO error was encountered, see the information about EXECIO for a description of the return code.
- If an allocation error was encountered, see the information about the ALLOCATE TSO/E command for a description of the ALLOCATE return code. Also, check the ALLOCATE error message that was returned by the stored procedure. The ALLOCATE error message is in the result set row after the row where the ALLOCATE return code is specified.
- If an assembly, link-edit, or IEBCOPY error was encountered, check the corresponding listing that is returned by the stored procedure.

Related reference:

➡ ADMIN_UPDATE_SYSPARM stored procedure (DB2 Administration Guide)

➡ EXECIO (TSO/E REXX Reference)

➡ ALLOCATE command (TSO/E Command Reference)

DSNA670I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **IS STOPPING**

Explanation: The specified administrative task scheduler is stopping.

csect-name
          The name of the control section that issued the message.

scheduler-name
          The name of the administrative task scheduler.

System action: The administrative task scheduler stops.

User response: No action is required.

Severity: 0

DSNA671I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **IS STARTING**

Explanation: The specified administrative task scheduler is starting.

csect-name
          The name of the control section that issued the message.

scheduler-name
          The name of the administrative task scheduler.

System action: The administrative task scheduler starts.

User response: No action is required.

Severity: 0

DSNA672I   *csect-name action-description scheduler-name* **NORMAL COMPLETION**

Explanation: The indicated action completed successfully for the specified administrative task scheduler.

csect-name
          The name of the control section that issued the message.

action-description
          The action that has completed. Possible actions include:
          • START COMMAND FOR ADMIN SCHEDULER
          • STOP COMMAND FOR ADMIN SCHEDULER
          • MODIFY COMMAND FOR ADMIN SCHEDULER

scheduler-name
          The name of the administrative task scheduler.

System action: The administrative task scheduler executed the specified action.

User response: No action is required.

Severity: 0

DSNA673I   *csect-name action-description scheduler-name* **ABNORMAL TERMINATION**

Explanation: The described action was terminated because of an error in the specified administrative task scheduler.

csect-name
          The name of the control section that issued the message.

action-description
>   The action that did not complete. Possible actions include:
>   - START COMMAND FOR ADMIN SCHEDULER
>   - STOP COMMAND FOR ADMIN SCHEDULER
>   - MODIFY COMMAND FOR ADMIN SCHEDULER
>   - EXECUTION LOOP OF ADMIN SCHEDULER

scheduler-name
>   The name of the administrative task scheduler.

System action: The administrative task scheduler terminates.

Programmer response: Investigate other error messages that were displayed on the operator's console prior to this message.

Severity: 12 (severe error)

DSNA674I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **IS ALREADY ACTIVE**

Explanation: The administrative task scheduler that is starting up has detected another address space with the same name that is already running on the system. Only one instance of the administrative task scheduler is allowed to run.

csect-name
>   The name of the control section that issued the message.

scheduler-name
>   The name of the administrative task scheduler.

System action: The administrative task scheduler that was already running remains active. The one that was starting terminates.

User response: No action is required.

Severity: 12 (severe error)

DSNA675I   *csect-name module-name* **COULD NOT RUN BECAUSE IT WAS NOT INVOKED FROM AN APF-AUTHORIZED LIBRARY**

Explanation: The load module could not run.

csect-name
>   The name of the control section that issued the message.

module-name
>   The name of the load module.

Either of the following conditions can cause this error:
- The load module that is identified by *module-name* was not invoked from an APF-authorized library.
- One or more libraries in the STEPLIB or JOBLIB concatenation in the job that invokes the load module that is identified by *module-name* is not APF-authorized.

System action: The administrative task scheduler or the stored procedure is not started.

System programmer response: Ensure that all libraries in the STEPLIB and JOBLIB concatenation for the job that invokes *module-name* are APF-authorized.

User response: Notify the system programmer.

Severity: 12 (severe error)

DSNA676I   *csect-name* **SEVERE ERROR ENCOUNTERED WHILE STARTING OR STOPPING ADMIN SCHEDULER** *scheduler-name*

Explanation: A severe error has been detected in the specified administrative task scheduler.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

System action: The administrative task scheduler terminates.

Programmer response: Gather the administrative task scheduler trace, and contact IBM support.

Severity: 12 (severe error)

DSNA677I   *csect-name* **THE** *application-type procedure-name* **CAN ONLY CREATE** *threads-created* **OF** *threads-requested* **THREADS**

Explanation: There are not enough resources available to create all the requested threads.

csect-name
> The name of the control section that issued the message.

application-type **and** *procedure-name*
> The type and name of the application.

> **ADMIN SCHEDULER**
>> The procedure is the administrative task scheduler. The name that is provided is the name of a scheduled task.

> **STORED PROCEDURE**
>> The procedure is a stored procedure. The name that is provided is the name of the stored procedure.

threads-created
> The number of threads that were created.

threads-requested
> The number of threads that were requested.

System action: The administrative task scheduler or stored procedure terminates.

System programmer response: Take the action indicated by the value of *application-type*:

ADMIN SCHEDULER
> Either increase system resources to allow the administrative task scheduler to create the number of threads needed, or lower the maximum number of threads that the administrative task scheduler is allowed to use. The parameter MAXTHD controls the maximum number of threads that the administrative task scheduler is allowed to use.

STORED PROCEDURE
> Either reduce the number of parallel threads, or increase the value of the CTHREAD subsystem parameter to allow the stored procedure to create the number of threads indicated by the value of *threads-requested*.

Severity: 12 (severe error)

Related concepts:

➡ Multi-threading in the administrative task scheduler (DB2 Administration Guide)

Related tasks:

➡ Installation step 22: Set up the administrative task scheduler (DB2 Installation and Migration)

Related reference:

➡ MAX USERS field (CTHREAD subsystem parameter) (DB2 Installation and Migration)

DSNA678I   *csect-name* **THE NUMBER OF TASKS TO BE CONCURRENTLY EXECUTED BY THE ADMIN SCHEDULER** *scheduler-name* **EXCEEDS** *max-threads*

Explanation: The current number of tasks to be executed concurrently exceeds the maximum number of threads that is allowed by the specified administrative task scheduler.

csect-name
>   The name of the control section that issued the message.

scheduler-name
>   The name of the administrative task scheduler.

max-threads
>   The number of threads that were created.

System action: The administrative task scheduler continues running, but some scheduled tasks cannot be executed.

Programmer response: Change the execution schedule of some tasks, if possible.

Severity: 12 (severe error)

Related concepts:

� Multi-threading in the administrative task scheduler (DB2 Administration Guide)

Related tasks:

� Interacting with the administrative task scheduler (DB2 Administration Guide)

DSNA679I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT ACCESS TASK LIST** *task-list*, **REASON =** *reason-indicator*

Explanation: The specified administrative task scheduler cannot access the task list.

csect-name
>   The name of the control section that issued the message.

scheduler-name
>   The name of the administrative task scheduler.

task-list   The task list, which can be either a VSAM data set that contains the task list or the DB2 table SYSIBM.ADMIN_TASKS that contains the redundant task list.

reason-indicator
>   An indicator of the reason for the failure, which can be a VSAM access error, an SQL error code, or a DB2 error code.

System action: The administrative task scheduler continues running with access to the other task list only.

System programmer response: Verify the name of the VSAM data set and modify the administrative task scheduler started task procedure, if necessary. Stop and restart the administrative task scheduler with this new configuration.

Verify that DB2 is up and running and that the scheduler can connect to it and has access to the DB2 task list.

According to the reason specified, make sure the following conditions are met.
- The VSAM data set exists.
- The VSAM data is allocated.
- The VSAM data can be accessed from this system.
- The user who started the administrative task scheduler is allowed to access it for read and write.
- The DB2 system is up and running.
- The administrative task scheduler packages are correctly bound in DB2.
- The DB2 table SYSIBM.ADMIN_TASKS containing the task list exists.
- The user who started the administrative task scheduler is allowed to access it for read and write.

Severity: 12 (severe error)

DSNA681I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **TASK LIST** *vsam-list* **AND** *table-list* **ARE INCONSISTENT**

Explanation: The specified administrative task scheduler has detected that the redundant task lists do not match.

csect-name
>   The name of the control section that issued the message.

scheduler-name
>The name of the administrative task scheduler.

vsam-list
>The task list that is stored as a VSAM data set.

table-list
>The task list that is stored in a DB2 table.

System action: The administrative task scheduler tries recovering the older task list from the newer task list.

Programmer response: Look for additional messages that indicate the result of the recovery process.

Severity: 0

Related concepts:

➦ Architecture of the administrative task scheduler (DB2 Administration Guide)

DSNA682I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **TASK LIST** *task-list* **IS SUCCESSFULLY RECOVERED**

Explanation: The specified administrative task scheduler has successfully recovered a corrupted or outdated task list from the redundant task list.

csect-name
>The name of the control section that issued the message.

scheduler-name
>The name of the administrative task scheduler.

task-list   The task list that has been recovered.

System action: Both task lists are in a consistent state.

User response: No action is required.

Severity: 0

DSNA683I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **TASK LIST** *task-list* **CANNOT BE RECOVERED**

Explanation: The specified administrative task scheduler could not recover a corrupted or outdated task list from the redundant task list.

csect-name
>The name of the control section that issued the message.

scheduler-name
>The name of the administrative task scheduler.

task-list   The task list that could not be recovered.

System action: The administrative task scheduler continues running with access to only one task list.

Programmer response: Gather the administrative task scheduler trace, and contact IBM support.

Severity: 12 (severe error)

DSNA684I   *csect-name* **DFLTUID CANNOT BE THE USER** *user-name* **THAT STARTS THE ADMIN SCHEDULER** *scheduler-name*

Explanation: The specified administrative task scheduler started task parameter (DFLTUID) is set to the specified user name. This user is the same user that is associated with the started task. For security reasons, this assignment is not allowed.

csect-name
>The name of the control section that issued the message.

user-name
> The user name that was provided to start the administrative task scheduler. This name is specified in the DFLTUID parameter.

scheduler-name
> The name of the administrative task scheduler.

System action: The administrative task scheduler does not start.

System programmer response: Specify another user name in the DFLTUID parameter. The default execution user should have minimal access rights on the system, and should have no access to administrative task scheduler resources.

Severity: 12 (severe error)

Related tasks:

Installation step 22: Set up the administrative task scheduler (DB2 Installation and Migration)

Related reference:

Security guidelines for the administrative task scheduler (DB2 Administration Guide)

DSNA685I   *csect-name application-name action-type* **PROCESSING DID NOT COMPLETE SUCCESSFULLY. OBJECT-TYPE=***object-type***, OBJECT-NAME=***object-name***, OBJECT-BACKUP-NAME=***object-backup-name***.** *additional-info*

Explanation: One of the following events occurred:

- An object was not restored after the application that replaced it failed to complete successfully.
- A new object or a backup object was not deleted after the application that created it failed to complete successfully.
- A backup object was not deleted after that application the created it completed successfully.

Message DSNA685I is returned in the result table of the SYSPROC.ADMIN_UPDATE_SYSPARM stored procedure.

csect-name
> The name of the control section that issued the message.

application-name
> The name of the application.

action-type
> The type of action that did not complete. *action-type* can be one of the following values:
>
> **CLEANUP**
> > The cleanup processing that occurs when *application-name* completes, regardless of whether the application is successful. An attempt to delete either *object-name* or *object-backup-name* failed.
>
> **RESTORE**
> > The restore processing that occurs when *application-name* does not complete successfully and must restore all replaced objects to their state before the application was run. An attempt to restore an object from the object backup failed.

object-type
> A numeric value that identifies the type of object that was not cleaned up or restored. *object-type* can be one of the following values:
>
> **01** The JCL to assemble and link-edit the subsystem parameters load module.
>
> **02** The subsystem parameters load module.
>
> **03** The active subsystem parameters load module in storage.

object-name
> The name of the object.

object-backup-name
> The name of the backup object.

DSNA685I

additional-info
> Additional text that describes the failure.

> Although *additional-info* is part of message DSNA685I, it is not returned in the same result set row. *additional-info* is returned in a separate row after DSNA685I.

System action: Processing continues.

User response: Depending on the values in the message, take the action as indicated in the following table:

action-type **value**	*object-type* **value**	**Description of the error**	*object-name* **value**	*object-backup-name* **value**	**User response**
CLEANUP	01	The member that contains the new JCL to assemble and link-edit the new subsystem parameters load module was not deleted.	The name of the member with the new JCL	NA	Delete the member with the specified object name (*object-name*) from the JCL source library.
CLEANUP	01	The backup of the member that the stored procedure replaced with the new JCL to assemble and link-edit the new subsystem parameters load module was not deleted.	NA	The backup name of the member that the stored procedure replaced with the new JCL	Delete the member with the specified backup object name (*object-backup-name*) from the JCL source library.
CLEANUP	02	The member with the new subsystem parameters load module was not deleted.	The name of the member with the new subsystem parameters load module	NA	Delete the member with the specified object name (*object-name*) from the subsystem parameters load module library.
CLEANUP	02	The backup of the member that the stored procedure replaced with the new subsystem parameters load module was not deleted.	NA	The backup name of the member that the stored procedure replaced with the new subsystem parameters load module	Delete the member with the specified backup object name (*object-backup-name*) from the subsystem parameters load module library.
RESTORE	01	The member that the stored procedure replaced with the new JCL that assembles and link-edits the new subsystem parameters load module was not restored.	The name of the member with the new JCL	The backup name of the member that the stored procedure replaced with the new JCL	Rename the member with the specified backup object name (*object-backup-name*) in the JCL source library to the specified object name (*object-name*).

action-type **value**	object-type **value**	**Description of the error**	object-name **value**	object-backup-name **value**	**User response**
RESTORE	02	The member that the stored procedure replaced with the new subsystem parameters load module was not restored.	The name of the member with the new subsystem parameters load module	The backup name of the member that the stored procedure replaced with the new subsystem parameters load module	Rename the member with the specified backup object name (*object-backup-name*) in the subsystem parameters load library to the specified object name (*object-name*).
RESTORE	03	The replaced active subsystem parameters load module in storage was not restored.	The name of the subsystem parameters load module in storage before the stored procedure issued the SET SYSPARM command with the LOAD option.	NA	From the console, issue the SET SYSPARM command with the LOAD(*object-name*) clause.

Related reference:

➡ ADMIN_UPDATE_SYSPARM stored procedure (DB2 Administration Guide)

➡ -SET SYSPARM (DB2) (DB2 Commands)

DSNA686I　　*csect-name* **BIND QUERY FOR QUERYNO =** *queryno-value* **SUCCESSFUL, BUT SET_PLAN_HINT IS UNABLE TO RETRIEVE THE** *information-type*. **REASON:** *reason-text*

Explanation:　A BIND QUERY command issued by the SET_PLAN_HINT stored procedure has completed successfully. However, the SET_PLAN_HINT stored procedure cannot retrieve the BIND QUERY message or the QUERYID value.

csect-name
　　　　The name of the control section that issued the message.

queryno-value
　　　　The QUERYNO value that identifies the SQL statement.

reason-text
　　　　Text the describes why the message output for the BIND QUERY message was not be retrieved.

information-type
　　　　A string that indicates the type of information that was not retrieved:

- QUERY MESSAGE
- QUERYID

System action:　The stored procedure performs terminates normally.

User response:　No action is required.

Severity:　0

| **Related reference**:

| ➡ SET_PLAN_HINT stored procedure (DB2 Administration Guide)

| ➡ BIND QUERY (DSN) (DB2 Commands)

DSNA690I   *csect-name* **THE TASK** *task-name* **WAS RUNNING WHEN THE ADMIN SCHEDULER** *scheduler-name*
            STOPPED

Explanation: A task was executing when the specified administrative task scheduler was forced to stop. The status of this task is set to "unknown".

csect-name
        The name of the control section that issued the message.

task-name
        The unique name of the scheduled administrative task.

scheduler-name
        The name of the administrative task scheduler.

System action: The administrative task scheduler recovers from a blocked state.

System programmer response: Avoid forcing the administrative task scheduler to stop when tasks are currently running. During a normal shutdown of the administrative task scheduler, make sure that no tasks are running.

For a JCL task, make sure that the JCL job, which is indicated in the JOBID in the task execution status, terminates before the next scheduled execution of this task. If the job is still running from a previous scheduled task, it must be stopped before the task runs again.

Severity: 12 (severe error)

DSNA691I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT GENERATE A PASSTICKET FOR**
            TASK *task-name*, **USER** *user-name*

Explanation: The administrative task scheduler could not generate a PassTicket to switch the security environment. This PassTicket is required for the administrative task scheduler to be able to execute the specified task under the authority of the specified user.

csect-name
        The name of the control section that issued the message.

scheduler-name
        The name of the administrative task scheduler.

task-name
        The unique name of the scheduled administrative task.

user-name
        The user name that must execute the task. This user is not always indicated in the task, and could be the default ID, which is set in the DFLTUID parameter.

System action: The administrative task scheduler does not execute the specified task.

System programmer response: Check the security configuration and make sure that the administrative task scheduler is allowed to create PassTickets for any user.

Severity: 12 (severe error)

Related tasks:

➡ Installation step 22: Set up the administrative task scheduler (DB2 Installation and Migration)

Related reference:

➡ Security guidelines for the administrative task scheduler (DB2 Administration Guide)

DSNA692I *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT LOG IN AS USER** *user-name*
  RC=*return-code* **RSN=***reason-code*

Explanation: The specified administrative task scheduler cannot switch the security environment in order to execute the specified task under the authority of the specified user.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

user-name
> The user name that was indicated for the task.

task-name
> The unique name of the scheduled administrative task.

return-code
> The return-code returned by the z/OS UNIX System Services callable service pthread_security_np which was used to log in.

reason-code
> The reason-code returned by the z/OS UNIX System Services callable service pthread_security_np which was used to log in.

System action: The administrative task scheduler does not execute the specified task.

System programmer response: Check the security configuration and make sure that the specified user exists and can log in.

Severity: 12 (severe error)

Related reference:

⇨ thread_security_np(), pthread_security_applid_np() -- Create or Delete Thread-level Security (C/C++ Run-Time Library Reference)

⇨ Return codes (errnos) z/OS UNIX System Services: Messages and Codes

⇨ Reason codes (errnojrs) z/OS UNIX System Services: Messages and Codes

---

DSNA693I *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT CONNECT TO DB2** *db2-ssid* **TO EXECUTE TASK** *task-name*

Explanation: The specified administrative task scheduler was unable to connect to the specified DB2 subsystem when executing the specified stored procedure task.

csect-name
> The name of the control section that issued the message.

scheduler-name
> The name of the administrative task scheduler.

db2-ssid The DB2 subsystem identifier.

task-name
> The unique name of the scheduled administrative task.

System action: The administrative task scheduler does not execute the specified task.

Programmer response: Make sure that DB2 is up and running when the task is scheduled to run. Make sure that the user that is to execute the task is allowed to connect to DB2.

Severity: 12 (severe error)

DSNA694I   *csect-name* **THE ADMIN SCHEDULER** *scheduler-name* **CANNOT PROCESS PARAMETER PROCEDURE_INPUT OF TASK** *task-name*

Explanation: The parameter PROCEDURE_INPUT for the specified task does not specify an SQL SELECT statement that returns one row of data.

csect-name
　　　　The name of the control section that issued the message.

scheduler-name
　　　　The name of the administrative task scheduler.

task-name
　　　　The unique name of the scheduled administrative task.

System action: The administrative task scheduler does not execute the specified task.

Programmer response: Remove the task and add it again.

Severity: 12 (severe error)

Related reference:

⮕ ADMIN_TASK_ADD stored procedure (DB2 Administration Guide)

⮕ ADMIN_TASK_REMOVE stored procedure (DB2 Administration Guide)

DSNA800I   **DSNACA00 TCB=***x* **ENTERED DSNACA00 ACTION=***x* **R1=***x* **CABPTR=***x* **CABFLAG1=***x* **CABFLAG2=***x*

Explanation: Messages in the DSNA800 series are intended for diagnosis and are documented in DB2 Diagnosis Guide and Reference under "Interpreting trace messages."

Related information:

⮕ DB2 Diagnosis Guide and Reference

Chapter 2. DSNB messages

Message identifiers that begin with DSNB communicate information about the buffer manager.

DSNB200I   UPDATE VVDS FAILED. DSNAME = *dsn*, **MMRC=***cccccccc* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id* **=***token*

Explanation: DB2 was unable to update the data set high-used RBA in the VSAM volume data set (VVDS) when it was ready to close a data set that was used by a RECOVER, LOAD, or REORG utility job. The error was detected by the media manager catalog update function that issues system message IEC161I. The *dsn* indicates the DSNAME of the data set. MMRC is the return code that DB2 received from the media manager catalog update service. Refer to *z/OS DFSMSdfp Diagnosis Reference* for a description of these codes.

The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

An LUW-ID of '*' indicates that the thread does not have an LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

System action: DB2 continues closing this data set. However, the utility job is informed with a 00C2010C 'resource unavailable' reason code. The utility job is left in a failed state.

Operator response: Notify the system programmer.

System programmer response: To determine the cause of the error, refer to system message IEC161I in the appropriate MVS publication. Correct the error, and then notify the user to reinvoke the application.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- IDCAMS LISTCAT output for the ICF (VSAM) catalog that contains the DB2 subsystem catalog that defined the failing index or table.
- IDCAMS LISTCAT output that shows the attributes of all active log data sets for the DB2 member that failed.

DSNB204I   OPEN OF DATA SET FAILED. DSNAME = *dsn*

Explanation: A required data set of a DB2 table or index space could not be opened. The error was detected by the media manager CONNECT function that issues system message IEC161I. The *dsn* indicates the DSNAME of the data set.

After being issued for a given data set, this message is disabled for that data set.

System action: Buffer manager functions that detect this condition terminate.

Operator response: Notify the system programmer.

System programmer response: To determine the cause of the error, refer to system message IEC161I in the appropriate MVS publication. Correct the error, and reinvoke the application.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IDCAMS LISTCAT output for the ICF (VSAM) catalog that contains the DB2 subsystem catalog that defined the failing index or table.
- VTOC listing of the disk volume on which the failing operation occurred.

.

DSNB207I   DYNAMIC ALLOCATION OF DATA SET FAILED. REASON = *rrrr* DSNAME = *dsn*

Explanation: A required data set of a DB2 table or index space could not be allocated. Reason code *rrrr*, which is returned by dynamic allocation, indicates the reason for the failure. Refer to the appropriate version of the *MVS Authorized Assembler Services Guide* (SA22-7608) publication for a description of this reason code. The *dsn* indicates the data set's DSNAME.

After being issued for a given data set, this message is disabled for that data set.

System action: Buffer manager functions that detect this condition terminate and pass a unique error return code.

Operator response: Notify the system programmer.

System programmer response: Correct the situation, and then reinvoke the application.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IDCAMS LISTCAT output for the ICF (VSAM) catalog that contains the DB2 subsystem catalog that defined the failing index or table.
- VTOC listing of the disk volume on which the failing operation occurred.

.

DSNB209I   *csect-name*-BUFFER MANAGER TABLESPACE/INDEXSPACE CLEAN-UP ROUTINE (DSNB1CFC)
    HAS BEEN SCHEDULED FOR DBNAME = *database-name*, SPACENAME = *space-name* ,
    INSTANCE*instance-number* , PSID = X'*psid*'

Explanation: DB2 detected an abnormal condition for one of the following reasons:

- The associated table space or index space contains modified pages that cannot be written to DASD for a period of time that is longer than expected. This situation can be caused by either an unusually long period of continuous updates or by an internal logic error.
- An abnormal termination occurred because of inconsistent data or an incomplete update to the page.

DB2 scheduled the table space or index space cleanup routine (DSNB1CFC) for the associated table space or index space. DSNB1CFC acquires a drain lock to ensure that no updates are in progress. Data is inaccessible to users during the cleanup process.

For recoverable inconsistent pages with unbalanced write intents, DB2 initiates the recovery process and reconstructs the pages using the DB2 log. For other modified pages with unbalanced write intents, the buffer manager forces the pages to be written back to DASD and then makes the buffer space available for reassignment.

csect-name
        The name of the control section that issued the message.

database-name
        The name of the database that is associated with the action.

space-name
        The name of the table space or index space that is associated with the action.

instance-number
        The instance number of the table space or index space.

psid   The internal identifier for the table space or index space. This value is used to associate this message with the follow-on message DSNB221I, which might be issued.

System action: DB2 is ready to clean up the associated table space or index space. However, before any cleanup is initiated, DB2 requests a drain on the table space or index space. DB2 waits indefinitely for this drain. Therefore, all activities against the table space or index space, or both, must be quiesced before the cleanup process can proceed. After the cleanup process completes, the table space or index space is automatically made accessible to users.

User response: If the cleanup activity does not seem to finish,DB2 might still be waiting for a drain on the table space or index space. Issue a -DISPLAY DATABASE CLAIMERS command for the table space or index space in

question to determine whether the cleanup routine is still waiting for a drain. If it is still waiting, identify the claimers of the table space or index space. If the jobs are holding claims for a very long duration (for example, they are waiting for input), you might need to cancel those jobs in order to allow the cleanup process to proceed.

Related reference:

⬈ -DISPLAY DATABASE (DB2) (DB2 Commands)

Related information:

"DSNB221I" on page 45

DSNB212I *csect-name*-RECOVER PAGE STARTING DBNAME = *database-name*, SPACENAME = *space-name*, INSTANCE = *instance-number*, PAGE NUMBER = X'*page-number*'

Explanation: The buffer manager subcomponent of DB2 discovered an inconsistent page with unbalanced write intents. DB2 is ready to do online recovery on the page by using the DB2 log.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database that is associated with the recovery.

space-name
> The name of the table space or index space that is associated with the recovery.

instance-number
> The instance number of the table space or index space.

page-number
> A hexadecimal number that identifies the page that is logically inconsistent.

This message is followed by message DSNB215I if the recovery was successful or by message DSNB217I if it was not successful.

System action: Processing continues.

User response: No action is required.

Related information:

"DSNB215I"

"DSNB217I" on page 44

DSNB215I *csect-name*-ONLINE RECOVERY FOR AN INCONSISTENT PAGE HAS BEEN COMPLETED FOR DBNAME = *database-name*, SPACENAME = *space-name*, INSTANCE *instance-number*, PAGE NUMBER = X'*page-number*'

Explanation: The recovery of an inconsistent page is complete.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database that is associated with the recovery.

space-name
> The name of the table space or index space that is associated with the recovery.

instance-number
> The instance number of the table space or index space.

page-number
> A hexadecimal number that identifies the page that is logically inconsistent.

System action: Processing continues.

User response: No action is required.

DSNB217I   *csect-name*-**ONLINE RECOVERY FOR AN INCONSISTENT PAGE WAS UNSUCCESSFUL FOR DBNAME =** *database-name* **, SPACENAME =** *space-name* **, INSTANCE=** *instance-number* **, PAGE NUMBER = X'***page-number***'**

Explanation: The recovery process for an inconsistent page failed. The page is written to disk as logically inconsistent. This message is issued in response to message DSNB212I, which was issued to indicate the start of the recovery process.

csect-name
>   The name of the control section that issued the message.

database-name
>   The name of the database that is associated with the recovery.

space-name
>   The name of the table space or index space that is associated with the recovery.

instance-number
>   The instance number of the table space or index space.

page-number
>   A hexadecimal number that identifies the page that is logically inconsistent.

System action: The page is marked as logically inconsistent and is unavailable until it is repaired. The execution unit driving this function proceeds normally.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

User response: If necessary, use the REPAIR utility to repair the page manually. If the problem persists, contact IBM Software Support.

Problem determination: Message DSNI012I might be issued before this message to indicate the same failure. If so, a 00C90102 abend dump might help determine the cause of this failure. Otherwise, see SYS1.LOGREC for any abend that occurred between the associated DSNB212I message and this message. Use the timestamp on the log records to find this abend.

Related reference:

⏎ REPAIR (DB2 Utilities)

Related information:

"DSNB212I" on page 43

"DSNI012I" on page 217

⏎ 00C90102 (DB2 Codes)

DSNB219I   *csect-name*-**BUFFER MANAGER TABLESPACE/INDEXSPACE CLEAN-UP PROCESS HAS BEEN COMPLETED FOR DBNAME =** *database-name***, SPACENAME =** *space-name***, INSTANCE** *instance-number*

Explanation: The buffer manager table space or index space cleanup process has completed for the associated table space or index space. Message DSNB209I, if present, provides additional information about why the buffer manager scheduled the table space or index space cleanup process.

csect-name
>   The name of the control section that issued the message.

database-name
>   The name of the database that is associated with the action.

space-name
>   The name of the table space or index space that is associated with the action.

instance-number
>   The instance number of the table space or index space.

System action: Processing continues.

User response: No action is required.

Related information:

"DSNB209I" on page 42

DSNB221I *csect-name*-**BUFFER MANAGER TABLESPACE/INDEXSPACE CLEAN-UP PROCESS WAS NOT REQUIRED FOR PSID = X'***psid***'**

Explanation: This message indicates that the buffer manager (BM) table space or index space cleanup process was bypassed for the associated table space or index space. After quiescing all other activities against the table space or index space, the buffer manager discovered that the table space or index space is no longer in active mode; that is, it was closed.

The abnormal condition detected earlier was resolved by buffer manager close page set processing. Refer to the corresponding message, DSNB209I, using *psid* as the matching field, for the reasons the buffer manager schedules the table space or index space cleanup process. The *psid* is the internal identifier for the associated table space or index space.

System action: Processing continues.

DSNB223I *csect-name*-**BUFFER MANAGER TABLESPACE/INDEXSPACE CLEANUP PROCESS WAS SKIPPED FOR PSID = X'***psid***'**

Explanation: This message indicates that the table space or index space cleanup process was bypassed for the associated table space or index space. During the process of quiescing all other activities against the table space or index space, an error that is not recoverable was detected. The error might be caused by any of the following conditions:

- An out-of-storage condition was detected by IRLM.
- The table space or index space was dropped because of a backout.

It is assumed that the abnormal condition detected earlier can later be resolved by the close page set processing. Refer to the corresponding message DSNB209I (use *psid* as the matching field) for the reasons for scheduling the table space or index space cleanup process. The *psid* specifies the internal identifier for the associated table space or index space.

System action: Processing continues.

System programmer response: The associated table space or index space might contain inconsistent data. Issue the DB2 STOP TABLESPACE or INDEXSPACE command to allow close page set processing to perform cleanup for the associated table space or index space.

DSNB224I *csect-name*- **BUFFER MANAGER I/O ERROR DURING** *function*, **DSNAME =** *data-set-name* , **PAGE NUMBER=X'***page-number***', MM ERROR FLAG=X'***aa***', CSW STATUS=X'***bbbb***', SENSE BYTES=X'***cccc***', SEEK ADDRESS=X'***dddddddddddddddd***'**

Explanation: An I/O error was encountered by the buffer manager (BM) subcomponent of DB2.

csect-name
    The name of the control section that issued the message.

function The function that was being performed when the error occurred. Possible values are READ, WRITE, or FORMAT.

data-set-name
    The name of the data set on which the I/O operation was being performed.

page-number
    The page number, in hexadecimal format, of the page on which the I/O operation was being performed.

aa, *bbbb*, *cccc*, **and** *dddddddddddddddd*
    Error information that is returned by the media manager and used for error analysis by IBM Software Support.

System action: The system action depends on the value of *function*. In most cases, DB2 returns a message that

DSNB225I

indicates that a resource is unavailable with a reason code that identifies the problem. However, in the following cases, the I/O-error page range of the data set is updated and message DSNU086I is issued:

- *function* is WRITE
- *function* is READ and the I/O operation was performed during a restart or rollback operation.

After being issued 10 times for a given data set, this message is disabled for that data set until the data set is closed and reopened.

Operator response: Notify the system programmer.

System programmer response: See the information for the reason code that is returned in the SQLCA or in the utility messages and take the specified action.

If you suspect an error in DB2, see the information about collecting data.

Problem determination: See the information about problem determination for the reason code that is returned in the SQLCA or in the utility messages.

Check the MVS console log for an IOS (I/O subsystem) error message that indicates the type of I/O error that occurred. If an IOS message was issued, follow the problem determination directions for that message.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

.

Related concepts:

⏏ DB2 reason codes (DB2 Codes)

Related tasks:

⏏ Collecting data (Collecting data)

Related information:

"DSNU086I" on page 617

DSNB225I   *csect-name*- **BUFFER MANAGER I/O ERROR DURING** *function*, **DSNAME =** *data-set-name* , **MM ERROR CODES=X'**_aaaaaaaa_**' DB2 REASON CODE=X'**_reason-code_**'**

Explanation: DB2 encountered an I/O error.

csect-name
      The name of the control section that issued the message.

function The function that was being performed when the error occurred. Possible values are READ, WRITE, CASTOUT, or FORMAT.

data-set-name
      The name of the data set on which the I/O operation was being performed.

aaaaaaaa
      Error information that is returned by the media manager at the time of the failure. This information used for error analysis by IBM Software Support.

reason-code
      A numeric value that indicates the reason for the failure.

System action: Except for a castout error, the application terminates with abend X'04E' and reason code 00C200A4 or 00C200C0. If an error occurs during a castout operation, processing continues.

Operator response: Notify the system programmer.

See the information for the given reason code and take the specified action.

System programmer response: If you suspect an error in DB2, see the information about collecting data.

Problem determination: Check the MVS console log for an IOS (I/O subsystem) error message that indicates the

type of I/O error that occurred. If an IOS message was issued, follow the problem determination directions for that message.

Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IDCAMS LISTCAT output for the ICF (VSAM) catalog that contains the DB2 subsystem catalog that defined the failing index or table.
- VTOC listing of the disk volume on which the failing operation occurred.

.

Related concepts:

➡ DB2 reason codes (DB2 Codes)

Related tasks:

➡ Collecting data (Collecting data)

Related information:

➡ 00C200A4 (DB2 Codes)

➡ 00C200C0 (DB2 Codes)

DSNB226I *csect-name* **- BUFFER MANAGER PAGE GET ROUTINE DETECTED AN INVALID PAGE IN DBNAME=***database-name***, SPACENAME=***space-name* **, INSTANCE=***instance-number***, DATA SET NUMBER=** *data-set-number***, PAGE EXPECTED=X'***expected page-number***', PAGE NUMBER FOUND=X'***page-number***'**

Explanation: An unexpected page was received during buffer manager (BM) get page processing.

csect-name
    The name of the control section that issued the message.

database-name
    The name of the database that contains the involved page set.

space-name
    The name of the table space or index space for the involved page set.

instance-number
    The instance number of the table space or index space.

data-set-number
    The data set number for the involved page set.

expected page-number
    The page number of the page in error. The buffer manager expected to find this page in the buffer.

page-number
    The page number of the page that was found in the buffer instead of the expected page.

System action: The application is terminated with abend X'04E' and reason code 00C200F7.

Operator response: Notify the system programmer.

See the information for the given reason code and take the specified action.

System programmer response: If you suspect an error in DB2, see the information about collecting data.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).

- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IDCAMS LISTCAT output for the ICF (VSAM) catalog that contains the DB2 subsystem catalog that defined the failing index or table.
- VTOC listing of the disk volume on which the failing operation occurred.
- DSN1COPY-generated copy of the table and index, taken before any recovery is performed.

.

Related concepts:

➡ DB2 reason codes (DB2 Codes)

Related tasks:

➡ Collecting data (Collecting data)

Related information:

➡ 00C200F7 (DB2 Codes)

DSNB227I   *csect-name* **- DFHSM RECALL FAILED FOR DSNAME=**_dataset-name_ **HSMRC=**_return-code_ **HSMRSN=**_reason-code_

Explanation: The attempt to recall a migrated data set failed.

csect-name
>The name of the control section that issued the message.

dataset-name
>The name of the data set.

return-code
>The return code issued by DFSMShsm. This value corresponds to a similarly numbered MVS ARC message:
>
>ARC11 + *return-code* + I

reason-code
>The reason code issued by DFSMShsm.

System action: A "resource not available" code is returned.

Operator response: Notify the system programmer.

System programmer response: Use the information provided in message DSNB227I to determine the reason for the DFSMShsm recall failure.

Related concepts:

➡ RECALL command (z/OS DFSMShsm Storage Administration Reference)

Related reference:

➡ ARC messages (MVS System Messages)

Related information:

➡ Return codes from DFSMShsm commands (DFSMShsm Managing Your Own Data)

DSNB228I   *csect-name* **GROUP BUFFER POOL** *gbpname* **CANNOT BE ACCESSED FOR** *function* **MVS IXLCACHE REASON CODE=**_reason_

Explanation: While accessing group buffer pool *gbpname* for function *function* DB2 received an error from the MVS IXLCACHE request.

function READ or WRITE

reason   Reason code from the MVS IXLCACHE request

System action: The system action depends on the function being performed. If the function is a WRITE, or a READ performed during rollback or DB2 restart, the page is added to the logical page list (LPL) for the data set and message DSNB250E is issued. Otherwise, 'resource unavailable' is returned to the user with a reason code that identifies the specific problem.

System programmer response: See *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the MVS IXLCACHE reason code.

DSNB229I   *csect-name* **UNEXPECTED ERROR WHEN CHECKING FOR REQUEST COMPLETION FOR GROUP BUFFER POOL** *gbp-name* **MVS IXLFCOMP REASON CODE=***reason*

Explanation: DB2 received an unexpected error from MVS when checking for the completion of a write request to the secondary group buffer pool for duplexing. *gbp-name* indicates the name of the group buffer pool. If the group buffer pool is duplexed and the error was encountered on a request to the secondary group buffer pool, then *gbp-name* indicates the name of the group buffer pool suffixed with '-SEC'.

reason is the reason code from the MVS IXLFCOMP service.

System action: DB2 returns the group buffer pool back to simplex mode. The secondary group buffer pool is deallocated.

System programmer response: See *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the MVS IXLFCOMP reason code.

DSNB232I   *csect-name* **- UNEXPECTED DATA SET LEVEL ID ENCOUNTERED - LEVEL ID =***lvlid* **EXPECTED LEVEL ID=***xlvlid* **DSN=***dsn*

Explanation: A required data set of a DB2 table space could not be opened. DB2 encountered an unexpected data set level ID.

lvlid   The unexpected level ID from the data set. If DB2 was unable to retrieve the level ID from the data set (for example, if the data set had been deleted and redefined), the level ID is shown as "X''".

xlvlid   The level ID recorded by DB2 previously and is the expected level ID for opening the data set

dsn   The DSNAME of the data set.

This problem could be the result of data sets that were restored to a level that is inconsistent with other data sets, inconsistent with the DB2 log, or both.

System action: DB2 sends back a 'resource unavailable' SQL code to the application and a reason code explaining the error.

Operator response: Notify the system programmer.

System programmer response: To enable access to a data set reported as down level, take one of the following actions:
- Restore the page set or partition to the proper level (use DSN1COPY, DFHSM, or another tool).

  Identify the level ID of a copy of the page set or partition by running the DSN1PRNT(FORMAT) utility. The level ID is found in the header page of the page set or partition.
- Recover the down level page set or partition to currency by running the RECOVER or RECOVER LOGONLY utility.
- Accept the down levelling by running the REPAIR utility with the LEVELID option.

  This option is only valid if DB2 is able to access the level ID (*lvlid* is not "X''").

  Inconsistent data problems might result from this action and are the user's responsibility.
- Replace the contents of the page set or partition using the LOAD REPLACE utility.
- Explicitly restore the page set to a prior point in time with the RECOVER utility.

If the expected level ID is less than the encountered level ID, it is possible that the object identified in the message was ported from another system improperly or it might indicate that DSNDB01.SYSLGRNX is down level. If the problem is in the object identified in the message, the actions listed above can be used to allow access to the data. If the problem is in SYSLGRNX, this is a serious situation that compromises the recoverability of all objects managed by DB2. Investigate whether SYSLGRNX might have inadvertently been restored to a point in time inconsistent with the rest of the DB2 system.

DSNB235A *csect-name* **- GRS RESOURCE SHORTAGE DETECTED BY DB2**

Explanation: A z/OS Global Resource Serialization (GRS) resource shortage was detected when DB2 opened or closed a data set.

csect-name
> The name of the control section that issued the message.

System action: To relieve the GRS resource shortage, DB2 might close data sets that have not been used recently.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Message DSNB235A is deleted when DB2 detects that the GRS resource shortage has been relieved.

System programmer response: Work with your DB2 administrator to reduce the DB2 workload, or reduce the number of open data sets.

Problem determination: GRS message ISG353E might be displayed when the resource shortage occurs. This message identifies the specific storage area that has a shortage.

GRS message ISG354I displays when the resource shortage has been relieved.

Related reference:

⮕ Message ISG353E (MVS System Messages)

⮕ Message ISG354I (MVS System Messages)

DSNB250E *csect-name* **A PAGE RANGE WAS ADDED TO THE LOGICAL PAGE LIST. DATABASE NAME=***database-name***, SPACE NAME=***space-name***, INSTANCE=***instance-number***, DATA SET NUMBER=***data-set-number***, PAGE RANGE X'***low-page***' TO X'***high-page***', START LRSN=X'***start-lrsn***', END LRSN=X'***end-lrsn***', START RBA=X'***start-rba***', LPL TRACE ID=***trace-id***, LPL REASON TYPE=***reason-type***

Explanation: DB2 added a page range to the logical page list (LPL) for the specified data set, because the pages in the page range had a problem.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

space-name
> The name of the table space or index space that is associated with the action.

instance-number
> The instance number of the table space or index space.

data-set-number
> The data set number.

low-page **and** *high-page*
> The range of pages that was added to the LPL. The values are hexadecimal page numbers.

start-lrsn **and** *end-lrsn*
> The log range that is needed to recover the pages in the page range. These values are RBA values unless DB2 is in a data sharing environment. In that case, these values are LRSN values. A zero value for *end-lrsn* refers to the current end of the recovery log.

start-rba
> In a data sharing environment, the starting RBA value of the recovery log range if the value is not zero. The *start-rba* value applies to log records for only the DB2 member that issued the message.

trace-id A value that is used for serviceability by IBM Software Support.

reason-type
> The reason why the pages were added to the LPL. *reason-type* can contain one of the following values:

> **DASD** DB2 encountered a DASD I/O error when trying to read or write pages on DASD.

LOGAPPLY
> DB2 cannot apply log records to the pages.

GBP DB2 cannot successfully read from or externalize the pages to the group buffer pool. This problem occurred for one of the following reasons:
> - a link or structure failure
> - the group buffer pool is in rebuild status
> - the group buffer pool was disconnected

LOCK DB2 cannot get the required page latch or page P-lock on the pages.

CASTOUT
> The DB2 castout processor cannot successfully cast out the pages.

NOTLOGGD
> Work on an object needs to be undone. However, the work was not logged, because the object has the NOT LOGGED attribute.

System action: The pages are added to the LPL

For any *reason-type* value other than NOTLOGGD, DB2 initiates the automatic LPL recovery. The pages remain inaccessible until they are recovered.

If you attempt to access a page in the LPL, DB2 returns SQLCODE -904 with reason code 00C20206.

If MVS detects a problem when accessing the group buffer pool, DB2 writes a SYS1.LOGREC entry that describes the error.

System programmer response: Take the appropriate actions for the *reason-type* value based on the console message.

To recover pages from the LPL, perform one of the following actions:
- For a *reason-type* value other than NOTLOGGD, issue the START DATABASE command with the SPACENAM option.
- Run the RECOVER utility
- Run the LOAD utility

Related reference:

-START DATABASE (DB2) (DB2 Commands)

RECOVER (DB2 Utilities)

LOAD (DB2 Utilities)

Related information:

-904 (DB2 Codes)

00C20206 (DB2 Codes)

DSNB260I *csect-name* **WARNING - A READER HAS BEEN RUNNING FOR** *duration* **MINUTES - CORRELATION NAME=**correlation-name**, CONNECTION ID=**connection-id**, LUWID=**luwid **ID=**token**, PLAN NAME=**plan-name**, AUTHID=**authorization-id**, END USER ID=**user-id**, TRANSACTION NAME=**transaction-name**, WORKSTATION NAME=**system-name

Explanation: A long-running reader has reached the maximum permitted time without issuing either a COMMIT or ROLLBACK statement.

csect-name
> The name of the control section that issued the message.

duration
> The time, in minutes, that has elapsed since the beginning of the unit of work

correlation-name
> The correlation name that is associated with the unit of work.

connection-id
> The connection ID that is associated with the unit of work.

DSNB301E

luwid   The logical unit of work ID, or an asterisk (*) to denote that the thread originated at the local site.

token   A unique token number that is associated with the logical unit of work ID.

plan-name
        The plan that is associated with the unit of work.

authorization-id
        The authorization ID that is associated with the unit of work.

user-id   The user ID that is associated with the unit of work, if known. Otherwise, an asterisk (*).

transaction-name
        The transaction that is associated with the unit of work, if known. Otherwise, an asterisk (*).

system-name
        The workstation host name that is associated with the unit of work, if known. Otherwise, an asterisk (*).

The amount of time that is permitted for a reader is controlled by the LONG-RUNNING READER value on panel DSNTIPE.

System action: Processing continues.

If IFCID 0313 trace is activated, an IFCID 0313 trace record is written. This message is issued every time that the threshold is reached.

System programmer response: Work with the application programmer to determine if there is a problem with the unit of work. If an application program caused the unit of work to reach its time limit threshold, you can use CANCEL THREAD to cancel the associated thread.

Administrator response: Increase the LONG-RUNNING READER value on panel DSNTIPE so that this message is issued less frequently.

Programmer response: Make sure that the application commits at appropriate intervals, or work with the DB2 administrator to determine an appropriate frequency for the COMMIT check.

Related concepts:

➡ Commit and rollback of transactions (Introduction to DB2 for z/OS)

Related reference:

➡ LONG-RUNNING READER field (LRDRTHLD subsystem parameter) (DB2 Installation and Migration)

➡ -CANCEL THREAD (DB2) (DB2 Commands)

➡ COMMIT (DB2 SQL)

➡ ROLLBACK (DB2 SQL)

DSNB301E   *csect-name* **GROUP BUFFER POOL** *gbp-name* **CANNOT BE CONNECTED DB2 REASON CODE =** *reason-code* **MVS IXLCONN REASON CODE =** *ixlconn-reason*

Explanation: DB2 was not able to connect to the named group buffer pool. This message is generally issued once for each DB2 subsystem for each DB2 reason code, except 00C20204, until the failure condition is corrected.

csect-name
        The name of the control section that issued the message.

gbp-name
        The name of the group buffer pool.

reason-code
        The DB2 reason code.

        For DB2 reason code 00C20204, which indicates a problem during MVS IXLCONN processing, the DSNB301E message is issued once for each DB2 subsystem for each IXLCONN reason code until the failure condition is corrected.

ixlconn-reason
>When the DB2 reason code is 00C20204, the MVS IXLCONN reason code. For all other DB2 reason code values, *ixlconn-reason* contains zeros.

System action: If the connection was requested on behalf of an access request by an SQL user on this DB2, SQLCODE -904 (resource unavailable) is returned to the SQL user with the DB2 reason code shown in this message.

If the connection was requested on behalf of an SQL user on another DB2 in the data sharing group, SQLCODE -904 (resource unavailable) is returned to the SQL user with DB2 reason code 00C20252. This indicates that the requested table space or index space is unavailable because another DB2 cannot connect to the group buffer pool.

If the MVS IXLCONN reason code is *xxxx*0C08 (structure allocation failed because there was no suitable facility to allocate based on the preference list in the MVS CFRM administrative policy), or if the MVS IXLCONN reason code is *xxxx*084C (user does not have proper Security Access Facility (SAF) authorization), then IFCID 0250, if active, contains additional diagnostic information.

System programmer response: See the specified DB2 reason code for information about appropriate actions.

Related reference:

Return and Reason Codes

Related information:

-904 (DB2 Codes)

00C20204 (DB2 Codes)

00C20252 (DB2 Codes)

00C202A8 (DB2 Codes)

DSNB302I   *csect-name* **GROUP BUFFER POOL** *gbpname* **IS ALLOCATED IN A VOLATILE STRUCTURE**

Explanation: The DB2 group buffer pool structure is volatile.

One DB2 member issues this message whenever the structure has the volatile attribute. DB2 detects this either on the first connection to the group buffer pool or when the attribute changes later. DB2 always requests nonvolatile group buffer pools but allows allocation in a volatile structure if that is all that is available.

System action: DB2 processing continues.

System programmer response: To cause the group buffer pools to be allocated in nonvolatile coupling facilities, ensure that:

- At least a subset of your coupling facilities are configured correctly for nonvolatility.
- The MVS CFRM policy correctly specifies the nonvolatile coupling facilities in the preference list for the group buffer pool coupling facility structures.

Even when these things are correctly done, there might be instances when a suitable coupling facility in the preference list cannot be found (for example, not enough free space), and therefore allocation occurs in a different coupling facility. MVS tries to allocate the structure in the facility that best meets the structure allocation request as follows:

1st choice:
>Coupling facility that meets the volatility requirement, the space requirement, and the exclusion list requirement.

2nd choice:
>Coupling facility that meets the volatility requirement and the storage requirement.

3rd choice:
>Coupling facility that has the best fit (most available free space) even though this might be less than the amount requested.

4th choice:
>Structure cannot be allocated.

IFCID 0250, if activated, contains a list of the coupling facilities on which allocation was attempted.

DSNB303E *csect-name* **A LOSS OF CONNECTIVITY WAS DETECTED TO GROUP BUFFER POOL** *gpbname*

Explanation: Connectivity to group buffer pool *gpbname* was lost.

System action: DB2 disconnects from the coupling facility structure with REASON=FAILURE.

The action taken in response to this problem depends on the attempted function, as follows:

- For a coupling facility read: SQLCODE -904 (resource unavailable) is returned to the SQL user with a reason code indicating the specific error.
- For a coupling facility read during rollback, backout, or restart: The inaccessible page is added to the logical page list (LPL) for the page set. Message DSNB250E is issued for the first page of each data set added to the LPL.
- For a coupling facility read during castout: The castout operation cannot complete.
- For a coupling facility write: The changed page is added to the LPL for the page set.

DSNB304I *csect-name* **GROUP BUFFER POOL** *gbpname* **WAS SET TO 'DAMAGE ASSESSMENT PENDING' STATUS**

Explanation: A coupling facility structure for a group buffer pool was damaged.

System action:

- DB2 puts the group buffer pool into 'damage assessment pending' (DAP) status. Message DSNB304I indicates the DAP status. The group buffer pool cannot be accessed until DAP status is cleared.
- The DB2 damage assessment process determines which table spaces and index spaces are using the failed group buffer pool and puts those spaces into 'group buffer pool recovery pending' (GRECP) status.
- After DB2 sets all the necessary spaces to GRECP status, the DAP status for the group buffer pool is cleared. Message DSNB305I indicates that DAP status is cleared.

DSNB305I *csect-name* **'DAMAGE ASSESSMENT PENDING' STATUS WAS CLEARED FOR GROUP BUFFER POOL** *gbpname*

Explanation: Damage assessment completed and the 'damage assessment pending' (DAP) status is cleared for the group buffer pool (GBP). At this point, the group buffer pool status is no longer restricted and DB2 attempts to reconnect to the group buffer pool on the next reference to a GBP-dependent page set within the associated buffer pool.

Message DSNB304I precedes this message.

DSNB306I *csect-name* **THE REQUESTED LENGTH OF THE LOCAL CACHE VECTOR COULD NOT BE ALLOCATED FOR GROUP BUFFER POOL** *gbpname* **REQUESTED LENGTH =** *length1* **ALLOCATED LENGTH =** *length2*

Explanation: DB2 connected to the group buffer pool indicated by *gbpname*, but the MVS Cross-system Extended Services (XES) component could not allocate the requested size of the local cache vector (LCV) in the hardware system area (HSA). DB2 requires an LCV size that is large enough to handle all pages in the member's buffer pool.

length1 indicates the LCV size (in number of bits) that DB2 requested. *length2* indicates the LCV size (in number of bits) that XES was able to allocate.

To prevent flooding the operator console, this message is only issued once per DB2 member per group buffer pool until the condition is corrected.

System action: DB2 disconnects from the GBP and issues message DSNB301E with reason code 00C20208, if it has not yet been issued.

SQLCODE -904 is returned to the user with a reason code of 00C20208.

System programmer response: There is not enough HSA storage on the processor to support the size of the local cache vector. The details of HSA allocation are machine-dependent and are defined by the hardware. To correct this problem, take one or both of the following actions:

- Consult the documentation for your processor hardware to determine if the HSA storage allocation can be increased, and if so, how to do it.

- Modify the software configuration in your parallel sysplex to reduce the number or size of the vectors that are in use by other coupling facility structure users. Consider the following possible solutions:
  - Use the DB2 ALTER BUFFERPOOL command to reduce the size of this local buffer pool. This means that you might have to reduce the size of the virtual pool.
  - Use the DB2 ALTER BUFFERPOOL command to reduce the size of another local buffer pool that is already connected to a group buffer pool. This means that you might have to reduce the size of the virtual pool. If you make a connected buffer pool smaller, DB2 can usually contract the size of the corresponding LCV dynamically, making HSA storage available. However, DB2 cannot always do the dynamic LCV contraction. In cases where DB2 cannot dynamically contract the LCV size, you can use the DB2 ALTER BUFFERPOOL command to either force a deallocation of the LCV or to force a contraction of the LCV as follows:
    1. Redefine the virtual pool to be smaller than the original size.
    2. Take one of the following actions:
       a. Make the virtual pool very small (perhaps 50 pages or less). Making the virtual pool sufficiently small should let DB2 dynamically contract the LCV, or
       b. Delete the virtual pool to let DB2 disconnect from the group buffer pool. This frees the entire LCV associated with that member's connection to that group buffer pool.
  - Consolidate your group buffer pool-dependent page sets into fewer buffer pools to reduce the number of group buffer pool connections.
  - Consult the documentation for the other software components that might be using a coupling facility cache or list structure with an associated LCV to determine whether it is possible to either delete the connection or to reduce the size of the LCV associated with the connection.

DSNB307I   *csect-name* **GROUP BUFFER POOL** *gbpname* **WAS SUCCESSFULLY ALLOCATED BUT THE EXCLUSION LIST WAS NOT HONORED**

Explanation: DB2 successfully connected to the group buffer pool and allocated the group buffer pool in a coupling facility. However, the group buffer pool was allocated in a coupling facility containing a structure in the exclusion list.

System programmer response: If you require that the exclusion list be honored, you must ensure that there is sufficient space in the coupling facilities that are not in the exclusion list. IFCID 0250 provides information about which facilities were attempted.

DSNB308I   *csect-name* **GROUP BUFFER POOL** *gbpname* **WAS SUCCESSFULLY ALLOCATED BUT THE ALLOCATED SIZE OF** *alloc_size* **IS LESS THAN THE REQUESTED SIZE OF** *req_size*

Explanation: DB2 successfully connected to the group buffer pool and allocated the group buffer pool in a coupling facility. However, the allocated size of the group buffer pool was less than the requested size in the MVS CFRM administrative policy because there is not enough space in the preference list facilities. Both *alloc_size* and *req_size* are in 4KB blocks.

System programmer response: IFCID 0250 provides information about which coupling facilities were used for attempted allocation.

DSNB309I   *csect-name* **GROUP BUFFER POOL** *gbpname* **HAS BEEN DISCONNECTED WITH A REASON OF '***reason***'**

Explanation: DB2 disconnected from the named group buffer pool.

reason indicates one of the following values:

NORMAL
     DB2 disconnected the group buffer pool normally. MVS deletes the connection.

FAILURE
     DB2 disconnected the group buffer pool abnormally. MVS puts the connection into a 'failed-persistent' state. DB2 disconnects the GBP with a reason of FAILURE when some type of error has been detected, and castout processing cannot proceed. Usually the error involves a connectivity failure to either the coupling facility or DASD. This message is preceded by message DSNB303E, and probably a series of DSNB311I and DSNB312I messages.

If this message is issued because connectivity was lost to the group buffer pool, this message signifies that this DB2 member has completed its processing of the error.

System programmer response: See the system programmer response for message DSNB311I or DSNB312I.

DSNB310I *csect-name* **GROUP BUFFER POOL** *gbpname* **IS ALLOCATED IN A NONVOLATILE STRUCTURE**

Explanation: The coupling facility storage in which the group buffer pool is allocated was previously volatile, but is now nonvolatile. For availability reasons, we recommend that DB2 group buffer pools be allocated in nonvolatile coupling facility storage. When DB2 connects a group buffer pool, it requests that the backing coupling facility structure be allocated in nonvolatile storage. If the request for nonvolatile storage is not honored, DB2 issues message DSNB302I.

System action: DB2 processing continues.

DSNB311I *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE** *instance-number* **HAS PAGES IN THE LOGICAL PAGE LIST**

Explanation: This message is issued after a loss of connectivity to a group buffer pool is detected. The named table space or index space has one or more pages in the logical page list (LPL) that need to be recovered. This message is issued after DB2 has completed its processing for the table space or index space; this point is the earliest point at which you can recover LPL pages for that table space or index.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database that is associated with the recovery.

space-name
> The name of the table space or index space that is associated with the recovery.

instance-number
> The instance number of the table space or index space.

System action: The pages in the LPL are not accessible until they have been recovered.

System programmer response: Issue the START DATABASE command with the named table space or index to recover the LPL pages. You must issue the START DATABASE command from a DB2 member that can connect to the group buffer pool. Alternatively, you can use the RECOVER utility or the LOAD utility to recover the LPL pages.

Related reference:

▶ -START DATABASE (DB2) (DB2 Commands)

▶ RECOVER (DB2 Utilities)

▶ LOAD (DB2 Utilities)

DSNB312I *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE** *instance-number* **PARTITION** *partition-number* **HAS PAGES IN THE LOGICAL PAGE LIST**

Explanation: This message is issued after a loss of connectivity to a group buffer pool is detected. A partition of the named table space or index has one or more pages in the logical page list (LPL) that need to be recovered. This message is issued after DB2 has completed its processing for the partition; this point is the earliest point at which you can recover the LPL pages for that partition.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database that is associated with the recovery.

space-name
> The name of the table space or index space that is associated with the recovery.

instance-number
> The instance number of the table space or index space.

partition-number
     The partition number.

System action: The pages in the LPL are not accessible until they have been recovered.

System programmer response: Issue the START DATABASE command on the table space or index space partition to recover the LPL pages. You must issue the START DATABASE command from a DB2 member that can connect to the group buffer pool. Alternatively, you can use the RECOVER utility or the LOAD utility to recover the LPL pages.

Related reference:

➡ -START DATABASE (DB2) (DB2 Commands)

➡ RECOVER (DB2 Utilities)

➡ LOAD (DB2 Utilities)

DSNB313I   *csect-name* **GROUP BUFFER POOL** *gpbname* **TO BE DISCONNECTED REASON=***reason* **LOSSCONN PERCENTAGE=***percentage*

Explanation: This DB2 member is disconnecting from group buffer pool *gpbname* because it has lost connectivity to the coupling facility in which the group buffer pool is allocated.

reason is one of the following:

NOSFMPOL
     MVS cannot determine the scope of the loss of connectivity because there is no active SFM policy with CONNFAIL(YES).

LOSSCONN
     MVS determined that the scope of the loss of connectivity was less than 100%.

NOXESEVL
     MVS started to try to determine the scope of the loss of connectivity, but some condition occurred which prevented MVS from being able to complete the evaluation. One example of such a condition would be that the SFM policy became inactive while MVS was trying to do the evaluation.

If *reason* is LOSSCONN, then *percentage* indicates the percentage of lost connectivity that was calculated by MVS. MVS evaluates the scope of the loss of connectivity in terms of SFM weights associated with the systems which have lost connectivity and which have active connectors to the coupling facility structure (if there are systems that do not have an active connector to the coupling facility structure, then those systems are not factored into the "percentage loss of connectivity" calculation).

If *reason* is not LOSSCONN, then *percentage* is not applicable (it should appear as zero).

System action: The DB2 member disconnects the group buffer pool, putting its connection into a "failed-persistent" state.

If there are transactions that are in progress at the time of the connectivity failure, DB2 might need to add pages to the logical page list (LPL).

System programmer response: Use the START DATABASE command to recover the LPL pages. If DSNDB01 or DSNDB06 objects have LPL pages, you might need to recover these first, before proceeding with LPL recovery for other objects.

DSNB314I   *csect-name* **DAMAGE ASSESSMENT TO BE TRIGGERED FOR GROUP BUFFER POOL** *gpbname* **REASON=***reason*

Explanation: DB2 is about to initiate the damage assessment function for group buffer pool *gpbname* because of a detected error in either the coupling facility in which the group buffer pool was allocated, or in the connectivity from the MVS systems to the coupling facility.

reason is one of the following:

100%LCON
     An error has occurred in a coupling facility or in the attachments of MVS systems to a coupling facility which has caused all DB2 members to lose connectivity to the named group buffer pool.

STRFAIL
> There was an error in the coupling facility in which the group buffer pool was allocated which caused a failure of the coupling facility structure.

CONERROR
> The named group buffer pool is currently allocated in a coupling facility, but DB2 cannot connect to the group buffer pool because the system on which it is running does not currently have connectivity, and MVS has indicated that currently no other member is actively connected and has connectivity.

DAP   The named group buffer pool is in damage assessment pending (DAP) state.

System action: DB2 triggers the damage assessment function to ensure that any changed pages that might have been cached in the affected group buffer pool get marked as "needing recovery" (the objects to which the pages belong are marked in GRECP state). This message should be immediately followed by a DSNB304I message indicating that the group buffer pool has been entered into damage assessment pending (DAP) state.

If there are transactions that are in progress at the time of the connectivity failure, DB2 might need to add pages to the logical page list (LPL).

There is one case where DB2 triggers damage assessment for a group buffer pool, but no DSNB314I message is issued. This case is where the group buffer pool was manually deleted (via MVS SETXCF FORCE commands) while all connections to the group buffer pool were failed-persistent (e.g. all DB2 members were down). In this case, the next DB2 member to connect to the group buffer pool will cause MVS to try to reallocate a new instance of the group buffer pool in some coupling facility in the CFRM policy preference list. If the allocation succeeds, then DB2, by doing a coupling facility structure version id comparison, will detect that the previous instance of the group buffer pool had been lost, and will trigger damage assessment. But you will not receive the DSNB314I message in this case, nor will you receive the DSNB304I or DSNB305I messages indicating that the group buffer pool has been put into damage assessment pending (DAP) state. You will only receive a series of DSNB322I and/or DSNB323I messages indicating those pagesets or partitions that have been put into GRECP state by the damage assessment process.

System programmer response: Use the START DATABASE command to recover the LPL pages and GRECP objects. If DSNDB01 or DSNDB06 objects are in GRECP state or have LPL pages, you might need to recover these first, before proceeding with GRECP or LPL recovery for other objects.

DSNB315I   *csect-name* **GROUP BUFFER POOL** *gpbname* **IS ALLOCATED AS GBPCACHE** *cacheatt*

Explanation:

YES   The group buffer pool will be used for data caching.

NO   The group buffer pool will not be used for data caching. It will be used for cross invalidation only.

One DB2 member issues this message whenever the structure is allocated. DB2 detects this on the first connection to the group buffer pool either in the mainline processing or during group buffer pool rebuild.

System action: DB2 processing continues.

DSNB316I   *csect-name* **GROUP BUFFER POOL** *gbp-name* **DELETED BECAUSE OF SUBSYSTEM PARAMETER SETTING**

Explanation: The specified group buffer pool was deleted during DB2 restart processing.

csect-name
> The name of the control section that issued the message.

gbp-name
> The name of the group buffer pool that was deleted.

The group buffer pool is deleted when there are no other DB2 subsystems connected to it, if the DEL_CFSTRUCT_ON_RESTART subsystem parameter indicates that DB2 should attempt to delete all group buffer pools.

System action: The group buffer pool named in the message is deleted.

User response: No action is required.

Related reference:

➡️ DEL CF STRUCTS field (DEL_CFSTRUCTS_ON_RESTART subsystem parameter) (DB2 Installation and Migration)

DSNB319A   *csect-name* **THERE IS A SHORTAGE OF SPACE IN GROUP BUFFER POOL** *gbp-name*

Explanation: The amount of storage in the group buffer pool (GBP) coupling facility structure available for writing new pages is low. The GBP is 75 percent full with changed data. As the GBP fills with changed data, the castout thresholds are continuously reached, and the castout engines must continuously work to attempt to keep the number of changed pages in the GBP below the castout thresholds. Also, the more changed data the GBP contains, the less efficiently it operates. Completely running out of GBP storage can result in transaction delays and in pages being added to the logical page list (LPL).

System action: DB2 processing continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

This message might not be automatically deleted in a pre SP5.2.1-JES3 environment.

Operator response: Contact the system programmer to fix the storage problem.

System programmer response: You can relieve the GBP storage shortage by taking one or more of the following actions:
- Use the ALTER GROUPBUFFERPOOL command to lower the CLASST and GBPOOLT castout thresholds.
- Use the ALTER GROUPBUFFERPOOL command to lower the GBPCHKPT GBP checkpoint frequency.
- Use the MVS SETXCF START,REBUILD command to expand the GBP coupling facility structure. If you rebuild, you can use the ALTER GROUPBUFFERPOOL command to lower the RATIO so that a higher percentage of the GBP storage is given to data entries over directory entries.
- If you are running on MVS 5.2 or a subsequent release, you can use the MVS SETXCF START,ALTER command to expand the size of the GBP (the maximum size to which you can expand is determined by the SIZE parameter in the CFRM policy that was used to allocate the GBP).

If you don't do anything to relieve the storage shortage, message DSNB325A appears if the GBP reaches 90 percent full.

Message DSNB327I appears when the GBP storage shortage is relieved.

You can monitor the amount of GBP storage in use with the DISPLAY GBPOOL command and with the DB2 statistics trace. Monitor the castout activity to ensure that castouts are keeping up with the rate of GBP-write activity. You can use Performance trace classes 20 and 21 (specifically, IFCIDs 0261, 0262 and 0263) to do more detailed monitoring of the DB2 castout processing.

DSNB320I   *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE***instance-number* **PARTITION** *partition-number* **IN GROUP BUFFERPOOL** *pool-name* **IS IN GROUP BUFFERPOOL RECOVERY PENDING STATE**

Explanation: The named table space or index space partition is in group buffer pool recovery pending (GRECP) status.

csect-name
>   The name of the control section that issued the message.

database-name
>   The name of the database.

space-name
>   The name of the table space or index space.

instance-number
>   The instance number of the table space or index space.

partition-number
>   The partition number.

pool-name
> The name of the group buffer pool.

System action: The data in the named partition is not accessible.

User response: Issue the START DATABASE command on the partition to remove the GRECP status and make the data accessible again. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

Related reference:

⇨ -START DATABASE (DB2) (DB2 Commands)

⇨ RECOVER (DB2 Utilities)

⇨ LOAD (DB2 Utilities)

⇨ Group buffer pool RECOVER-pending status (DB2 Utilities)

DSNB321I   *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE** *instance-number* **IN GROUP BUFFERPOOL** *pool-name* **IS IN GROUP BUFFERPOOL RECOVERY PENDING STATE**

Explanation: The named table space or index space is in group buffer pool recovery pending (GRECP) status.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

space-name
> The name of the table space or index space.

instance-number
> The instance number of the table space or index space.

pool-name
> The name of the group buffer pool.

System action: The table space or index space is not accessible.

User response: Issue the START DATABASE command on the partition to remove the GRECP status and make the data accessible again. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

Related reference:

⇨ -START DATABASE (DB2) (DB2 Commands)

⇨ RECOVER (DB2 Utilities)

⇨ LOAD (DB2 Utilities)

⇨ Group buffer pool RECOVER-pending status (DB2 Utilities)

DSNB322I   *csect-name* **- TABLE SPACE OR INDEX SPACE WITH DBID X'***dbid***' PSID X'***psid***' PARTITION** *part-number* **IN GROUP BUFFERPOOL** *gbpname* **IS IN GROUP BUFFERPOOL RECOVERY PENDING STATE AND CANNOT BE AUTOMATICALLY RECOVERED**

Explanation: This message reports the group buffer pool recovery pending (GRECP) status of the table space or index space partition identified by the internal identifiers DBID and PSID. This message is issued instead of message DSNB320I when the name of the table space or index space is not available. The partition cannot be automatically recovered because there are retained physical locks (P-locks) on the partition.

System action: The data in the identified partition is not accessible.

User response: Issue the -DISPLAY DATABASE(*) SPACENAM(*) RESTRICT command to determine the table spaces or index spaces partition in GRECP status. First, start the failed DB2 members that hold retained P-locks on the partition.Issue the -START DATABASE command on the partition to remove the GRECP status and make the data accessible again. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

DSNB323I *csect-name* **- TABLE SPACE OR INDEX SPACE WITH DBID X'*dbid*' PSID X'*psid*' IN GROUP BUFFERPOOL *gbpname* IS IN GROUP BUFFERPOOL RECOVERY PENDING STATE AND CANNOT BE AUTOMATICALLY RECOVERED**

Explanation: This message reports the group buffer pool recovery pending (GRECP) status of the table space or index space identified by the internal identifiers DBID and PSID. This message is issued instead of message DSNB321I when the name of the table space or index space is not available. The table space or index space cannot be automatically recovered because there are retained physical locks (P-locks) on the space.

System action: The table space or index space is not accessible.

User response: Issue the -DISPLAY DATABASE(*) SPACENAM(*) RESTRICT command to determine the table spaces or index spaces in GRECP status. First, start the failed DB2 members that hold retained P-locks on the table space or index space.Issue the -START DATABASE command on the table space or index space to remove the GRECP status and make the data accessible again. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

DSNB325A *csect-name* **THERE IS A CRITICAL SHORTAGE OF SPACE IN GROUP BUFFER POOL *gbp-name***

Explanation: The amount of storage in the group buffer pool (GBP) coupling facility structure available for writing new pages is critically low. The GBP is 90 percent full with changed data. As the GBP fills with changed data, the castout thresholds are continuously reached, and the castout engines must continuously work to attempt to keep the number of changed pages in the GBP below the castout thresholds. In addition, the more the GBP is filled with changed data, the less efficient are the coupling facility storage management algorithms. Completely running out of GBP storage can result in transaction delays and in pages being added to the logical page list (LPL).

System action: DB2 processing continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

This message might not be automatically deleted in a pre-SP5.2.1-JES3 environment.

Operator response: Contact the system programmer to fix the storage problem.

System programmer response: You can relieve the GBP storage shortage by taking one or more of the following actions:

- Use the ALTER GROUPBUFFERPOOL command to lower the CLASST and GBPOOLT castout thresholds.
- Use the ALTER GROUPBUFFERPOOL command to lower the GBPCHKPT GBP checkpoint frequency.
- Use the MVS SETXCF START,REBUILD command to expand the GBP coupling facility structure. If you rebuild, you can use the ALTER GROUPBUFFERPOOL command to lower the RATIO so that a higher percentage of the GBP storage is given to data entries over directory entries.
- If you are running on MVS 5.2 or a subsequent release, you can use the MVS SETXCF START,ALTER command to expand the size of the GBP (the maximum size to which you can expand is determined by the SIZE parameter in the CFRM policy that was used to allocate the GBP).

If you do not do anything to relieve the storage shortage, message DSNB228I appears if a write to the GBP fails due to a 'GBP full' condition. If DB2 detects a 'full' condition, it will suspend the transaction, initiate castout activity, and then retry the write to the GBP. If the GBP write still fails after several retry attempts, then DB2 adds the page to LPL.

Message DSNB327I is displayed when the GBP storage shortage is relieved.

You can monitor the amount of GBP storage in use with the DISPLAY GBPOOL command and with the DB2 statistics trace. Monitor the castout activity to ensure that castouts are keeping up with the rate of GBP-write activity. You can use Performance trace classes 20 and 21 (specifically, IFCIDs 0261, 0262 and 0263) to do more detailed monitoring of the DB2 castout processing.

DSNB327I   *csect-name* **GROUP BUFFER POOL** *gbp-name* **HAS ADEQUATE FREE SPACE**

Explanation: The group buffer pool (GBP) coupling facility structure has adequate space available for the writing of new pages to the GBP.

This message may be issued when either of the following occurs:

• Rebuilding of GBP reduces changed page count to 60% or less of GBP capacity.

• 'changed data.' entries decrease to 60% or lower when they were previously 75% GBP capacity or higher.

System action: DB2 processing continues.

DSNB330I   *csect-name* **GROUP BUFFER POOL** *gbpname* **CANNOT BE CONNECTED FOR REBUILD DB2 REASON CODE =** *reason1* **MVS IXLCONN REASON CODE =** *reason2*

Explanation: DB2 was not able to connect to the named group buffer pool for rebuild. *reason1* is the DB2 reason code. *reason2* is the MVS IXLCONN reason code. See *MVS/ESA Programming: Sysplex Services Reference* for an explanation of MVS IXLxxxx reason codes.

System action: DB2 stops the rebuild.

Problem determination: If the MVS IXLCONN reason code is xxxx0C08 (structure allocation failed because there was no suitable facility to allocate based on the preference list in the MVS CFRM administrative policy), or if the MVS IXLCONN reason code is xxxx084C (user does not have proper Security Access Facility (SAF) authorization), then IFCID 0250, if active, contains additional diagnostic information.

DSNB331I   *csect-name* **REBUILD STARTED FOR GROUP BUFFER POOL** *gbpname* **REASON =** *reason*

Explanation: A rebuild has been initiated for the named group buffer pool. *reason* indicates the reason why the rebuild was initiated. *reason* can be one of the following:

OPERATOR
> The operator initiated a SETXCF command to initiate the rebuild.

LOSSCONN
> MVS detected a loss of connectivity to the group buffer pool.

This message is issued by each DB2 member participating in the rebuild.

System action: The DB2 members that are actively connected to the named group buffer pool participate in the rebuild. DB2 accomplishes the group buffer pool rebuild processing in two phases:

1. Quiesce phase. Each member quiesces all local access to the group buffer pool.

2. Connect phase. Each member connects to the new structure. Each page set or partition castout owner that has connectivity to original structure is responsible for writing pages from the original structure to the new one. If there are not enough data entries in the new structure, the page set and partition owners cast those pages out to DASD instead. Each member issues message DSNB332I after it completes this activity. After all members have completed their castout owner rebuild work, the group buffer pool structure owner does the final sweep and checks to see if there are any changed pages still remaining in the original structure. If so, the structure owner reads the changed pages from the original structure and writes them to the new one. The structure owner issues message DSNB333I after it has completed the final sweep.

After all members have responded to MVS indicating that its connect phase processing is complete, MVS then notifies each member that the rebuild is complete. Each DB2 member issues message DSNB338I indicating that the rebuild has completed successfully.

If an error occurs during the rebuild, the rebuild is stopped. Each DB2 member issues message DSNB335I indicating the reason why the rebuild is stopping, followed by message DSNB339I indicating that the rebuild has not completed successfully. The operator can also stop rebuilds using the MVS SETXCF command.

DSNB332I *csect-name* **THIS MEMBER HAS COMPLETED CASTOUT OWNER WORK FOR GROUP BUFFER POOL** *gbpname* **PAGES CAST OUT FROM ORIGINAL STRUCTURE =** *value1* **PAGES WRITTEN TO NEW STRUCTURE =** *value2*

Explanation: This DB2 member has completed its part of the rebuild processing for the named group buffer pool for the page sets and partitions for which it is the castout owner.

value1 indicates the number of pages that this member has castout to DASD for the rebuild. If the new structure has a sufficient number of data entries, then *value1* should normally be zero because DB2 writes the changed pages from the original structure to the new structure. However, if the new structure does not have sufficient space, then *value1* could be greater than zero because DB2 responds to insufficient space in the new structure by casting the changed pages in the original structure to DASD instead of writing them to the new structure.

value2 indicates how many changed pages this member wrote to the new structure from the original as part of the rebuild process. If the new structure has sufficient space, then each DB2 that has connectivity to the original structure writes the changed pages for page sets and partitions for which it is the castout owner from the original structure to the new structure, and *value2* indicates the number of pages that the DB2 member wrote.

value1 and *value2* would be zero if this member has no page set or partition castout ownership responsibility.

This message is issued by each DB2 member that has connectivity to the original group buffer pool, and it is always be preceded by message DSNB331I or DSNB740I.

System action: If this DB2 member is not the group buffer pool structure owner, then its rebuild work is done, and it notifies MVS that it has completed its part of the rebuild and awaits later notification that all members have completed. If this member is the structure owner, then it proceeds to its final sweep processing (see DSNB333I).

Refer to "System Action" for message DSNB331I for an explanation of the group buffer pool rebuild processing.

DSNB333I *csect-name* **FINAL SWEEP COMPLETED FOR GROUP BUFFER POOL** *gbpname* **PAGES WRITTEN TO NEW STRUCTURE =** *value1*

Explanation: DB2 has completed the final sweep for the rebuild of the named group buffer pool. A final sweep means that the DB2 member that is the group buffer pool structure owner ensures that all changed pages have either been cast out to DASD or have been written to the new structure after all the page set and partition owners have finished their work. If any changed pages are found in the original structure during the final sweep, then the structure owner writes them out to the new structure. *value1* should be zero in most cases for a manual rebuild. *value1* will probably be greater than zero in most cases where the rebuild was initiated because a DB2 lost connectivity to the group buffer pool.

This message is issued by the structure owner and is always preceded by message DSNB332I.

System action: Immediately after issuing this message, the structure owner notifies MVS that the rebuild has completed, and MVS should soon begin notifying each participating member that the rebuild is complete.

Refer to "System Action" for message DSNB331I for an explanation of the group buffer pool rebuild processing.

DSNB335I *csect-name* **REBUILD STOPPED FOR GROUP BUFFER POOL** *gbpname* **REASON =** *reason1* **DB2 REASON CODE =** *reason2*

Explanation: A rebuild has been stopped for the named group buffer pool.

reason1 indicates why the rebuild was stopped:

OPERATOR
> The operator initiated an MVS SETXCF command to stop the rebuild.

LOSSCONNNEW
> One or more of the DB2 members lost connectivity to the new group buffer pool structure while the rebuild was in progress.

LOSSCONNOLD
> One or more of the DB2 members lost connectivity to the original group buffer pool structure while the rebuild was in progress.

STRFAILNEW
> The new group buffer pool structure failed while the rebuild was in progress.

STRFAILOLD
> The original group buffer pool structure failed while the rebuild was in progress.

DB2REASON
> DB2 decided to stop the rebuild. The reason code given in *reason2* indicates why DB2 stopped the rebuild.

INSUFFCONN
> New structure does not provide connectivity that is better than or equivalent to the current structure.

NOBETRCONN
> New structure does not provide better connectivity than the current structure for a *LossConn* rebuild

If *reason1* indicates 'DB2REASON', then *reason2* specifies a DB2 reason code that indicates why DB2 stopped the rebuild. If *reason1* is not 'DB2REASON', then *reason2* is zero.

This message is normally preceded by message DSNB331I, which indicates that the rebuild was started and the reason why the rebuild was started. This message is issued by each DB2 member participating in the rebuild. This message is followed by message DSNB339I.

System action: The new structure is discarded. If the original structure is still usable, then all members that still have connectivity to the original structure resume using it. Members that do not have connectivity to the original structure disconnect, and their connections become failed-persistent.

If the original structure is not usable, then DB2 initiates damage assessment processing for the group buffer pool.

System programmer response: If the rebuild stopped because of "DB2REASON", see the explanation for the reason code given.

DSNB338I   *csect-name* **THE REBUILD FOR GROUP BUFFER POOL** *gbpname* **HAS COMPLETED SUCCESSFULLY**

Explanation: The rebuild for the named group buffer pool has successfully completed.

This message is preceded by message DSNB331I which indicates the reason why the rebuild was started. This message is issued by each DB2 member participating in the rebuild.

System action: The original structure is discarded and all DB2s that are connected to the group buffer pool begin using the new structure.

DSNB339I   *csect-name* **THE STOP OF THE REBUILD FOR GROUP BUFFER POOL** *gbpname* **HAS COMPLETED**

Explanation: The stop of the rebuild for the named group buffer pool has completed.

This message is preceded by message DSNB335I which indicates the reason why the rebuild was stopped. This message is issued by each DB2 member participating in the rebuild.

System action: The new structure is discarded. If the original structure is still usable, then all members that still have connectivity to the original structure resume using it, and members that do not have connectivity to the original structure disconnect and their connection goes into a failed-persistent state.

If the original structure is not usable, then DB2 initiates damage assessment processing for the group buffer pool.

DSNB340I   *csect-name* **UNEXPECTED ERROR DURING REBUILD FOR GROUP BUFFER POOL** *gbpname* **MVS** *service-name* **RETURN CODE =** *rc* **MVS** *service-name* **REASON CODE =** *reason*

Explanation: DB2 received an unexpected error from MVS during rebuild processing for the named group buffer pool.

service-name is one of the following:
- IXLREBLD
- IXLEERSP

rc is the return code from the MVS service and *reason* is the reason code.

System action: DB2 stops the rebuild.

System programmer response: See *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the return and reason code for the named MVS service.

DSNB350I   *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE** *instance-number* **CANNOT BE AUTOMATICALLY RECOVERED. DB2 REASON CODE =** *reason-code*

Explanation: Following a group buffer pool failure, DB2 cannot automatically recover the named table space or index space.

csect-name
>   The name of the control section that issued the message.

database-name
>   The name of the database that is associated with the recovery.

space-name
>   The name of the table space or index space that is associated with the recovery.

instance-number
>   The instance number of the table space or index space.

reason-code
>   A numeric value that indicates the reason for the failure.

System action: The table space or index space remains in group buffer pool recovery pending (GRECP) status and is inaccessible.

User response: Use the START DATABASE command on the table space or index space to remove the GRECP status and make the data accessible. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

Related reference:

⇨ -START DATABASE (DB2) (DB2 Commands)

⇨ RECOVER (DB2 Utilities)

⇨ LOAD (DB2 Utilities)

⇨ Group buffer pool RECOVER-pending status (DB2 Utilities)

DSNB351I   *csect-name* **DBNAME** *database-name* **SPACENAME** *space-name* **INSTANCE** *instance-number* **PARTITION** *partition-number* **CANNOT BE AUTOMATICALLY RECOVERED. DB2 REASON CODE =** *reason-code*

Explanation: Following a group buffer pool failure, DB2 cannot automatically recover the named table space or index space partition.

csect-name
>   The name of the control section that issued the message.

database-name
>   The name of the database that is associated with the recovery.

space-name
>   The name of the table space or index space that is associated with the recovery.

instance-number
>   The instance number of the table space or index space.

partition-number
>   The partition number.

reason-code
>   A numeric value that indicates the reason for the failure.

System action: The partition remains in group buffer pool recovery pending (GRECP) status and is inaccessible.

User response: Use the START DATABASE command on the partition to remove the GRECP status and make the

data accessible. You can also use the RECOVER or LOAD utility with the REPLACE option to clear the GRECP status.

Severity: 4 (warning)

Related reference:

➡ -START DATABASE (DB2) (DB2 Commands)

➡ RECOVER (DB2 Utilities)

➡ LOAD (DB2 Utilities)

➡ Group buffer pool RECOVER-pending status (DB2 Utilities)

DSNB352I *csect-name* **AUTOMATIC RECOVERY CANNOT BE INITIATED FOR GROUPBUFFER POOL** *gbp-name* **REASON =** *reason*

Explanation: Following a failure of the named group buffer pool, DB2 is not able to initiate automatic recovery.

reason indicates the reason why automatic recovery was not initiated:

AUTORECNO
        The AUTOREC option is set to NO for the group buffer pool.

CONNRELEASE
        At least one actively connected DB2 member is not at the proper release to support automatic recovery.

System action: The page sets and partitions that are in group buffer pool recovery pending (GRECP) status remain inaccessible.

User response: Use the DB2 -START DATABASE command on the page sets and partitions to remove the GRECP status and make the data accessible. You can also use the RECOVERor LOAD utility with the REPLACE option to clear the GRECP status.

If *reason* is AUTORECNO and you want to enable automatic recovery for the group buffer pool, use the ALTER GROUPBUFFERPOOL command to set AUTOREC to YES for the group buffer pool.

If *reason* is CONNRELEASE, then you must ensure that all DB2 members in the group are running at the appropriate release. Automatic recovery is not supported unless all members are running Version 5 or a later release.

Severity: 4 (warning)

Related reference:

➡ -START DATABASE (DB2) (DB2 Commands)

➡ -ALTER GROUPBUFFERPOOL (DB2) (DB2 Commands)

➡ LOAD (DB2 Utilities)

➡ RECOVER (DB2 Utilities)

➡ Group buffer pool RECOVER-pending status (DB2 Utilities)

➡ Data sharing status descriptions (DB2 Data Sharing Planning and Administration)

DSNB353I *csect-name* **ASYNCHRONOUS GRECP RECOVERY IN PROGRESS FOR GROUP BUFFER POOL** *bpoolname*

Explanation: Following a group buffer pool failure, DB2 triggers an automatic recovery process to attempt to recover GRECP page sets for the specified group buffer pool. This message is issued once by any DB2 member who is the page set castout owner on one of the objects being recovered.

System action: Check for the DSNB354I message to verify the completion of the automatic recovery process.

DSNB354I   *csect-name* **ASYNCHRONOUS GRECP RECOVERY HAS COMPLETED FOR GROUP BUFFER POOL**
bpoolname.

Explanation: DB2 has completed the GRECP recovery process for those GRECP page sets in the group buffer pool specified in the message. This message is issued once by any DB2 member who previously issued a DSNB353I message, who is the page set castout owner on one of the objects being recovered.

DSNB355I   *csect-name* **GROUP BUFFER POOL** *gbpname* **RECOVERY LRSN VALUES MIGHT CAUSE A DELAY
IN GRECP RECOVERY. PAGE LEVEL =** *minimum-page-lrsn* **MEMBER LEVEL =** *minimum-member-lrsn*

Explanation: This message is issued at a group buffer pool checkpoint, if the minimum page LRSN or the minimum member LRSN that are used in recovery from the group buffer pool recovery pending (GRECP) state meet either of the following conditions:

* The minimum page LRSN or minimum member LRSN is older than the time when the third-to-last group buffer pool checkpoint was taken.

  For example, suppose that group buffer pool checkpoints are taken at times T1, T2, T3, and T4. At time T4, if DB2 detects that the time of the minimum page LRSN or minimum member LRSN is earlier than time T1, this message is issued.

* The minimum page LRSN or the minimum member LRSN is X'FFFFFFFFFFFF' or X'FFFFFFFFFFFFFFFFFFFF', depending on whether the LRSN is six bytes or ten bytes.

minimum-page-lrsn
>   The minimum page LRSN. This value is the oldest changed page clean-to-dirty LRSN that was recorded at the time of the last group buffer pool checkpoint.

minimum-member-lrsn
>   The minimum member LRSN. This value is the minimum member-level, write-pending LRSN that was recorded at the last group buffer pool checkpoint. DB2 records a restart/redo LRSN for each member of the data sharing group at a group buffer pool checkpoint. The minimum member LRSN reflects the minimum value across all members. If the group buffer pool fails, DB2 does a merge log scan to recover the page sets and partitions that are in GRECP status. DB2 uses each member's restart/redo LRSN in conjunction with the minimum page LRSN to determine the starting point for the merge log scan.

System action: DB2 processing continues. This message is displayed on the console as a highlighted message until either of the following situations occurs:

* The minimum page LRSN and minimum member LRSN are newer than the third-to-last group buffer pool checkpoint.

* The member that issued the message disconnects from the group buffer pool.

User response: If this message is displayed after all DB2 data sharing members are running normally or have shut down normally, contact IBM Software Support for assistance.

If this message is displayed after some DB2 data sharing members shut down abnormally, restart those DB2 members.

DSNB357I   *csect-name* **AUTOMATIC LPL RECOVERY SUPPRESSED FOR OBJECT TYPE** *object-type* **OBJECT
NAME** *object-name* **REASON** *reason-type*

Explanation: One or more pages in the object that is identified in the message have been added to the logical page list (LPL). However, DB2 cannot initiate automatic LPL recovery due to an environmental condition that prevents automatic LPL recovery.

The object could be a table space, an index space, or a partition within a table space or an index space that is identified by the *object-type*. *object-name* identifies the name.

If the object is a table space or an index space, the format of the name is *dbname.spacename*. If the object is a partition within a table space or an index space, the format of the name is *dbname.spacename.partnumber*. *dbname* is the database name. *spacename* is the table space or the index space name. *partnumber* is the partition number.

reason indicates the condition that prevents DB2 from being able to initiate automatic LPL recovery. Here is a list of the potential values for *reason*:

DASD   DB2 encountered a DASD I/O error.

DSNB401I

RESTART
> DB2 is in Restart or End_Restart mode.

GBP   DB2 Group Buffer Pool structure failure or 100% loss of connectivity.

System action: The Automatic LPL Recovery is not initiated.

Programmer response: Check the reason that is specified to solve the system condition. To recover the LPL page, perform one of the following actions:

- Issue the START DATABASE command with the SPACENAM option
- Run the RECOVER or LOAD utility with REPLACE keyword.

DSNB401I   **BUFFERPOOL NAME** *bp-name*, **BUFFERPOOL ID** *bp-id*, **USE COUNT** *use-count*

Explanation: This message displays output from the DISPLAY BUFFERPOOL command. For each buffer pool, this message marks the beginning of multiple lines of information about that buffer pool. Some lines in the output have their own message numbers or alphanumeric identifiers to assist with identification.

The first line (DSNB401I) contains the following information:

bp-name
> The external name of the buffer pool. *bp-name* can be one of the following values: BP0, BP1 - BP49, BP8KB, BP16KB, BP32K, BP32K1 - BP32K9.

bp-id   The internal identifier for the buffer pool. *bp-id* can be one of the following values: 0 - 49, 80 - 89.

use-count
> The number of open table spaces or index spaces that use this buffer pool. A value of zero means that the buffer pool is inactive.

The remaining output for each buffer pool consists of one or more of the following sections, in the indicated order:

- DSNB402I: Information about allocation status
- DSNB404I: Threshold information
- DSNB406I: PGFIX and PGSTEAL information
- DSNB409I: Start time of the statistics interval for DETAIL(INTERVAL)
- DSNB410I: Start time of the statistics interval for DETAIL(*)
- DSNB411I: Page read statistics
- DSNB412I: Sequential prefetch statistics
- DSNB413I: List prefetch statistics
- DSNB414I: Dynamic prefetch statistics
- DSNB415I: Prefetch statistics
- DSNB420I: Page write statistics
- DSNB421I: Page-write threshold statistics
- DSNB440I: Parallel activity statistics
- DSNB441I: LPL activity statistics
- DSNB453I: Cached and changed page statistics
- DSNB455I: Synchronous I/O delay statistics
- DSNB456I: Asynchronous I/O delay statistics
- DSNB460I: Page set and partition list information (for a data-sharing environment)
- DSNB464I: Page set and partition list information (for a non-data-sharing environment)
- DSNB466I: Page set and partition statistics
- DSNB467I: Data set statistics
- DSNB546I: Frame size allocation

If information cannot be reported, one or more of the following messages is returned:

- DSNB408I: No detail statistics available
- DSNB459I: Open failure for a data set

- DSNB463I: No objects matched selection criteria

The DISPLAY BUFFERPOOL output ends with one of the following messages:
- DSN9022I: Normal completion
- DSNB499I: Display terminated because of insufficient space

DSNB402I: Information about allocation status:

The basic buffer pool information is followed by a description of the allocation status of the buffer pool.

The values of VPSIZE MINIMUM and VPSIZE MAXIMUM are used only when AUTOSIZE =YES.

BUFFER POOL SIZE = *pool-size* **BUFFERS**
> The user-specified buffer pool size.

AUTOSIZE = *autosize*
> The buffer pool AUTOSIZE attribute that is applicable to the current allocation of the buffer pool.
>
> **YES** The buffer pool uses Workload Manager (WLM) services, if available, to automatically adjust the size of the buffer pool. The size is adjusted based on dynamic monitoring of the workload goals and the storage that is available on the system.
>
> **NO** The buffer pool does not use WLM services for automatic sizing adjustment of buffer pools.

VPSIZE MINIMUM = *minimum size*
> The minimum size of the buffer pool. This value is meaningful only when the value of *autosize* is YES. If the value was never specified, a value of 0 is shown, and DB2 uses a default value of 75% of the specified value of VPSIZE.

VPSIZE MAXIMUM = *maximum-size*
> The maximum size of the buffer pool. This value is meaningful only when the value of *autosize* is YES. If the value was never specified, a value of 0 is shown, and DB2 uses the defaults value, which is 25% greater than the specified VPSIZE value.

ALLOCATED = *allocated-buffers*
> The number of allocated buffers in an active buffer pool.

TO BE DELETED = *delete-buffers*
> The number of buffers to be deleted in an active buffer pool because of pool contraction.

IN-USE/UPDATED = *current-buffers*
> The number of currently active buffers in the buffer pool. These buffers cannot be stolen.

DSNB404I: Threshold information:

The information about allocation status is followed by information about the user-modifiable thresholds for the buffer pool.

```
DSNB404I - THRESHOLDS -
          VP SEQUENTIAL            = vpseq
          DEFERRED WRITE           = dwt
          VERTICAL DEFERRED WRT    = dwv1,dwv2
          PARALLEL SEQUENTIAL      = vppseqt
          ASSISTING PARALLEL SEQT  = vpxpseqt
```

vpseq The sequential steal threshold for the virtual pool, expressed as a percentage of the total buffer pool size.

dwt The free-buffer deferred write threshold for the buffer pool, expressed as a percentage of the total buffer pool size.

dwv1 The vertical deferred write threshold for the buffer pool, expressed as a percentage of the total buffer pool size.

dwv2 The vertical deferred write threshold for the buffer pool, expressed as an absolute number of buffers. *dwv2* is used to determine the threshold only if *dwv1* is 0 and *dwv2* is non-zero. Otherwise, *dwv1* is used to determine the threshold.

vppseqt The sequential threshold for parallel query processing, expressed as a percentage of the virtual sequential steal threshold. When the threshold is set to 0, parallel query processing is not supported.

DSNB401I

vpxpseqt
> The assisting parallel sequential threshold, expressed as a percentage of the sequential threshold for parallel query processing. Use the *vpxpseqt* value to control how much buffer resource is used when this DB2 member is assisting another member of the group in parallel processing. When the threshold is set to 0, this buffer pool is not used to assist other data-sharing members in processing a query. In a non-data- sharing environment, this value is ignored.

DSNB406I: PGFIX and PGSTEAL information:

The DISPLAY BUFFERPOOL output includes a description of the PGFIX and PGSTEAL attributes for the buffer pool.

CURRENT = *current-pgfix*
> The value of the page fix (PGFIX) attribute for the current allocation of the buffer pool.

> **YES** The buffer pool is fixed in real storage for the long term.

> **NO** The buffer pool is fixed in real storage only during an I/O operation.

PENDING = *pending-pgfix*
> The value of the PGFIX attribute that is to be applied for the next allocation of the virtual buffer pool.

> **YES** The buffer pool is to be fixed in real storage for the long term.

> **NO** The buffer pool is to be fixed in real storage only during an I/O operation.

PAGE STEALING METHOD = *current-pgsteal*
> The page stealing method (PGSTEAL) that is in use for the buffer pool.

> **LRU** The least recently used (LRU) algorithm is used to manage page stealing. This method is the default value.

> **FIFO** The first-in, first-out (FIFO) algorithm is used to manage page stealing.

> **NONE** No page stealing occurs. Objects that use this buffer pool are kept resident.

Related information:

Fixing a buffer pool in real storage (DB2 Performance)

Choosing a page-stealing algorithm (DB2 Performance)

DSNB408I: No detail statistics available:

DSNB408I indicates that, although you specified the DETAIL option on the DISPLAY BUFFERPOOL command, no detail statistics are available for the requested buffer pool. The statistics are not available because the pool has not been activated since DB2 started.

BUFFER POOL *bp-name*
> The name of the buffer pool.

DSNB409I: Start time of the statistics interval for DETAIL(INTERVAL):

When you specify the DETAIL(INTERVAL) option, the output includes the start time of the interval for which the statistics were accumulated.

`DSNB409I - INCREMENTAL STATISTICS SINCE ` *`base-time`*

base-time
> The start time. This value is either the time of the previous incremental display or, if no previous incremental display exists, the time that the buffer pool was first activated.

> The format is *hh:mm:ss month dd, yyyy*

> *hh:mm:ss*
>> The time expressed as hour:minutes:seconds.

> *month* An alphanumeric abbreviation for the month. For example, a value of OCT means October.

> *dd* The day of the month.

> *yyyy* The year.

DSNB410I: Start time of the statistics interval for DETAIL(*):

When you specify the DETAIL(*) option, the output includes the start time of the interval for which the statistics were accumulated.

DSNB410I - CUMULATIVE STATISTICS SINCE *base-time*

base-time

> The start time. This value is the time the buffer pool was first activated.

> The format is *hh:mm:ss month dd, yyyy*

> *hh:mm:ss*

>> The time expressed as hour:minutes:seconds.

> *month* An alphanumeric abbreviation for the month. For example, a value of OCT means October.

> *dd* The day of the month.

> *yyyy* The year.

DSNB411I: Page read statistics:

When you specify the DETAIL option, the output includes the page read statistics for the buffer pool.

```
DSNB411I - RANDOM GETPAGE    = rgp
           SYNC READ I/O (R) = srr
           SEQ.   GETPAGE    = sgp
           SYNC READ I/O (S) = srs
           DMTH HIT          = dmt
           PAGE-INS REQ      = pir
           SEQUENTIAL        = seq
           VPSEQT HIT        = vsh
           RECLASSIFY        = rcy
```

RANDOM GETPAGE = *rgp*
> The number of nonsequential GETPAGE requests.

SYNC READ I/O (R) = *srr*
> The number of synchronous read I/O operations for nonsequential GETPAGE requests.

SEQ. GETPAGE = *sgp*
> The number of sequential GETPAGE requests.

SYNC READ I/O (S) = *srs*
> The number of synchronous read I/O operations for sequential GETPAGE requests.

DMTH HIT = *dmt*
> The number of times that the data management threshold was reached.

PAGE-INS REQ = *pir*
> The number of page-in operations that are required for read I/O.

SEQUENTIAL = *seq*
> The number of buffers on the sequential least-recently-used (SLRU) chain.

VPSEQT HIT = *vsh*
> The number of times that the size of the SLRU chain reached the sequential steal threshold (the VPSEQT value) for the buffer pool.

RECLASSIFY = *rcy*
> A statistic that is used by IBM for serviceability.

Related information:

> Buffer pool thresholds that you can change (DB2 Performance)

> -ALTER BUFFERPOOL (DB2) (DB2 Commands)

DSNB412I: Sequential prefetch statistics:

When you specify the DETAIL option, the output includes the sequential prefetch statistics for the buffer pool.

REQUESTS = *pft*
> The number of times that sequential prefetch was requested.

DSNB401I

| **PREFETCH I/O =** *pio*
| The number of sequential prefetch read I/O operations.

| **PAGES READ =** *pfp*
| The number of pages that are read because of sequential prefetch.

| **Related information:**
| Sequential prefetch (PREFETCH='S') (DB2 Performance)

| **DSNB413I: List prefetch statistics:**

| When you specify the DETAIL option, the output includes the list prefetch statistics for the buffer pool.

| **REQUESTS =** *pft*
| The number of times that list prefetch was requested.

| **PREFETCH I/O =** *pio*
| The number of list prefetch read I/O operations.

| **PAGES READ =** *pfp*
| The number of pages that are read because of list prefetch.

| **Related information:**
| List prefetch (PREFETCH='L' or 'U') (DB2 Performance)

| **DSNB414I: Dynamic prefetch statistics:**

| When you specify the DETAIL option, the output includes the dynamic prefetch statistics for the buffer pool.

| **REQUESTS =** *pft*
| The number of times that dynamic prefetch was requested.

| **PREFETCH I/O =** *pio*
| The number of dynamic prefetch read I/O operations.

| **PAGES READ =** *pfp*
| The number of pages that are read because of dynamic prefetch.

| **Related information:**
| Dynamic prefetch (PREFETCH='D') (DB2 Performance)

| **DSNB415I: Prefetch statistics:**

| When you specify the DETAIL option, the output includes the prefetch statistics for the buffer pool.

| **NO BUFFER =** *pfd*
| The number of times that prefetch was disabled for one of the following reasons:
| • The buffer pool reached the prefetch disabled threshold (90% full).
| • A user disabled prefetch by setting the VPSEQT threshold for the buffer pool to zero.

| **NO READ ENGINE =** *ree*
| The number of times that prefetch was disabled because an asynchronous read processor was not available.

| **Related information:**
| Prefetch I/O (DB2 Performance)

| **DSNB420I: Page write statistics:**

| When you specify the DETAIL option, the output includes the page write statistics for the buffer pool.

| **SYS PAGE UPDATES =** *pages-updated*
| The number of buffer updates.

| **SYS PAGES WRITTEN =** *pages-written*
| The number of pages that are written to disk.

| **ASYNC WRITE I/O =** *async-writes*
| The number of asynchronous write I/O operations.

| **SYNC WRITE I/O =** *sync-writes*
        The number of synchronous write I/O operations.

| **PAGE-INS REQ =** *page-ins*
        The number of page-ins that are required for write I/O.

| **DSNB421I: Page-write threshold statistics:**

| When you specify the DETAIL option, the output includes the page-write threshold statistics for the buffer pool.

| **DWT HIT =** *dwt*
        The number of times that the deferred write threshold was reached.

| **VERTICAL DWT HIT =** *vdw*
        The number of times that the vertical deferred write threshold was reached.

| **Related information:**
     Buffer pool thresholds that you can change (DB2 Performance)

| **DSNB440I: Parallel activity statistics:**

| When you specify the DETAIL option, the output includes statistics about parallel activities for the buffer pool.

| **PARALLEL REQUEST =** *tpa*
        The total number of negotiations with the buffer pool for the requested number of sequential prefetch
        streams.

| **DEGRADED PARALLEL =** *dpa*
        The total number of times that the negotiation resulted in the degraded mode of parallel operations.

| **DSNB441I: LPL activity statistics:**

| When you specify the DETAIL option, the output includes statistics about LPL activity for the buffer pool.

```
| DSNB441I - LPL ACTIVITY -
|           PAGES ADDED        =         pages
```

| *pages*   The total number of pages for all page sets that are added to the logical page list (LPL) in this buffer pool.
           This value is equal to the number of DSNB250E messages that are written to the system log after the most
           recent execution of the DISPLAY BUFFERPOOL command with the DETAIL option.

| **Related information:**
     "DSNB250E" on page 50

     Displaying the logical page list (DB2 Administration Guide)

     Characteristics of pages that are in error (DB2 Administration Guide)

| **DSNB453I: Cached and changed page statistics:**

| The output includes the number of cached pages and changed pages in the buffer pool for a data set if all of the
| following conditions are true:
| • You specified the LSTATS option on the DISPLAY BUFFERPOOL command.
| • The buffer pool is an active buffer pool.
| • The number of cached and changed pages are not zero.

| The relevant table space or index space is identified in either line DSNB464I or line DSNB465I. The data set is
| identified in line DSNB466I.

| **CURRENT =** *vcount*
        The number of cached pages in the virtual pool for the data set. This value is the number of buffers that
        contain pages for the data set in the buffer pool.

| **MAX =** *mvcount*
        The maximum number of cached pages in the virtual pool for the data set since the last DISPLAY
        BUFFERPOOL command with the LSTATS option was issued.

DSNB401I

CHANGED = *ccount*
> The number of changed pages in the virtual pool for the data set. This value is the number of buffers that were changed in the buffer pool for the data set.

MAX = *mccount*
> The maximum number of changed pages in the virtual pool for the data set since the last DISPLAY BUFFERPOOL command with the LSTATS option was issued.

DSNB455I: Synchronous I/O delay statistics :

The output includes synchronous I/O delay statistics if all of the following conditions are true:

- You specified the LSTATS option on the DISPLAY BUFFERPOOL command.
- The buffer pool is an active buffer pool.
- The values of the synchronous I/O delay statistics are not zero.

These synchronous I/O delay statistics are reported for a data set for an open table space or index space that is associated with the buffer pool. The values that are listed are the statistics that were gathered since the last display for the data set.

The relevant table space or index space is identified in either line DSNB464I or line DSNB465I. The relative data set within the table space or index space is identified in line DSNB466I.

```
DSNB455I - SYNCHRONOUS I/O DELAYS -
             AVERAGE DELAY         = avd
             MAXIMUM DELAY         = mxd
             TOTAL PAGES           = tpg
```

avd   The average I/O delay in milliseconds for pages in the data set.

mxd   The maximum I/O delay in milliseconds for pages in the data set.

tpg   The total number of pages that are read or written for the data set.

Related information:
> Read operations (DB2 Performance)

DSNB456I: Asynchronous I/O delay statistics :

The output includes asynchronous I/O delay statistics if all of the following conditions are true:

- You specified the LSTATS option on the DISPLAY BUFFERPOOL command.
- The buffer pool is an active buffer pool.
- The values of the asynchronous I/O delay statistics are not zero.

These asynchronous I/O delay statistics are reported for a data set for an open table space or index space that is associated with the buffer pool. The values that are listed are the statistics that were gathered since the last display for the data set.

The relevant table space or index space is identified in either line DSNB464I or line DSNB465I. The relative data set within the table space or index space is identified in line DSNB466I.

```
DSNB456I - ASYNCHRONOUS I/O DELAYS -
             AVERAGE DELAY         = avd
             MAXIMUM DELAY         = mxd
             TOTAL PAGES           = tpg
             TOTAL I/O COUNT       = tio
```

avd   The average I/O delay in milliseconds for pages in the data set.

mxd   The maximum I/O delay in milliseconds for pages in the data set.

tpg   The total number of pages that are read or written for the data set.

tpg   The total number of I/O operations that are issued for the data set.

DSNB459I: Open failure for a data set:

Message DSNB459I indicates that a previous attempt to access a data set failed because of an allocation or open error.

This message is displayed only when you specify the LIST option on the DISPLAY BUFFERPOOL command for an active buffer pool and this error condition occurs.

The relevant table space or index space is identified in either line DSNB464I or line DSNB465I.

csect-name
>The name of the control section that issued the message.

DATASET = *dsn*
>The data set number. This value is the relative data set number within a table space or index space.

Other data set and buffer pool information might be displayed in subsequent messages.

DSNB460I: Page set and partition list information (for a data-sharing environment):

When you specify the LIST option, the output includes information about page sets and partition lists.

The message output begins with the introductory text `PAGE SET/PARTITION LIST INFORMATION`. The introductory text is followed by column headers and multiple lines of information.

The output contains the following columns, in the indicated order:

DATABASE
>The name of the database. This field is blank when the line provides information about the same database as the preceding line or lines.

SPACE NAME
>The name of the table space. This field is blank when the line provides information about the same table space as the preceding line or lines.

INST   The instance number.

PART   One of the following values:
>- The partition number.
>- For a simple table space or simple index space: a blank.
>- For non-partitioned indexes on a partitioned table space: the logical partition number preceded by the character L (for example, L01).

TS IX   The type of object: either TS for table space or IX for index space.

GBP DEP
>An indicator of group buffer pool (GBP) dependency. The indicator can be either of the following values:
>
>**Y**   The page set or partition is GBP-dependent.
>
>**N**   The page set or partition is not GBP-dependent.

MEMBER NAME
>The name of the member that the detail line pertains to.

CASTOUT OWNER
>An indicator of whether the member is the castout owner. The indicator can be either of the following values:
>
>**Y**   The member is the castout owner.
>
>**Blank**   The member is not the castout owner.

USE COUNT
>The number of active claimers or drainers for the page set or partition for the member.

P-LOCK STATE
>The P-lock state that the member currently holds. The state can be any of the following values:
>
>**IS**   R/O interest. Other members have interest in this page set or partition. The page set or partition is GBP-dependent.
>
>**IX**   R/W interest. Other members have interest in this page set or partition. The page set or partition is GBP-dependent.

| **S** | R/O interest. Other members might be reading the page set or partition. The page set or partition is not GBP-dependent. |

| **SIX** | R/W interest. Other members might be reading the page set or partition. The page set or partition is GBP-dependent. |

| **NSU** | R/W interest. The page set or partition is GBP-dependent. |

| **X** | R/W interest. No other members are accessing the page set or partition. The page set or partition is not GBP-dependent. |

| **US** | A temporary state that can be held by a restarting DB2 when "waiting for retained locks" is enabled. |

| *number* | A number for use as a diagnostic aid. A number is displayed only when a DISPLAY BUFFERPOOL command encounters an undefined lock state. |

If the DISPLAY BUFFERPOOL LIST command finds more than 255 lines of output to display, the output is presented in multiple sections. Each section is a new instance of message DSNB460I with the addition of "(CONTINUED)" in the message heading:

```
DSNB460I  @ (CONTINUED)
```

Message DSNB460I is issued in a data-sharing environment. In a non-data-sharing environment, message DSNB464I is issued instead.

If no information is available, message DSNB460I is followed by message DSNB463I.

DSNB463I: No objects matched selection criteria:

Message DSNB463I indicates that DB2 did not find any page sets or partitions that matched the selection criteria.

For example, this message is displayed for DIS BPOOL(BP0) GBPDEP(Y) if DB2 did not find any page sets or partitions that are group-buffer-pool dependent (GBP-dependent).

When message DSNB463I is returned, the DISPLAY BUFFERPOOL command terminates normally.

If you expected to see a list of page sets or partitions, review and correct the syntax of the DISPLAY BUFFERPOOL command as needed. Consider adding or changing the filter keywords to obtain a list of page sets or partitions.

DSNB464I: Page set and partition list information (for a non-data-sharing environment):

When you specify the LIST option, the output includes information about page sets and partition lists.

The message output begins with the introductory text PAGE SET/PARTITION LIST INFORMATION. The introductory text is followed by column headers and multiple lines of information.

```
DSNB464I  @
PAGE SET/PARTITION LIST INFORMATION
                             TS  USE
DATABASE SPACE NAME  INST PART IX COUNT
======== =========== ==== ==== == =====
```

DATABASE
> The name of the database. This field is blank when the line provides information about the same database as the preceding line or lines.

SPACE NAME
> The name of the table space. This field is blank when the line provides information about the same table space as the preceding line or lines.

INST The instance number.

PART One of the following values:
- The partition number.
- For a simple table space or simple index space, a blank.
- For non-partitioned indexes on a partitioned table space, the logical partition number preceded by the character L (for example, L01).

TS IX The type of object: either TS for table space or IX for index space.

USE COUNT
> The number of active claimers or drainers for the page set or partition for the member.

If the DISPLAY BUFFERPOOL LIST command finds more than 255 lines of output to display, the output is presented in multiple sections. Each section is a new instance of message DSNB464I with the addition of "(CONTINUED)" in the message heading:

```
DSNB464I   @ (CONTINUED)
```

Message DSNB464I is issued in a non-data-sharing environment. In a data-sharing environment, message DSNB460I is issued instead.

If no information is available, message DSNB464I is followed by message DSNB463I.

DSNB466I: Page set and partition statistics:

When you specify the LSTATS option, the output includes statistics for data sets. Those statistics are introduced by message DSNB466I.

DSNB467I: Data set statistics:

When you specify the LSTATS option, the output includes statistics about data sets for objects that are associated with active buffer pools.

STATISTICS FOR *object-type*
> The type of object: either TABLE SPACE or INDEX SPACE.

database-name
> The name of the database.

space-name
> The name of the table space or index space.

instance-number
> The instance number of the table space or index space.

DATA SET #: *set-number*
> The relative data set number within the table space or index space.

USE COUNT: *application-count*
> The number of applications that have a claim or drain on the page set or partition.

This message is followed by one or more of the following messages: DSNB453I, DSNB455I, or DSNB456I.

Related information:
> Claims and drains (DB2 Performance)

DSNB546I: Frame size allocation:

Message DSNB546I describes the frame size allocation status of the buffer pool. If the specified number of buffers with the specified frame size are not available, DB2 allocates some of the buffers with a different frame size. In that case, DB2 issues multiple instances of this message to show the number of buffers that are allocated with each frame size.

```
DSNB546I - PREFERRED FRAME SIZE preferred-size buffer-count BUFFERS USING frame-size FRAME SIZE ALLOCATED
```

preferred-size
> The requested frame size. If you used the ALTER BUFFERPOOL command with the FRAMESIZE option, the specified FRAMESIZE value is displayed as *preferred-size*.
>
> This value can be 4K, 1M, or 2 GB.

buffer-count
> The number of buffers that are allocated with the *frame-size* value.

frame-size
> The actual frame size that DB2 used for the buffers.
>
> This value can be 4K, 1M, or 2 GB.

| **DSN9022I: Normal completion:**

| The DISPLAY THREAD output normally ends with message DSN9022I.

| **Related information:**
| "DSN9022I" on page 1144

| **DSNB499I: Display terminated because of insufficient space:**

| If the DISPLAY BUFFERPOOL output is too long, the output ends with message DSNB499I. This message indicates
| that the command was unable to obtain storage for more messages. This situation occurs only for a long display
| request, such as a detail display for many buffer pools.

| The output is truncated.

| Reissue the DISPLAY BUFFERPOOL command, and specify a smaller number of buffer pools.

| **System action:** Processing continues.

| **Related tasks**:

| ➡ Monitoring and tuning buffer pools by using online commands (DB2 Performance)

| **Related reference**:

| ➡ -DISPLAY BUFFERPOOL (DB2) (DB2 Commands)

| ➡ -ALTER BUFFERPOOL (DB2) (DB2 Commands)

| **DSNB402I**   **BUFFER POOL SIZE =** *pool-size* **BUFFERS AUTOSIZE =** *autosize* **VPSIZE MINIMUM =** *minimum size*
| **VPSIZE MAXIMUM =** *maximum-size* **ALLOCATED =** *allocated-buffers* **TO BE DELETED =** *delete-buffers*
| **IN-USE/UPDATED =** *current-buffers*

| **Explanation:** This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
| message "DSNB401I" on page 68.

DSNB404I   **THRESHOLDS — VP SEQUENTIAL =** *vpseq* **DEFERRED WRITE =** *dwt* **VERTICAL DEFERRED**
WRT = *dwv1,dwv2* **PARALLEL SEQUENTIAL =** *vppseqt* **ASSISTING PARALLEL SEQT =** *vpxpseqt*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB406I   **PGFIX ATTRIBUTE - CURRENT =** *current-pgfix* **PENDING =** *pending-pgfix* **PAGE STEALING**
METHOD = *current-pgsteal*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB408I   **NO DETAIL STATISTICS AVAILABLE FOR BUFFER POOL** *bp-name*

Explanation: This message is part of the output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB409I   **INCREMENTAL STATISTICS SINCE** *base-time*

Explanation: This message is part of the output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB410I   **CUMULATIVE STATISTICS SINCE** *base-time*

Explanation: This message is part of the output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB411I   RANDOM GETPAGE = *rgp* SYNC READ I/O (R) = *srr* SEQ. GETPAGE = *sgp* SYNC READ I/O (S) = *srs* DMTH HIT = *dmt* PAGE-INS REQ = *pir* SEQUENTIAL = *seq* VPSEQT HIT = *vsh* RECLASSIFY = *rcy*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB412I   SEQUENTIAL PREFETCH - REQUESTS = *pft* PREFETCH I/O = *pio* PAGES READ = *pfp*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB413I   LIST PREFETCH - REQUESTS = *pft* PREFETCH I/O = *pio* PAGES READ = *pfp*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB414I   DYNAMIC PREFETCH - REQUESTS = *pft* PREFETCH I/O = *pio* PAGES READ = *pfp*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB415I   PREFETCH DISABLED - NO BUFFER = *pfd* NO READ ENGINE = *ree*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB420I   SYS PAGE UPDATES = *pages-updated* SYS PAGES WRITTEN = *pages-written* ASYNC WRITE I/O = *async-writes* SYNC WRITE I/O = *sync-writes* PAGE-INS REQUIRED = *page-ins*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB421I   DWT HIT= *dwt* VERTICAL DWT HIT=*vdw*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB440I   PARALLEL ACTIVITY — PARALLEL REQUEST = *tpa* DEGRADED PARALLEL = *dpa*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB441I   LPL ACTIVITY - PAGES ADDED = *pages*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB453I   VP CACHED PAGES - CURRENT = *vcount* MAX = *mvcount* CHANGED = *ccount* MAX = *mccount*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB455I   SYNCHRONOUS I/O DELAYS — AVERAGE DELAY = *avd* MAXIMUM DELAY = *mxd* TOTAL PAGES = *tpg*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB456I ASYNCHRONOUS I/O DELAYS — AVERAGE DELAY = *avd* MAXIMUM DELAY = *mxd* TOTAL
PAGES = *tpg* TOTAL I/O COUNT = *tio*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB459I *csect-name* **OPEN FAILURE HAS OCCURRED FOR DATASET** *dsn*

Explanation: This message is output from the DISPLAY BUFFERPOOL command, which is part of message
"DSNB401I" on page 68.

DSNB460I PAGE SET/PARTITION LIST INFORMATION

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB463I * * * NO OBJECTS MATCHED LIST/LSTATS SELECTION CRITERIA

Explanation: This message is output from the DISPLAY BUFFERPOOL command, which is part of message
"DSNB401I" on page 68.

DSNB464I PAGE SET/PARTITION LIST INFORMATION

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB466I ------------------PAGE SET/PARTITION STATISTICS-------------------

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB467I STATISTICS FOR *object-type database-name.space-name* INSTANCE *instance-number* DATA SET #:
set-number USE COUNT: *application-count*

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of
message "DSNB401I" on page 68.

DSNB468I *keyword* KEYWORD IS IGNORED FOR NON-DATA SHARING SYSTEMS

Explanation: The identified keyword is valid only in a data-sharing environment. Keywords GBPDEP and
CASTOWNR can cause this message to be displayed.

System action: The command ignores the keyword and terminates normally.

User response: Re-enter the command without the keyword since the DB2 is a non-data-sharing DB2.

DSNB469I *keyword1* KEYWORD IS IGNORED WHEN *keyword2* KEYWORD IS SPECIFIED

Explanation: *keyword1* is incompatible with *keyword2*. For example, in command DIS BPOOL(BP0) GBPDEP(Y)
LSTATS(*), keyword GBPDEP requests a list of page sets or partitions (as in LIST keyword); however, keyword
GBPDEP is ignored because LIST and LSTATS are two different reports.

System action: The command ignores keyword1 and terminates normally.

User response: Omit keyword1 if keyword2 is used on the next entry of the command.

DSNB499I DISPLAY TERMINATED DUE TO INSUFFICIENT MESSAGE SPACE

Explanation: This message is included in the following output:
• Output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68
• Output from the DISPLAY GROUPBUFFERPOOL command

25

The DISPLAY BUFFERPOOL or the DISPLAY GROUPBUFFERPOOL command was unable to obtain storage for additional messages. This situation should happen for only a particularly long display request, such as a detail display for a large number of buffer pools or group buffer pools.

System action: The display is truncated.

System programmer response: Reissue the DISPLAY BUFFERPOOL or the DISPLAY GROUPBUFFERPOOL command, and specify a smaller number of buffer pools or group buffer pools.

Related reference:

➡ -DISPLAY BUFFERPOOL (DB2) (DB2 Commands)

➡ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

DSNB501I   *bpname* **IS NOT A VALID BUFFER POOL NAME**

Explanation: An invalid buffer pool name was specified on an ALTER BUFFERPOOL or DISPLAY BUFFERPOOL command. THE VALID NAMES ARE BP0, BP1,..., BP49, BP32K, BP32K1,...BP32K9.

System action: The request is rejected.

System programmer response: Reissue the ALTER BUFFERPOOL or DISPLAY BUFFERPOOL command, specifying a valid buffer pool name.

DSNB502I   **ENCOUNTERED AN ERROR WHILE VALIDATING INPUT** *vp-parm* **FOR** *bp-name* **MUST BE BETWEEN** *lower-limit* **AND** *upper-limit* **. SPECIFIED VALUE IS** *specified-value***.**

Explanation: An ALTER BUFFERPOOL command was issued that specified incompatible values for one or more of the following options: VPSIZE, VPSIZEMIN, or VPSIZEMAX.

vp-parm
> The option for which an incompatible value was specified. VPSIZE, VPSIZEMIN, VPSIZEMAX

bp-name
> Buffer pool name.

lower-limit
> Lower limit for the VPSIZE value for a buffer pool. The same limit applies to the VPSIZEMIN value for a buffer pool. For BP0, the lower limit is 2000. For BP8K0, the lower limit is 1000. For BP16K0, the lower limit is 500. For all other buffer pools, the lower limit is 0.

upper-limit
> Upper limits for the VPSIZE of a buffer pool. The same limit applies to the VPSIZEMAX value for a buffer pool. The upper limit is based on the page-size of the buffer pool:
> * 250000000 for 4KB page-size buffer pools
> * 125000000 for 8KB page-size buffer pools
> * 62500000 for 16KB page-size buffer pools
> * 31250000 for 32KB page-size buffer pools

specified-value
> The value that is specified for VPSIZE, VPSIZEMIN or VPSIZEMAX in the ALTER BUFFERPOOL command.

System action: The buffer pool is not modified.

System programmer response: Reissue the ALTER BUFFERPOOL command, and specify a valid value for compatible values for VPSIZE, VPSIZEMIN, and VPSIZEMAX.

Related reference:

➡ -ALTER BUFFERPOOL (DB2) (DB2 Commands)

DSNB504I   VPSEQT FOR *bpname* MUST BE BETWEEN 0 AND 100

Explanation: An ALTER BUFFERPOOL command was issued to try to alter a buffer pool's VPSEQT attribute to an invalid value. VPSEQT is the buffer pool sequential steal threshold. Valid values range from 0 to 100.

System action: The request is rejected.

System programmer response: Reissue the ALTER BUFFERPOOL command, specifying a legitimate VPSEQT value.

DSNB506I   DWQT FOR *bpname* MUST BE BETWEEN 0 AND 90

Explanation: An attempt was made to change a buffer pool's DWQT attribute value using the ALTER BUFFERPOOL command. DWQT is the free buffer deferred write threshold. Valid values range from 0 to 90.

System action: The request is rejected.

System programmer response: Reissue the ALTER BUFFERPOOL command, specifying a legitimate DWQT value.

DSNB507I   VDWQT FOR *bpname* MUST BE BETWEEN 0 AND 90

Explanation: An attempt was made to change a buffer pool's VDWQT attribute to an invalid value using the ALTER BUFFERPOOL command. VDWQT is the vertical deferred write threshold. Valid values range from 0 to 90.

System action: The request is rejected.

System programmer response: Reissue the ALTER BUFFERPOOL command, specifying a legitimate VDWQT value.

DSNB508I   *csect-name* THE TOTAL VPSIZE IS BEYOND THE MAXIMUM LIMIT

Explanation: An ALTER BUFFERPOOL command specified a VPSIZE or VPTYPE value that would cause the sum of the *ssnm*DBM1 address space storage requirements for all buffer pools to exceed 1TB. When calculating the storage requirements for the buffer pools, DB2 uses the page size (4K, 8K, 16K, or 32K) for each pool buffer.

System action: The buffer pool's VPSIZE or VPTYPE is not changed. Processing for other parameters, if requested, proceeds normally.

System programmer response: Rebalance VPSIZE for all buffer pools.

DSNB510I   NO ACTION REQUESTED

Explanation: An ALTER BUFFERPOOL command was issued, but no buffer pool attributes were specified.

System action: The buffer pool's attributes are not changed.

System programmer response: Reissue the ALTER BUFFERPOOL command specifying one or more buffer pool attributes.

DSNB511I   VDWQT CANNOT BE GREATER THAN DWQT

Explanation: An ALTER BUFFERPOOL command specified a DWQT and/or VDWQT value that would result in the buffer pool's VDWQT parameter being greater than its DWQT parameter.

System action: The buffer pool's VDWQT or DWQT is not changed. Processing for other parameters, if requested, proceeds normally.

System programmer response: Rebalance DWQT and VDWQT for the buffer pool.

DSNB512I   *integer* IS THE CURRENT VPSIZE FOR BUFFER POOL *bpname*

Explanation: An ALTER BUFFERPOOL command specified a VPSIZE equal to the buffer pool's current buffer pool size.

System action: The buffer pool size is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB514I   *integer* **IS THE CURRENT VPSEQT FOR BUFFER POOL** *bpname*

Explanation: An ALTER BUFFERPOOL command specified an VPSEQT equal to the current sequential steal threshold of the buffer pool.

System action: The buffer pool sequential threshold is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB516I   *integer* **IS THE CURRENT DWQT FOR BUFFER POOL** *bpname*

Explanation: An ALTER BUFFERPOOL command specified a DWQT equal to the current deferred write threshold of the buffer pool.

System action: The deferred write threshold of the buffer pool is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB517I   *int1,int2* **IS THE CURRENT VDWQT FOR BUFFER POOL** *bpname*

Explanation: An ALTER BUFFERPOOL command specified a VDWQT equal to the current vertical deferred write threshold of the buffer pool.

System action: The vertical deferred write threshold of the buffer pool is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB519I   *csect-name* **PGSTEAL FOR** *bufferpool-name* **HAS BEEN SET TO** *new-pgsteal*.

Explanation: This message is issued in response to an ALTER BUFFERPOOL command to indicate that the requested change to the PGSTEAL parameter was accepted.

csect-name
>   The name of the control section that issued the message.

command-name
>   The name of the MVS command that was issued: START, STOP, or MODIFY. The MODIFY command refreshes the DB2 health monitor.

bufferpool-name
>   The name of the buffer pool.

new-pgsteal
>   The page stealing method (PGSTEAL) that will be used for the buffer pool.
>
>   **LRU**   The buffer pool buffers are managed using the least recently used (LRU) algorithm. This is the default option.
>
>   **FIFO**   The buffer pool buffers are managed using the first-in, first-out (FIFO) algorithm.
>
>   **NONE**   The buffer pool buffers are managed with the no-stealing algorithm. Objects that use this buffer pool are kept resident.

System action: Processing continues.

Related reference:

➡ -ALTER BUFFERPOOL (DB2) (DB2 Commands)

DSNB520I   **READY TO EXPLICITLY DELETE BUFFER POOL** *bpname*

Explanation: This message is issued before draining an active buffer pool in response to an ALTER BUFFERPOOL command, which specified VPSIZE=0 against an active buffer pool.

System action: After this message is issued, DB2 quiesces all current activities against this buffer pool. DB2 locks the table space exclusively and performs physical close for all open table spaces or index spaces that reference this pool. DB2 also prevents subsequent database access that references this pool. SQL applications get an SQLCODE -904 (resource not available) with reason code 00C200E0 to indicate that the requested buffer pool size is not initialized. After the requested buffer pool is deleted, message DSNB521I is issued.

DSNB521I   BUFFER POOL *bpname* HAS BEEN EXPLICITLY DELETED

Explanation: This message is issued after an active buffer pool is explicitly deleted with an ALTER BUFFERPOOL command. This message is always preceded by a corresponding DSNB520I message.

System action: The virtual storage that is associated with this buffer pool is released.

DSNB522I   *specified-option* FOR *bp-name* HAS BEEN SET TO *specified-value*

Explanation: This message is issued in response to an ALTER BUFFERPOOL command to indicate that the requested change to the specified VPSIZE, VPSIZEMIN, VPSIZEMAX or FRAMESIZE option was accepted. For an active pool, the actual expansion, contraction, or deletion does not take place until after this message is issued.

specified-option
        VPSIZE, VPSIZEMIN, VPSIZEMAX or FRAMESIZE

bp-name
        The name of the buffer pool.

specified-value
        The value that was specified in the command.

System action: Processing continues.

System programmer response: Issue the DISPLAY BUFFERPOOL command to determine the actual virtual storage usage for this pool.

DSNB524I   VPSEQT FOR *bpname* HAS BEEN SET TO *integer*

Explanation: This message is issued in response to an ALTER BUFFERPOOL command. It indicates the successful modification of the sequential steal threshold of the buffer pool.

DSNB526I   DWQT FOR *bpname* HAS BEEN SET TO *integer*

Explanation: This message is issued in response to an ALTER BUFFERPOOL command. It indicates the successful modification of the deferred write threshold of the buffer pool.

DSNB527I   VDWQT FOR *bpname* HAS BEEN SET TO *int1,int2*

Explanation: This message is issued in response to an ALTER BUFFERPOOL command. It indicates the successful modification of the vertical deferred write threshold of the buffer pool.

DSNB529I   VPPSEQT FOR *bpname* HAS BEEN SET TO *integer*

Explanation: This message is issued in response to the ALTER BUFFERPOOL command. It indicates the successful modification of the parallel sequential steal threshold of the buffer pool.

DSNB530I   *integer* IS THE CURRENT VPPSEQT FOR BUFFER POOL *bpname*

Explanation: An ALTER BUFFERPOOL command specified a VPPSEQT equal to the current parallel sequential steal threshold of the buffer pool.

System action: The parallel sequential steal threshold of the buffer pool is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB531I   VPPSEQT FOR *bpname* MUST BE BETWEEN 0 AND 100

Explanation: An ALTER BUFFERPOOL command was issued to alter a buffer pool's VPPSEQT attribute to an invalid value. VPPSEQT is the parallel sequential steal threshold of the buffer pool. Valid values range from 0 to 100.

System action: The VPPSEQT value is not changed.

System programmer response: Change the VPPSEQT attribute to a valid value and reenter the command.

DSNB532I   VPXPSEQT FOR *bpname* HAS BEEN SET TO *integer*

Explanation: This message is issued from DSNB1ABP csect in response to the ALTER BUFFERPOOL command. It indicates the successful modification of the assisting parallel sequential threshold of the buffer pool.

DSNB533I   *integer* IS THE CURRENT VPXPSEQT FOR BUFFER POOL *bpname*

Explanation: An ALTER BUFFERPOOL command specified a VPXPSEQT equal to the current assisting parallel sequential threshold of the buffer pool.

System action: The assisting parallel sequential threshold of the buffer pool is not changed. Processing for other parameters, if requested, proceeds normally.

DSNB534I   VPXPSEQT FOR *bpname* MUST BE BETWEEN 0 AND 100

Explanation: An ALTER BUFFERPOOL command was issued to change a buffer pool's VPXPSEQT attribute to an invalid value. VPXPSEQT is the assisting parallel sequential threshold of the virtual buffer pool. Valid values range from 0 to 100. This message is issued from DSNB1ABP csect.

System action: The VPXPSEQT value is not changed.

System programmer response: Change the VPXPSEQT attribute to a valid value and reenter the command.

DSNB536I   *csect-name* THE TOTAL VIRTUAL BUFFER POOL STORAGE EXCEEDS THE REAL STORAGE
              CAPACITY

Explanation: This message is issued as a warning that the total amount of allocated virtual buffer pool storage for this DB2 subsystem has exceeded the real storage capacity of the z/OS image. The amount of real storage should be at least as much as the total amount of virtual buffer pool storage, and should also be enough to accommodate the EDM pool and DB2 working storage in addition to the virtual buffer pool storage. Paging activity in the buffers is an indication of a performance problem.

System action: Processing continues.

System programmer response: Allocate more real storage to the z/OS image. If there is not enough real storage to hold the buffers, then the number of buffers needs to be reduced. This might mean fewer concurrent users. If the number of buffers is reduced, then DB2 will issue the DSNB538I message when the total amount of virtual buffer pool storage no longer exceeds the real storage capacity.

DSNB538I   *csect-name* THE TOTAL BUFFER POOL STORAGE NO LONGER EXCEEDS THE REAL STORAGE
              CAPACITY

Explanation: This message is issued following a previous DSNB536I message to indicate that the total amount of allocated buffer pool storage for this DB2 subsystem no longer exceeds the real storage capacity of the z/OS image.

System action: Processing continues.

DSNB539I   *csect-name* AUTOLPL FOR *bpname* HAS BEEN SET TO *value*. IT WILL TAKE EFFECT ON THE NEXT
              ALLOCATION

Explanation: DB2 issues this message in response to an ALTER BUFFERPOOL command to ind that the requested change to the AUTOLPL parameter was accepted. The new AUTOLPL value takes effect the next time the buffer pool is allocated.

bpname is the name of the buffer pool.

value is one of the following:

YES   DB2 automatically recovers pages that are in LPL status.

NO   DB2 does not automatically recover pages in LPL status. To recover the LPL page, perform one of the
          following actions:

          • Issue the START DATABASE command with the SPACENAM option
          • Run the RECOVER or LOAD utility

System action: Processing continues

DSNB540I   *csect-name* **INSUFFICIENT LOCAL CACHE VECTOR STORAGE TO EXPAND BUFFER POOL**
bpname **TO SPECIFIED SIZE** *size1*. **BUFFER POOL SIZE IS NOW** *size2*.

Explanation: The specified buffer pool could not be expanded to the requested size (*size1*). The associated group
buffer pool is connected, and the local cache vector could not be expanded to cover the requested buffer pool size.

The local cache vector is a bit array which tracks whether the page in a given buffer was modified by another system
(cross-invalidated). There must be one bit in the vector for each buffer in the buffer pool. For more information on
the local cache vector, see *MVS/ESA Programming: Sysplex Services Reference*.

System action: DB2 downgrades the expansion request to the size that can be supported by the available local cache
vector storage (*size2*).

System programmer response: Free local cache vector storage by contracting or deleting other buffer pools that are
connected to a group buffer pool. Then reattempt the pool expansion by reissuing the ALTER BUFFERPOOL
command.

DSNB541I   *csect-name* **THE TOTAL ACTIVE PGFIX YES BUFFER POOL STORAGE EXCEEDS THE DB2**
ALLOWED REAL STORAGE CAPACITY WHEN ALLOCATING STORAGE FOR BUFFER POOL
bpname **REAL STORAGE CAPACITY =** *rsc* **MB.**

Explanation: When allocating storage for a PGFIX YES buffer pool, DB2 detects that the total storage allocated for
the PGFIX YES buffer pools will exceed 80% of the real storage capacity on this z/OS image. The buffer pool name
identified on this message (*bpname*) will be allocated with the PGFIX NO attribute. The real storage capacity of this
z/OS image (*rsc*) is displayed in this message in units of megabytes (1 MB = 2**20 bytes).

System action: The buffer pool is created using the PGFIX NO attribute.

System programmer response: Either allocate more real storage to this z/OS image, reduce the requested size of the
buffer pool or use the PGFIX NO attribute.

DSNB542I   *csect-name* **THE TOTAL ACTIVE PGFIX YES BUFFER POOL STORAGE EXCEEDS THE DB2**
ALLOWED REAL STORAGE CAPACITY WHEN EXPANDING STORAGE FOR BUFFER POOL
bpname **CURRENT VPSIZE =** *cbpsize* **NEW VPSIZE =** *nbpsize* **REAL STORAGE CAPACITY =** *rsc* **MB.**

Explanation: When expanding the VPSIZE for an active PGFIX YES buffer pool, DB2 detects that the total storage
allocated for all active PGFIX YES buffer pools will exceed 80% of the real storage capacity on this z/OS image. The
ALTER BUFFERPOOL command fails and the buffer pool will continue to be active with the current VPSIZE value.

The buffer pool attributes and real storage capacity are:

bpname
   The buffer pool name.

cbpsize
   The current VPSIZE for the buffer pool.

nbpsize
   The new VPSIZE that was specified on the ALTER BUFFERPOOL command.

rsc   The real storage capacity of this z/OS image in units of megabytes (1 MB = 2**20 bytes).

System action: The ALTER BUFFERPOOL command fails. The buffer pool continues to be active with the current
VPSIZE.

System programmer response: Either allocate more real storage to this z/OS image, reduce the requested size of the
buffer pool or use the PGFIX NO attribute.

DSNB543I   **THE PGFIX ATTRIBUTE IS ALTERED FOR BUFFER POOL** *bpname* **CURRENT ATTRIBUTE =** *cpgfix*
NEW ATTRIBUTE = *npgfix* **THE NEW ATTRIBUTE IS IN PENDING STATE.**

Explanation: This message is issued in response to an ALTER BUFFERPOOL command to indicate that the
requested change to the PGFIX attribute has been accepted. The new PGFIX value will take effect the next time the
buffer pool is allocated.

bpname
> The buffer pool name.

cpgfix   The current PGFIX value for the buffer pool.

npgfix   The new PGFIX value that was specified on the ALTER BUFFERPOOL command.

System action: Processing continues.

DSNB544I   **AUTOSIZE FOR** *bpname* **HAS BEEN SET TO** *nasize*

Explanation: This message is issued in response to an ALTER BUFFERPOOL command to indicate that the requested change to the AUTOSIZE attribute has been accepted.

bpname   The buffer pool name.

nasize   The new AUTOSIZE value specified in the ALTER BUFFERPOOL command, either YES or NO.

System action: Processing continues.

DSNB545I   **FOR** *bp-name*, **VPSIZE** *specified-value* **NOT VALID WITH VPSIZEMIN** *minimum-size* **AND VPSIZEMAX** *maximum-size*. **VPSIZE MUST BE BETWEEN VPSIZEMIN AND VPSIZEMAX**

Explanation: An ALTER BUFFERPOOL command was issued that specified incompatible values for one or more of VPSIZE, VPSIZEMAX, or VPSIZEMIN. The value of VPSIZE must be greater than or equal to the value of VPSIZEMIN, and less than or equal to the value of VPSIZEMAX.

bp-name
> The name of the buffer pool.

specified-value
> The value that is specified for VPSIZE in the ALTER BUFFER POOL command or if no value was specified, the current value.

minimum-size
> The value that is specified for VPSIZEMIN in the ALTER BUFFER POOL command or if no value was specified, the current value.

maximum-size
> The value that is specified for VPSIZEMAX in the ALTER BUFFER POOL command or if no value was specified, the current value.

System action: The buffer pool is not modified.

User response: Reissue the ALTER BUFFERPOOL command, and specify compatible values for the VPSIZE, VPSIZEMIN and VPSIZEMAX options.

Related reference:

↪ -ALTER BUFFERPOOL (DB2) (DB2 Commands)

DSNB546I   **PREFERRED FRAME SIZE** *preferred-size buffer-count* **BUFFERS USING** *frame-size* **FRAME SIZE ALLOCATED**

Explanation: This message is a section of output from the DISPLAY BUFFERPOOL command, which is part of message "DSNB401I" on page 68.

DSNB551I   **BSDS READ ERROR, RC=** *integer*, **RSN=** *integer*

Explanation: This message is issued in response to a failure of a READ function on the bootstrap data set (BSDS). Users should check any previous error messages issued by MVS and DB2 against the corresponding BSDS. RC and RSN indicate an error return code and reason code from the BSDS function.

System action: Processing continues.

Operator response: Notify the system programmer.

System programmer response: Analyze the return and reason codes to determine what type of error occurred. Refer to message DSNJ107I for more information.

DSNB552I   **BSDS WRITE ERROR, RC=** *integer***, RSN=** *integer*

Explanation: This message is issued in response to a failure of a WRITE function on the bootstrap data set (BSDS). Check any previous error messages issued by MVS and DB2 against the corresponding BSDS. RC and RSN indicate an error return code and reason code from the BSDS function.

System action: Processing continues.

Operator response: Notify the system programmer.

System programmer response: Analyze the return and reason codes to determine what type of error occurred. Refer to message DSNJ108I for more information.

DSNB553I   **BSDS INSERT ERROR, RC=** *integer***, RSN=** *integer*

Explanation: This message is issued in response to a failure of an INSERT function on the bootstrap data set (BSDS). Check any previous error messages issued by MVS and DB2 against the corresponding BSDS. RC and RSN indicate an error return code and reason code from the BSDS function.

System action: Processing continues.

Operator response: Notify the system programmer.

System programmer response: Analyze the return and reason codes to determine what type of error occurred. For more information, refer to message DSNJ109I or other DSNJ messages related to BSDS.

DSNB555I   **WLM RECOMMENDATION TO ADJUST SIZE FOR BUFFER POOL** *bpname* **HAS COMPLETED OLD SIZE =** *csize* **BUFFERS NEW SIZE =** *nsize* **BUFFERS**

Explanation: This message is issued when Workload Manager (WLM) notifies DB2 to adjust the size of a buffer pool. This recommendation is made based on WLM's dynamic monitoring of the effects of buffer pool I/O on the achievement of workload goals, and also on the amount of available storage on the system. This message is issued after the adjustment has completed.

bpname   The buffer pool name.

csize   The old size of the buffer pool.

nsize   The new size of the buffer pool.

System action: Processing continues.

DSNB601I   **BUFFER POOL** *bpname* **FULL**

Explanation: A buffer is required to access a database page that is not currently contained within the buffer pool. A buffer cannot be assigned because all buffers in the buffer pool are in a nondisplaceable state. (Buffers are in a nondisplaceable state when they contain actively referenced pages or updated pages for which writes are pending.)

This message can occur if there is an overload of concurrent activity for the buffer pool size. The variable *bpname* is the buffer pool external name.

System action: Reason code 00C200E9 and the buffer pool external name are recorded in the cursor table (CT), which is available in the SQL communication area (SQLCA) or messages. An SQLCODE -904 (resource not available) is returned to the user.

Operator response: Notify the system programmer.

System programmer response: Do one of the following:

• Reduce concurrent activity by reducing the number of DB2 threads.

• Use the ALTER BUFFERPOOL command to increase the size of the buffer pool.

DSNB602I   **UNABLE TO CREATE BUFFER POOL** *bpname***. BUFFER POOL SIZE IS ZERO**

Explanation: An attempt to create the buffer pool failed while opening a DB2 table space or index that specified its use. The buffer pool size for *bpname* was zero. This indicates that the use of the pool is invalid.

System action: Reason code 00C200E0 and the buffer pool external name are recorded in the cursor table (CT),

which is available in the SQL communication area (SQLCA) or messages. An SQLCODE -904 (resource not available) is returned to the user.

Operator response: Notify the system programmer.

System programmer response: Use the ALTER TABLESPACE or ALTER INDEX statement to modify affected table spaces or indexes to indicate a usable buffer pool. Use the DISPLAY BUFFERPOOL command to determine a usable buffer pool. If necessary, use the ALTER BUFFERPOOL command to activate the failing buffer pool by setting its VPSIZE to a nonzero value.

DSNB603I   **INSUFFICIENT STORAGE TO CREATE/EXPAND BUFFER POOL** *bpname* **TO SPECIFIED SIZE** *integer*. **BUFFER POOL SIZE IS NOW** *integer*.

Explanation: A buffer pool could not be created or expanded to the requested size. Insufficient storage exists within the *ssnm*DBM1 address space to build the buffer pool. The buffer pool size is set by the ALTER BUFFERPOOL command and can be queried by using the DISPLAY BUFFERPOOL command.

System action: The buffer pool is created or expanded to the size allowed by the available storage. If the buffer pool cannot be created at all, and the creation request was triggered by a page set open by an application, then an SQLCODE -904 (resource not available) is returned to the user. Reason code 00C200E8 and the buffer pool external name are recorded in the cursor table (CT), which is available in the SQL communication area (SQLCA) or messages.

Operator response: Notify the system programmer.

System programmer response: Analyze the storage content of the *ssnm*DBM1 address space to determine further action. If appropriate, use the ALTER TABLESPACE or ALTER INDEX statement to use a different buffer pool. Otherwise, use ALTER BUFFERPOOL commands to balance storage use for all buffer pools.

DSNB605I   **UNABLE TO CREATE BUFFER POOL** *bpname* **TO SPECIFIED SIZE DURING STARTUP/RESTART. INSUFFICIENT VIRTUAL STORAGE**

Explanation: An attempt to create the buffer pool failed at the time that DB2 was performing its startup or restart processing. Insufficient virtual storage exists within the *ssnm*DBM1 address space for building the buffer pool. The name of the buffer pool is *bpname*.

System action: The buffer pool is not created. The table space or index open request, which triggered the attempt to create the buffer pool, fails. An SQLCODE -904 (resource not available) is returned to the user. Reason code 00C200E8 and the buffer pool external name are recorded in the cursor table (CT), which is available in the SQL communication area (SQLCA) or messages.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer. This message remains on the console until the operator deletes it.

System programmer response: Analyze the virtual storage content of the DSNDBM1 *ssnm*DBM1 address space to determine further action. If appropriate, use ALTER BUFFERPOOL commands to balance virtual storage use for all buffer pools.

DSNB609I   **VIRTUAL BUFFER POOL** *bpname* **CREATED USING DEFAULT SIZE DURING STARTUP/RESTART. VIRTUAL BUFFER POOL SIZE IS ZERO**

Explanation: While activating a buffer pool during DB2 startup or restart processing, the buffer pool size was zero. A zero value might be caused by an unusable bootstrap data set (BSDS) or by a DB2 abnormal termination after buffer pool size was altered to zero. The name of the buffer pool is *bpname*.

System action: DB2 creates the buffer pool using the startup restart default value.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer. This message remains on the console until deleted by the operator.

System programmer response: Use the ALTER TABLESPACE or ALTER INDEX statement to modify the affected table spaces or indexes to indicate a usable buffer pool. To determine a usable buffer pool, use the DISPLAY BUFFERPOOL command. If necessary, use the ALTER BUFFERPOOL command to activate the failing buffer pool by setting the VPSIZE to a nonzero value.

DSNB610I   *csect-name* **MAXIMUM ALLOCATABLE BUFFER POOL STORAGE OF** *int1***MB HAS BEEN REACHED. UNABLE TO CREATE/EXPAND BUFFER POOL** *bp-name* **TO SPECIFIED SIZE** *int2***. BUFFER POOL IS NOW** *int3*

Explanation: A buffer pool could not be created or expanded to the requested size. The maximum amount of allowable buffer pool storage, indicated by *int1*, has been reached. The maximum allowable buffer pool storage is calculated to be twice the amount of real storage that is available on the z/OS image, or 1TB, whichever is less.

bpname
> Indicates the name of the buffer pool that was affected.

int2   Indicates the size of the buffer pool that was requested.

int3   Indicates the actual size of the buffer pool that was created/expanded.

System action: The buffer pool is created or expanded to the size allowed by the available storage within the maximum. If the maximum has already been reached, then the buffer pool is allocated at a minimum size as follows:

- 2,000 pages for 4K page size
- 1,000 pages for 8K page size
- 500 pages for 16K page size
- 250 pages for 32K page size

Once the maximum has been reached, buffer pool sizes are not allowed to be increased.

System programmer response: Analyze the buffer pool sizes and determine if any can be decreased, or consider adding more memory.

DSNB740I   *csect-name* **ATTEMPTING TO ESTABLISH DUPLEXING FOR GROUP BUFFER POOL** *gbpname* **REASON =** *reason*

Explanation: DB2 has been requested to establish duplexing for the named group buffer pool. *reason* indicates the reason why duplexing was initiated. *reason* can be one of the following:

OPERATOR
> The operator initiated a SETXCF command to initiate duplexing.

POLICY
> The currently active CFRM policy indicated that duplexing should be automatically established for the group buffer pool.

This message is issued by each DB2 member participating in establishing duplexing.

System action: The DB2 members that are actively connected to the named group buffer pool participate in establishing duplexing for the group buffer pool. Access to the group buffer pool is quiesced while duplexing is being established.

DSNB741I   *csect-name* **DUPLEXING COULD NOT BE ESTABLISHED FOR GROUP BUFFER POOL** *gbpname* **REASON =** *reason*

Explanation: DB2 was unable to establish duplexing for the named group buffer pool. *reason* indicates the DB2 reason code indicating why duplexing could not be established.

System action: The group buffer pool goes back to simplex mode.

System programmer response: See the explanation for the reason code given.

DSNB742I   *csect-name* **DUPLEXING HAS BEEN SUCCESSFULLY ESTABLISHED FOR GROUP BUFFER POOL** *gbpname*

Explanation: The process to establish duplexing for the named group buffer pool has successfully completed.

This message is preceded by message DSNB740I which indicates the reason why duplexing was initiated. This message is issued by each DB2 member participating in establishing duplexing.

System action: Processing of mainline requests is resumed for the group buffer pool. Writes of changed pages to the

group buffer pool will be duplexed writes from this point forward.

DSNB743I   *csect-name* **DUPLEXING IS BEING STOPPED FOR GROUP BUFFER POOL** *gbpname* **FALLING BACK TO PRIMARY REASON =** *reason1* **DB2 REASON CODE =** *reason2*

Explanation: Duplexing is being stopped for the named group buffer pool. To stop the duplexing, the secondary group buffer pool will be deallocated, and DB2 will go back to simplex mode on the primary group buffer pool.

reason1 indicates why duplexing was stopped:

OPERATOR
>   The operator initiated an MVS command to stop duplexing.

POLICY
>   A new CFRM policy has been activated which specifies DUPLEX(DISABLED) for the group buffer pool.

LOSSCONNSEC
>   One or more of the DB2 members lost connectivity to the secondary group buffer pool structure.

STRFAILSEC
>   The secondary group buffer pool structure failed.

DB2REASON
>   DB2 decided to stop duplexing. The reason code given in *reason2* indicates why DB2 stopped duplexing.

DUPCONNCHG
>   Duplexing rebuild stopped because of insufficient connectivity due to a change in the set of connectors.

If *reason1* indicates 'DB2REASON', then *reason2* specifies a DB2 reason code that indicates why DB2 stopped duplexing. If *reason1* is not 'DB2REASON', then *reason2* is zero.

This message is issued by each DB2 member connected to the duplexed group buffer pool.

System action: The secondary group buffer pool structure is discarded, and DB2 enters back into simplex mode on the primary group buffer pool structure. If DUPLEX(AUTO) is specified in the active CFRM policy for the group buffer pool, then duplexing will be automatically reinitiated, if duplexing is still possible.

System programmer response: If the rebuild stopped because of "DB2REASON", see the explanation for the reason code given. If duplexing is not automatically reinitiated and duplexing is needed for the group buffer pool, then use the MVS SETXCF command to reestablish duplexing.

DSNB744I   *csect-name* **DUPLEXING IS BEING STOPPED FOR GROUP BUFFER POOL** *gbpname*, **SWITCHING TO SECONDARY REASON =** *reason1* **DB2 REASON CODE =** *reason2*

Explanation: Duplexing is being stopped for the named group buffer pool. To stop the duplexing, the primary group buffer pool will be deallocated, and DB2 will switch to the secondary group buffer pool to go back to simplex mode.

reason1 indicates why duplexing was stopped:

OPERATOR
>   The operator initiated an MVS command to stop duplexing.

POLICY
>   A new CFRM policy has been activated which specifies DUPLEX(DISABLED) for the group buffer pool.

LOSSCONNPRI
>   One or more of the DB2 members lost connectivity to the primary group buffer pool structure.

STRFAILPRI
>   The primary group buffer pool structure failed.

DB2REASON
>   DB2 decided to stop duplexing. The reason code given in *reason2* indicates why DB2 stopped duplexing.

If *reason1* indicates 'DB2REASON', then *reason2* specifies a DB2 reason code that indicates why DB2 stopped duplexing. If *reason1* is not 'DB2REASON', then *reason2* is zero.

This message is issued by each DB2 member connected to the duplexed group buffer pool.

System action: The primary group buffer pool structure is discarded, and DB2 enters back into simplex mode on the secondary group buffer pool structure. If DUPLEX(ENABLED) is specified in the active CFRM policy for the group buffer pool, then duplexing will be automatically reinitiated, if duplexing is still possible.

System programmer response: If the rebuild stopped because of "DB2REASON", see the explanation for the reason code given. If duplexing is not automatically reinitiated and duplexing is needed for the group buffer pool, then use the MVS SETXCF command to reestablish duplexing.

DSNB745I   *csect-name* **THE TRANSITION BACK TO SIMPLEX MODE HAS COMPLETED FOR GROUP BUFFER POOL** *gbpname*

Explanation: The named group buffer pool has completed the transition from duplex mode back to simplex mode.

This message is preceded either by message DSNB743I or by message DSNB744I which indicates the reason why duplexing was stopped. This message is issued by each DB2 member connected to the group buffer pool.

System action: The group buffer pool is now back in simplex mode which means that writes of changed pages to the group buffer pool are now simplexed. If DUPLEX(ENABLED) is specified in the active CFRM policy for the group buffer pool, then duplexing will be automatically reinitiated, if duplexing is still possible. This would mean that the group buffer pool is only temporarily back into simplex mode (it is soon to be entered back into duplex mode).

DSNB749I   **GBPCACHE NO IS NOT ALLOWED FOR GROUP BUFFER POOL** *gbpname* **BECAUSE IT IS DUPLEXED**

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that requests a change of the GBPCACHE attribute from YES to NO, but because the group buffer pool is duplexed, the change from YES to NO is not allowed.

System action: The request to change from GBPCACHE(YES) to GBPCACHE(NO) is rejected.

System programmer response: GBPCACHE(NO) and duplexing are mutually exclusive. If you want GBPCACHE(NO), then you must put the group buffer in simplex mode before specifying GBPCACHE(NO). To put a duplexed group buffer pool in simplex mode, do any of the following procedures:

- Activate a new CFRM policy that specifies DUPLEX(DISABLED) for the group buffer pool. This is the preferred method if you do not intend to duplex the group buffer pool again for awhile.
- Ensure that DUPLEX(AUTO) is not specified in the active CFRM policy for the group buffer pool, and then issue the MVS command SETXCF STOP,REBUILD,DUPLEX.

DSNB750I   **DISPLAY FOR GROUP BUFFER POOL** *gbpname* **FOLLOWS**

Explanation: This message is issued in response to a a DISPLAY GROUPBUFFERPOOL command. It signals the beginning of a display for group buffer pool *gbpname* if the following are true:

- It meets the selection criteria on the command.
- The MVS CFRM administrative policy information for the group buffer pool is accessible.

If the CFRM policy information is unavailable, message DSNB751I signals the partial display of group buffer pool information.

The series of messages DSNB755I through DSNB759I provide information on the group buffer pool's allocation characteristics. Messages DSNB765I through DSNB769I provide information on DB2 connections to the group buffer pool, while DSNB770I through DSNB788I provide statistics on group buffer pool usage. Note that all message numbers in the range might not be used. Message DSNB790I ends the display.

DSNB751I   **MVS CFRM POLICY INFORMATION NOT AVAILABLE FOR GROUP BUFFER POOL** *gbpname* **MVS IXCQUERY RETURN CODE =** *retcode* **MVS IXCQUERY REASON CODE =** *reason* **A PARTIAL DISPLAY FOLLOWS**

Explanation: DB2 could not retrieve group buffer pool policy information in response to a DISPLAY GROUPBUFFERPOOL command. Either the MVS query request (IXCQUERY) did not successfully complete, or the backing structure for the group buffer pool is not defined in the MVS CFRM administrative policy. To determine the

exact reason, examine return code *retcode* and reason code *reason* returned by MVS on the query request. For an explanation of the IXCQUERY codes, see *MVS/ESA Programming: Sysplex Services Reference*. If the IXCQUERY request did not complete successfully, no CONNLIST information, even if requested, is provided.

System action: Processing continues, but the information displayed is incomplete.

DSNB752I   GROUP DETAIL STATISTICS ARE NOT AVAILABLE FOR THE GROUP BUFFER POOL BECAUSE A REBUILD IS IN PROGRESS

Explanation: Because a rebuild is in progress for the group buffer pool, DB2 cannot retrieve group buffer pool group (GDETAIL) statistics in response to a DISPLAY GROUPBUFFERPOOL command. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

System action: Processing continues.

DSNB753I   GROUP BUFFER POOL STRUCTURE OWNER INFORMATION NOT AVAILABLE IRLM NOTIFY RETURN CODE = *retcode* IRLM NOTIFY REASON CODE = *reason*

Explanation: DB2 could not retrieve group buffer pool structure owner information for message DSNB798I because the IRLM Notify request did not successfully complete. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

retcode   Return code from IRLM on the Notify request

reason   Reason code from IRLM on the Notify request

System action: Processing continues.

Related concepts:

➥ IRLM return and reason codes (IRLM)

Related information:

"DSNB750I" on page 92
"DSNB751I" on page 92

DSNB755I   DB2 GROUP BUFFER POOL STATUS CONNECTED = *status1* CURRENT DIRECTORY TO DATA RATIO = *ratio1* PENDING DIRECTORY TO DATA RATIO = *ratio2* CURRENT GBPCACHE ATTRIBUTE = *gcache1* PENDING GBPCACHE ATTRIBUTE = *gcache2*

Explanation: In response to a DISPLAY GROUPBUFFERPOOL command, this is the first of two messages displaying information about the status of the group buffer pool as it is known to DB2. This message is followed by DSNB756I. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

status1   Whether this DB2 is connected to the group buffer pool:

  YES   This DB2 is actively connected to the group buffer pool.

  NO   This DB2 is not actively connected to the group buffer pool.

ratio1   The ratio of directory entries to data pages, as specified for the current allocation of this group buffer pool.

ratio2   The requested ratio of directory entries to data pages to take effect at the next reallocation of the group buffer pool.

gcache1   The GBPCACHE attribute as specified for the current allocation of this group buffer pool.

gcache2   The value for GBPCACHE attribute that will take effect at the next reallocation of the group buffer pool.

DSNB756I   CLASS CASTOUT THRESHOLD = *class-threshold1*,*class-threshold2* GROUP BUFFER POOL CASTOUT THRESHOLD = *gbp-threshold* % GROUP BUFFER POOL CHECKPOINT INTERVAL = *interval* MINUTES RECOVERY STATUS = *status1* AUTOMATIC RECOVERY = *auto-recovery*

Explanation: This is the second of two messages that display information about the status of the group buffer pool as it is known to DB2. If DB2 was successful in getting MVS CFRM policy information for the group buffer pool, messages DSNB757I through DSNB759I immediately follow this message. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

DSNB757I

class-threshold1

> The threshold at which class castouts are to be initiated, expressed as a percentage of the group buffer pool size.

class-threshold2

> The threshold at which class castouts are to be initiated, expressed as an absolute number of buffers. *class-threshold2* is used to determine the threshold only if *class-threshold1* is zero and *class-threshold2* is non-zero. Otherwise, *class-threshold1* is used to determine the threshold.

gbp-threshold

> The threshold at which castout is to be initiated at the group buffer pool level. It is expressed as a percentage of the group buffer pool size.

interval   The time, in minutes, between successive initiations of group buffer pool checkpoint.

status1   The recovery status for the group buffer pool:

> **DAP**   Damage assessment pending. DB2 is in the process of doing damage assessment for the group buffer pool in response to a coupling facility structure failure. The group buffer pool is temporarily stopped internally by DB2 and is unavailable for use until the damage assessment process completes.
>
> **NORMAL**
> > There is no pending recovery action for this group buffer pool.

auto-recovery

> Automatic recovery parameter specification:

> **YES**   DB2 should automatically initiate recovery for the objects that use this group buffer pool following a failure of the group buffer pool, This is the default.
>
> **NO**   DB2 should not initiate automatic group buffer pool recovery. If the group buffer pool fails, you must manually issue DB2 START DATABASE commands to do the recovery.

System action: Processing continues.

User response: No action is required.

Related reference:

[→] -START DATABASE (DB2) (DB2 Commands)

DSNB757I   **MVS CFRM POLICY STATUS FOR** *structure-name* = *status2* **MAX SIZE INDICATED IN POLICY =** *psize* **KB DUPLEX INDICATOR IN POLICY =** *dupstat* **CURRENT DUPLEXING MODE =** *dupmode* **ALLOCATED =** *status1*

Explanation: For a DISPLAY GROUPBUFFERPOOL command, this message gives the status for the structure associated with the group buffer pool. It is displayed only if DB2 was successful in querying the MVS CFRM administrative policy. It is immediately preceded by message DSNB756I, and if the structure is allocated, followed by messages DSNB758I and DSNB759I. A preceding DSNB750I message identifies the associated group buffer pool.

structure-name

> The coupling facility structure name for this group buffer pool, such as DSNCAT_GBP0.

status2   MVS global status of the backing coupling facility structure:

> **NORMAL**
> > Structure is defined in the current CFRM policy and has no current MVS restrictions.
>
> **CHNGP**
> > Change pending in the structure policy.

dupstat   The duplexing attribute for the coupling facility structure, as indicated by the DUPLEX parameter in the CFRM policy definition.

> **DISABLED**
> > DUPLEX(DISABLED) is specified for the coupling facility structure. The DUPLEX parameter is not specified.

94   Messages

ENABLED
> DUPLEX(ENABLED) is specified for the coupling facility structure.

ALLOWED
> DUPLEX(ALLOWED) is specified for the coupling facility structure.

dupmode
> Indication of current status of the group buffer pool in relation to duplexing.

SIMPLEX
> Group buffer pool is currently in simplex mode.

DUPLEX
> Group buffer pool is currently in duplex mode.

INTRAN
> Group buffer pool is currently in a transition state. Either from simplex mode to duplex mode, or from duplex mode to simplex mode. The rebuild status indicator in message DSNB758I gives information about the transition state.

psize The size (in kilobytes) of the coupling facility structure as indicated in the CFRM policy definition.

status1 Whether the backing coupling facility structure is currently allocated:

> **YES** The coupling facility structure is allocated in a coupling facility.

> **NO** The coupling facility structure is not allocated in any coupling facility.

DSNB758I   **ALLOCATED SIZE =** *asize* **KB VOLATILITY STATUS =** *volstat* **REBUILD STATUS =** *rbldstat*
  CFNAME = *cfname* **CFLEVEL - OPERATIONAL =** *cflevel1* **CFLEVEL - ACTUAL =** *cflevel2*

Explanation: This message is the first of two messages giving the allocation details for the group buffer pool on a DISPLAY GROUPBUFFERPOOL command. It is issued only if DB2 determines, by successfully querying the MVS CFRM policy, that the group buffer pool is allocated. It is immediately preceded by message DSNB757I. A preceding DSNB750I message identifies the associated group buffer pool.

asize The actual allocated group bufferpool structure size (in kilobytes).

volstat The volatility status of the backing group buffer pool structure for this group buffer pool:

VOLATILE
> The backing coupling facility structure is volatile.

NON-VOLATILE
> The backing coupling facility structure is non-volatile.

rbldstat Indication of whether or not rebuild is in progress for the group buffer pool structure, and if so, which part of the rebuild process is happening:

NONE Rebuild is not in progress.

QUIESCE
> Rebuild is in progress and the rebuild is currently in the rebuild quiesce phase.

CONNECT
> Rebuild is in progress and the rebuild is currently in the rebuild connect phase.

CLEANUP
> The rebuild has completed, and the connected DB2 members are making final preparations to use the new group buffer pool structure. If the duplexing status in message DSNB757I indicates 'INTRAN', then the 'CLEANUP' indication means that DB2 is in the final stages of transitioning from duplex mode back to simplex mode, switching to the secondary group buffer pool.

STOPPING
> The rebuild is stopping, and the connected DB2 members are making preparations to fall back to the original group buffer pool structure. If the duplexing status in message DSNB757I indicates 'INTRAN', then the 'STOPPING' indication means that DB2 is transitioning from duplex mode back to simplex mode, reverting back to the primary group buffer pool.

DUPLEXED

The group buffer pool is in steady-state duplexing mode.

SWITCHING

DB2 is in the initial phase of transitioning from duplex mode back to simplex mode, switching to the secondary group buffer pool. The group buffer pool is transitioning from duplex mode back to simplex mode, switching to the secondary group buffer pool.

cfname The name of the coupling facility in which the group buffer pool is allocated. If the group buffer pool is duplexed, this is the coupling facility name associated with the primary group buffer pool.

cflevel1 The coupling facility level from a DB2 functionality perspective for the coupling facility in which this group buffer pool is allocated. If the group buffer pool is duplexed, *cflevel1* is the coupling facility level associated with the primary group buffer pool.

cflevel2 The coupling facility level for the coupling facility in which this group buffer pool is allocated. If the group buffer pool is duplexed, *cflevel2* is the coupling facility level associated with the primary group buffer pool."

DSNB759I **NUMBER OF DIRECTORY ENTRIES =** *dir#* **NUMBER OF DATA PAGES =** *data#* **NUMBER OF CONNECTIONS =** *conn#*

Explanation: This message is the second of two messages giving the allocation details for the group buffer pool for a DISPLAY GROUPBUFFERPOOL command. It is issued only if DB2 determines, by successfully querying the MVS CFRM administrative policy, that the group buffer pool is allocated. It is immediately preceded by message DSNB758I. A preceding DSNB750I message identifies the associated group buffer pool.

dir# Actual number of directory entries allocated for the coupling facility structure.

data# Actual number of data pages allocated for the coupling facility structure.

conn# The number of DB2 subsystems that are connected to the coupling facility structure. This includes both active connections and failed-persistent connections. The CONNLIST option of the DISPLAY GROUPBUFFERPOOL command gives details of the connection status.

DSNB760I **GBPCACHE OPTION SET TO** *value* **FOR** *gbpname*. **IT WILL TAKE EFFECT ON THE NEXT REALLOCATION**

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that requests a change in the GBPCACHE attribute. It is also issued for the first ALTER GROUPBUFFERPOOL command for a group buffer pool, even when the GBPCACHE option is not specified on the command. (In this case, the message shows YES for the GBPCACHE option.) The new setting takes effect the next time the coupling facility cache structure is allocated.

System action: Processing continues.

System programmer response: Use the MVS command SETXCF START, REBUILD command when you are ready for the change to take effect.

DSNB761I **THE NEW SETTING WILL TAKE EFFECT AFTER THE GBPCACHE ATTRIBUTE HAS BEEN CHANGED TO YES**

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that requests a change to RATIO, CLASST, GBPOOLT or GBPCHKPT when GBPCACHE NO is also specified on the command or when the current GBPCACHE attribute is NO. The new setting takes effect the next time the coupling facility cache structure is allocated after the GBPCACHE attribute has been changed to YES.

System action: Processing continues.

System programmer response: Use the ALTER GROUPBUFFERPOOL command to change the GBPCACHE attribute to YES and the MVS command SETXCF START, REBUILD when you are ready for the change to take effect.

DSNB762I   DUPLEXING STATISTICS FOR *secondary-gbpname* **WRITES FAILED DUE TO LACK OF STORAGE = *writes-full* CHANGED**

Explanation: This message is displayed as a result of the DISPLAY GROUP BUFFERPOOL command with the GDETAIL option if the group buffer pool is duplexed. It lists the statistics on writes to the secondary group buffer pool and the total number of changed pages for the secondary group buffer pool.

secondary-gbpname
>   The normal group buffer pool name (such as GBP0) suffixed with '-SEC'.

writes-full
>   The write miss cache full counter. Number of times a coupling facility write request was not completed because of a lack of storage in the secondary group buffer pool.

Related concepts:

▐▶ How DB2 writes from the group buffer pool to disk (DB2 Data Sharing Planning and Administration)

▐▶ Coupling facility storage considerations for duplexing (DB2 Installation and Migration)

Related reference:

▐▶ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

▐▶ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:
"DSNB750I" on page 92

DSNB763I   *gbpname* **IS NOT DEFINED WITH GBPCACHE NO**

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with TYPE(NOCACHE) and either one name or a list of names specified in the selection list. The specified group buffer pool is not defined with GBPCACHE NO, and therefore does not meet the qualification to be displayed. This message is issued once for each group buffer pool that does not match the "GBPCACHE NO" criterion.

System action: The display is terminated for this group buffer pool. Command processing continues, if there are other group buffer pool names to process.

DSNB764I   DUPLEXING STATISTICS FOR *secondary-gbpname* **WRITES FAILED DUE TO LACK OF STORAGE = *s2* ASYNC COMPLETION CHECKS = *s3***

Explanation: This message is displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option if the group buffer pool is duplexed. It lists the statistics on writes to the secondary group buffer pool. *secondary-gbpname* is the normal group buffer pool name (e.g. GBP0) suffixed with '-SEC'.

s2
>   Coupling facility write requests that were not completed because of a lack of storage in the secondary group buffer pool.

s3
>   The number of times DB2 checked for the completion of the write of a changed page to the secondary group buffer pool, but the write had not yet completed and thus the transaction had to wait until the write to the secondary group buffer pool completed.

System action: Processing continues.

User response: No action is required.

Related reference:

▐▶ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

▐▶ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

Related information:
"DSNB750I" on page 92

DSNB765I   NO CONNLIST REPORT IS GENERATED. THERE ARE NO CONNECTIONS TO THIS GROUP
                 BUFFER POOL.

Explanation: DB2 displays this message in response to a DISPLAY GROUPBUFFERPOOL command with the
CONNLIST option when there are currently no connection names to report for the group buffer pool. A preceding
DSNB750I message identifies the associated group buffer pool.

DSNB766I   THE CONNLIST REPORT FOLLOWS

Explanation: DB2 displays this message in response to a DISPLAY GROUPBUFFERPOOL command with the
CONNLIST option to indicate the beginning of the CONNLIST display for this group buffer pool. A preceding
DSNB750I message identifies the associated group buffer pool.

DSNB767I   CONNECTION NAME = *conname*, **CONNECTION STATUS =** *status* **CONNECTOR'S RELEASE =**
                 DB2 release

Explanation: This message contains the display information that results from a DISPLAY GROUPBUFFERPOOL
command with the CONLIST option. One DSNB767I message is issued for each defined connection to the group
buffer pool. The associated group buffer pool is identified on a preceding DSNB750I message.

conname
        Name of the connection. DB2 subsystems use the DB2 member name as the connection name.

status   Status of the connection.

   A   Active.

   F   Failed-Persistent.

   D   Dually connected to both the original structure and the new structure during a rebuild.

DB2 release
        The release of the connected DB2.

DSNB768I   THERE ARE NO GROUP BUFFER POOLS DEFINED WITH GBPCACHE NO

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with the
TYPE(NOCACHE) option and an asterisk (*) specified as the selection list when DB2 determines that there are no
group buffer pools defined with the GBPCACHE(NO) attribute.

System action: Command processing is terminated.

DSNB769I   THE CONNLIST REPORT IS COMPLETE

Explanation: DB2 displays this message in response to a DISPLAY GROUPBUFFERPOOL command with the
CONNLIST option to indicate the end of the CONNLIST display for this group buffer pool. A preceding DSNB750I
message identifies the associated group buffer pool.

DSNB770I   MEMBER DETAIL STATISTICS ARE NOT AVAILABLE FOR THE GROUP BUFFER POOL. THIS
                 DB2 HAS NEVER BEEN CONNECTED TO THE GROUP BUFFER POOL.

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with MDETAIL
specified. MDETAIL statistics are not available for the requested group buffer pool because this DB2 has not
connected to the group buffer pool since startup. A preceding DSNB750I or DSNB751I message identifies the
associated group buffer pool.

System action: Processing continues.

DSNB771I   INCREMENTAL MEMBER DETAIL STATISTICS SINCE *basetime*

Explanation: This message is displayed in response to the DISPLAY GROUPBUFFERPOOL command with
MDETAIL(INTERVAL) specified. It indicates the start time of the interval for which the statistics were accumulated.
This is the time of the previous MDETAIL incremental display on this DB2 or the time DB2 first connected to the

group buffer pool if there was no previous MDETAIL incremental display. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

System action: Processing continues.

DSNB772I   **CUMULATIVE MEMBER DETAIL STATISTICS SINCE** *basetime*

Explanation: This message is displayed in response to the DISPLAY GROUPBUFFERPOOL command with MDETAIL(*) specified. It indicates the start time of the interval for which the statistics were accumulated. This is the time this DB2 first connected to the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

System action: Processing continues.

DSNB773I   **MEMBER DETAIL STATISTICS SYNCHRONOUS READS DUE TO BUFFER INVALIDATION DATA RETURNED =** *r1* **DATA NOT RETURNED =** *r2*

Explanation: This is the first of eight messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists some synchronous read statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

The counter descriptions are listed below. In the descriptions, the term "synchronous" means that the coupling facility interaction is done under the DB2 user's execution unit.

r1   Synchronous coupling facility read requests issued because the virtual buffer was marked "invalid". Data is successfully returned from the group buffer pool.

r2   Synchronous coupling facility read requests issued because the virtual buffer was marked "invalid". Data is *not* returned from the group buffer pool.

DSNB774I   **DUE TO DATA PAGE NOT IN BUFFER POOL DATA RETURNED =** *r3* **DATA NOT RETURNED =** *r4*

Explanation: This is the second of eight messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists more local synchronous read statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

The counter descriptions are listed below. In the descriptions, the term "synchronous" means that the coupling facility interaction is done under the DB2 user's execution unit.

r3   Synchronous coupling facility read requests issued because the required data was not in the buffer pool buffer. Data is successfully returned from the group buffer pool.

r4   Synchronous coupling facility read requests issued because the required data was not in the buffer pool buffer. Data is *not* returned from the group buffer pool.

DSNB776I   **SYNCHRONOUS WRITES CHANGED PAGES =** *w1* **CLEAN PAGES =** *w2*

Explanation: This is the fifth of eight messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists some local write statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

The counter descriptions are listed below. In the descriptions, the term "synchronous" means that the coupling facility interaction is done under the DB2 user's execution unit.

w1   Number of changed pages that were written synchronously, under a user's execution unit, to the coupling facility.

w2   Number of clean pages that were written synchronously, under a user's execution unit, to the coupling facility

---

| **DSNB777I** **ASYNCHRONOUS WRITES CHANGED PAGES =** *changed-pages-count* **CLEAN PAGES =**
| *clean-pages-count* **FAILED DUE TO LACK OF STORAGE =** *failed-writes-count* **WRITE-AROUND**
| **PAGES=** *write-around-count*

| **Explanation:** This message is the sixth of eight messages that are displayed as a result of the DISPLAY
| GROUPBUFFERPOOL command with the MDETAIL option. It lists local write and cross-invalidation statistics for the
| group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

| The following list describes the counters in the message. In the descriptions, the term "asynchronous" means that the
| coupling facility interaction is done under a DB2 system execution unit, such as with a prefetch.

| *changed-pages-count*
| The number of changed pages that were written asynchronously, under a system execution unit, to the
| coupling facility

| *clean-pages-count*
| The number of clean pages that were written asynchronously, under a system execution unit, to the coupling
| facility.

| *failed-writes-count*
| The number of coupling facility write requests that were not completed because of a lack of storage in the
| group buffer pool.

| *write-around-count*
| The number of changed pages that are written to disk through group-buffer-pool write-around protocol.

| **System action:** Processing continues.

| **User response:** No action is required.

| **Related reference**:

| � Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning
| and Administration)

| � -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

| **Related information**:
| "DSNB750I" on page 92
| "DSNB751I" on page 92

DSNB778I **CASTOUT THRESHOLDS DETECTED FOR CLASSES =** *t1* **FOR GROUP BUFFER POOL =** *t2* **GBP**
CHECKPOINTS TRIGGERED = *k1* **PARTICIPATION IN REBUILD =** *k2*

Explanation: This is the seventh of eight messages containing local statistics, displayed as a result of the DISPLAY
GROUPBUFFERPOOL command with the MDETAIL option. It lists local castout threshold statistics for the group
buffer pool and also group buffer pool checkpoint and rebuild statistics. A preceding DSNB750I or DSNB751I
message identifies the associated group buffer pool.

Counters:

t1 Number of times the class castout threshold was detected

t2 Number of times the group buffer pool castout threshold was detected

k1 Number of group buffer pool checkpoints that were triggered.

k2 Number of times that this member participated in a rebuild of the group buffer pool structure.

DSNB780I **GROUP DETAIL STATISTICS ARE NOT AVAILABLE FOR THE GROUP BUFFER POOL. THIS**
DB2 IS NOT CURRENTLY CONNECTED TO THE GROUP BUFFER POOL.

Explanation: DB2 could not retrieve group buffer pool group (GDETAIL) statistics in response to a DISPLAY
GROUPBUFFERPOOL command because this DB2 subsystem is not currently connected to the group buffer pool. A
preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

System action: Processing continues.

DSNB781I   **GROUP DETAIL STATISTICS ARE NOT AVAILABLE FOR THE GROUP BUFFER POOL. MVS IXLCACHE RETURN CODE =** *retcode* **MVS IXLCACHE REASON CODE =** *reason*

Explanation: DB2 could not retrieve group buffer pool group (GDETAIL) statistics in response to a DISPLAY GROUPBUFFERPOOL command because the statistics request (IXLCACHE macro) did not successfully complete. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

retcode   Return code from MVS on the statistics request

reason   Reason code from MVS on the statistics request

For an explanation of IXLCACHE return and reason codes, see *MVS/ESA Programming: Sysplex Services Reference*.

System action: Processing continues.

DSNB782I   **INCREMENTAL GROUP DETAIL STATISTICS SINCE** *basetime*

Explanation: This message is displayed by the DISPLAY GROUPBUFFERPOOL command with GDETAIL(INTERVAL) specified. It indicates the start time of the interval for which the statistics were accumulated. This is the time of the previous GDETAIL incremental display from this DB2 or the time the group buffer pool was allocated or reallocated, if there was no previous GDETAIL incremental display.

System action: Processing continues.

DSNB783I   **CUMULATIVE GROUP DETAIL STATISTICS SINCE** *basetime*

Explanation: This message is displayed in response to the DISPLAY GROUPBUFFERPOOL command with GDETAIL(*) specified. It indicates the start time of the interval for which the statistics were accumulated. This is the time the group buffer pool was first allocated or reallocated.

System action: Processing continues.

DSNB784I   **GROUP DETAIL STATISTICS READS DATA RETURNED =** *r1*

Explanation: This is the first of five messages containing group statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the GDETAIL option. It lists the successful reads for the group buffer pool. A preceding DSNB750Ior DSNB751I message identifies the associated group buffer pool. This message is always followed by messages DSNB785I through DSNB788I.

r1   The "read hit" counter. The number of times that a page was returned on a coupling facility read request.

Related reference:

➡️ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

➡️ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:

"DSNB750I" on page 92

"DSNB751I" on page 92

DSNB785I   **DATA NOT RETURNED DIRECTORY ENTRY EXISTED =** *r2* **DIRECTORYENTRY CREATED =** *r3* **DIRECTORY ENTRY NOT CREATED =** *r4*, *r5*

Explanation: This is the second of five messages containing group statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the GDETAIL option. It continues the read statistics for the group buffer pool. The associated group buffer pool is identified in a preceding DSNB750I or DSNB751I message. This message is always preceded by message DSNB784I and followed by messages DSNB786I through DSNB788I.

All the "read counters" in this message represent cases where data was not returned on the coupling facility read request.

r2   "Read miss directory hit" counter. Number of times that a coupling facility read request specified a page for which a directory entry exists, but no datawas cached for that page.

r3	"Read miss name assigned" counter. Number of times that a coupling facility read request specified a page for which a directory entry was created.
r4	"Read miss assignment suppressed" counter. Number of times that a coupling facility read request specified a page for which no directory entry exists and no directory entry is created.
r5	"Read miss cache full" counter. Number of times that a coupling facility read request specified a page for which no directory entry exists and no directory entry is created because of a lack of storage in the group buffer pool.

Related reference:

➡ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

➡ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:

"DSNB750I" on page 92

"DSNB751I" on page 92

DSNB786I	**WRITES CHANGED PAGES=** *w1* **CLEAN PAGES =** *w2* **FAILEDDUE TO LACK OF STORAGE =** *w3* **CHANGED PAGES SNAPSHOT VALUE=** *p1*

Explanation: This is the third of five messages containing group statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the GDETAIL option. It is followed by messages DSNB787I and DSNB788I. It lists the statistics on writes to the group buffer pool and the total number of changed pages for the group buffer pool. The associated group buffer pool is identified in a preceding DSNB750I or DSNB751I message. This message is always preceded by messages DSNB784I and DSNB785I.

w1	"Changed page write hit" counter. Number of changed pages that were successfully written.
w2	"Clean page write hit" counter. Number of clean pages that were successfully written.
w3	"Write miss cache full" counter. Number of times a coupling facility write request was not completed because of a

lack of storage in the group bufferpool. |
| *p1* | "Total changed" counter. Snapshot value of the current number of changed pages. The displayed value of this counter is not affected by whether thisis an incremental or cumulative display. |

Related reference:

➡ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

➡ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:

"DSNB750I" on page 92

"DSNB751I" on page 92

DSNB787I	**RECLAIMS FOR DIRECTORY ENTRIES =** *d1* **FOR DATA ENTRIES =** *d2* **CASTOUTS=** *c1*

Explanation: This is the fourth of five messages containing group statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the GDETAIL option. It is always preceded by messages DSNB784I through DSNB786I and followed by message DSNB788I. It lists the statistics on reclaims and castouts for the group buffer pool. The associated group buffer pool is identified in a preceding DSNB750I or DSNB751I message.

d1	"Directory entry reclaim" counter. Number of times a name assignment required that a directory entry be reclaimed by the coupling facility.
d2	"Data entry reclaim" counter. Number of times a name assignment required that a data page be reclaimed by the coupling facility.
c1	"Castout" counter. Number of pages for which castout occurred.

Related reference:

 -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

 Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:

"DSNB750I" on page 92

DSNB788I   **CROSS INVALIDATIONS DUE TO DIRECTORY RECLAIMS =** *x1* **DUE TO WRITES =** *x2* **EXPLICIT =** *x3*

Explanation: This is the last of five messages containing group statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the GDETAIL option. It lists the cross-invalidation (XI) statistics for the group buffer pool. The associated group buffer pool is identified in a preceding DSNB750I or DSNB751I message. This message is always preceded by messages DSNB784I through DSNB787I.

x1   "XI for directory reclaim" counter. Number of times a directory entry was reclaimed (stolen) and XI signals had to be sent (that is, the stolen directory entry had registered DB2 interest).

x2   "XI for write" counter. Number of times a coupling facility write request caused XI to one or more registered DB2s.

x3   "XI for explicit invalidation" counter. Number of times an explicit coupling facility XI request caused XI to one or more registered DB2s.

Related concepts:

 Group buffer pool size is too small (DB2 Data Sharing Planning and Administration)

Related reference:

 Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

 -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

Related information:

"DSNB750I" on page 92
"DSNB784I" on page 101

DSNB789I   **REGISTER PAGE LIST =** *r7* **PAGES RETRIEVED =** *r8*

Explanation: This is the fourth of eight messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists more local read statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

The counter descriptions are listed below.

r7   The number of requests to the coupling facility to register interest in a list of pages. DB2 does this when prefetch (sequential or list) is used when the following conditions are met:
   • The DB2 member is running on a particular service level of MVS 5.2 or above
   • The GBP is allocated in a coupling facility with CFLEVEL=2 or above.

r8   Coupling facility read requests that were done for prefetch to retrieve a page in the group buffer pool as a result of feedback from the request to register a list of pages.

DSNB790I   **DISPLAY FOR GROUP BUFFER POOL** *gbpname* **IS COMPLETE**

Explanation: For a DISPLAY GROUPBUFFERPOOL command, this message signals the end of the display for group buffer pool *gbpname*.

DSNB791I   **THIS DB2 IS NOT CONNECTED TO** *gbpname*

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with TYPE(MCONN) and either one name or a list of names specified in the selection list. The DB2 on which the DISPLAY GROUPBUFFERPOOL command was issued is not currently connected to the specified group buffer pool. This message is issued once for each group buffer pool that does not match the "member connectivity" criterion.

System action: The display is terminated for this group buffer pool. Command processing continues, if there are other group buffer pool names to process.

DSNB792I   **NO DB2 IN THE GROUP IS CONNECTED TO** *gbpname*

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with TYPE(GCONN) and either one name or a list of names specified in the selection list. No DB2 in the group is currently connected to the specified group buffer pool. This message is issued once for each group buffer pool that does not match the "group connectivity" criterion.

System action: The display is terminated for this group buffer pool. Command processing continues, if there are other group buffer pool names to process.

DSNB793I   **DELETE NAME LIST =** *s5* **READ CASTOUT STATISTICS =** *s6* **DELETE NAME =** *s7* **OTHER ASYNCHRONOUS GBP REQUESTS =** *s8*

Explanation: This message is displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option if the group buffer pool is duplexed. It lists the statistics on requests to the secondary group buffer pool that are related to castout processing and to garbage collection processing.

s5   The number of Delete Name List requests to delete a set of pages from the secondary group buffer pool that have just been cast out from the primary. If the group buffer pool has been duplexed for the entire reporting interval, then this number should be roughly equal to the number of Unlock Castout requests to the primary (indicated in message DSNB796I).

s6   The number of Read Castout Statistics requests to check for "orphaned data entries" in the secondary group buffer pool. The DB2 member that is the group buffer pool structure owner periodically issues these requests to determine whether garbage collection is necessary to clean up orphaned data entries from the secondary. Orphaned data entries are "changed" data entries in the secondary group buffer pool without a corresponding "changed" data entry in the primary. Orphaned data entries are to be expected from time to time and do not represent any kind of data inconsistency problem (because the page has already been cast out from the primary and such that the version of the page in the secondary is the same as the version of the page on DASD). Orphaned data entries can be introduced by certain transaction or system failures, or by new group buffer pool write activity that encounters very unlikely timing windows when castout is concurrently processing.

s7   The number of Delete Name requests to delete "orphaned data entries" from the secondary group buffer pool. The DB2 member that is the group buffer pool structure owner issues these requests if it determines that garbage collection is necessary. See the description above under *s6* for a description of "orphaned data entries".

s8   The number of asynchronous CF requests sent to the secondary group buffer pool. This excludes ASYNCHRONOUS COMPLETION CHECKS as reported on the DSNB764I message.

Related reference:

⬆ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

⬆ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

Related information:
"DSNB750I" on page 92
"DSNB764I" on page 97
"DSNB796I" on page 105

DSNB794I   **THIS DB2 IS NOT CONNECTED TO ANY GROUP BUFFER POOL**

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with the TYPE(MCONN) option and an asterisk (*) specified as the selection list. This message is issued when DB2 determines that there are no group buffer pools having "member connectivity" from the DB2 on which the DISPLAY GROUPBUFFERPOOL command was issued.

System action: Command processing is terminated.

DSNB795I   **NO DB2 IN THE GROUP IS CONNECTED TO ANY GROUP BUFFER POOL**

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command with the TYPE(GCONN) option and an asterisk (*) specified as the selection list. When DB2 successfully determines (that is, no error is encountered during command processing) that there are no group buffer pools connected to any DB2 in the data sharing group.

System action: Command processing is terminated.

DSNB796I   **CASTOUTS PAGES CASTOUT =** *c1* **UNLOCK CASTOUT =** *c2* **READ CASTOUT CLASS =** *c3* **READ CASTOUT STATISTICS =** *c4* **READ DIRECTORY INFO =** *c5*

Explanation: This is the eighth of ten messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists local castout statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

Counters:

c1   Number of pages cast out.

c2   Number of requests to unlock for castout.

c3   Number of read castout class requests.

c4   Number of read castout statistics requests.

c5   Number of read directory info requests. These requests are made during group buffer pool checkpoint.

DSNB797I   **OTHER INTERACTIONS REGISTER PAGE =** *o1* **UNREGISTER PAGE =** *o2* **DELETE NAME =** *o3* **READ STORAGE STATISTICS =** *o4* **EXPLICIT CROSS INVALIDATIONS =** *o5* **ASYNCHRONOUS GBP REQUESTS =** *o6*

Explanation: This is the last of ten messages containing local statistics, displayed as a result of the DISPLAY GROUPBUFFERPOOL command with the MDETAIL option. It lists local castout statistics for the group buffer pool. A preceding DSNB750I or DSNB751I message identifies the associated group buffer pool.

Counters:

o1   Number of requests to the coupling facility to register interest in a single page.

o2   Number of requests to the coupling facility to remove the registered interest in a single page.

o3   Number of requests to the coupling facility to delete the directory entries associated with a set of pages.

o4   Number of requests to the coupling facility to read the statistics counters associated with the structure. DB2 generally issues this request once per group buffer pool checkpoint, and once per GDETAIL request on the DB2 DISPLAY GROUPBUFFERPOOL command.

o5   Number of explicit cross-invalidations for pages in the group buffer pool.

o6   Number of coupling facility requests to the primary group buffer pool that were converted from synchronous to asynchronous.

DSNB798I   **LAST GROUP BUFFER POOL CHECKPOINT** *timestamp* **MINIMUM CLEAN-TO-DIRTY LRSN =** *lrsn1* **MINIMUM MEMBER RESTART/REDO LRSN =** *lrsn2*, *member-name1* **STRUCTURE OWNER =** *member-name2*

Explanation: This message is displayed in response to the DISPLAY GROUPBUFFERPOOL command. It is issued only if DB2 determines, by successfully querying the MVS CFRM policy, that the group buffer pool is allocated. It is

DSNB799I

immediately preceded by message DSNB759I. A preceding DSNB750I message identifies the associated group buffer pool.

timestamp
> The date and time of the last group buffer pool checkpoint.

lrsn1
The recovery log record sequence number (LRSN) that was recorded by the last group buffer pool checkpoint that reflects the oldest changed page clean-to-dirty LRSN value at the time of the checkpoint. If the group buffer pool were to fail, *lrsn1* is used by DB2 as input to determine the starting points for the merged log scan to recover the page sets and partitions that are entered into group buffer pool recovery pending (GRECP) status. You can do more detailed monitoring of the group buffer pool checkpoint processing with IFCID 0261.

lrsn2
The minimum member-level write-pending (restart/redo) LRSN that was recorded by the last group buffer pool checkpoint. *member-name1* is the DB2 member associated with the minimum restart/redo LRSN. Group buffer pool checkpoint records a restart/redo LRSN for each member - *lrsn2* reflects only the single minimum value across all members. If the group buffer pool were to fail, DB2 uses each member's restart/redo LRSN in conjunction with *lrsn1* to determine the starting points for the merged log scan to recover the page sets and partitions that are entered into group buffer pool recovery pending (GRECP) status. You can do more detailed monitoring of the group buffer pool checkpoint processing with IFCID 0261.

member-name2
The member name of the group buffer pool structure owner. The structure owner's responsibilities include the following:

- Issue the group buffer pool checkpoint.
- Detect the GBPOOLT castout threshold and initiate castout if this threshold has been reached (by notifying the pageset or partition castout owners).
- Coordinate the damage assessment process in case the group buffer pool fails.
- Coordinate the rebuild process for the group buffer pool.

'UNKNOWN' is displayed if DB2 is not able to determine which member is the structure owner for the group buffer pool. This should only happen in an unexpected error situation. See the message accompanying that error.

System action: Processing continues.

DSNB799I **SECONDARY GBP ATTRIBUTES ALLOCATED SIZE =** *ssize* **KB VOLATILITY STATUS =** *svolstat* **CFNAME =** *cfname* **CFLEVEL - OPERATIONAL =** *cflevel1* **CFLEVEL - ACTUAL =** *cflevel2* **NUMBER OF DIRECTORY ENTRIES =** *sdir#* **NUMBER OF DATA PAGES =** *sdata#*

Explanation: For a DISPLAY GROUPBUFFERPOOL command, this message gives the attributes of the secondary group buffer pool if the group buffer pool is duplexed. It is displayed only if the group buffer pool is duplexed and if DB2 was successful in querying the MVS CFRM administrative policy. This message is not displayed if the group buffer pool is in a transition state between simplexing and duplexing.

ssize The actual allocated size (in kilobytes) of the secondary group buffer pool coupling facility structure.

svolstat The volatility status of the secondary group buffer pool:

> **VOLATILE**
> > The structure is volatile.
>
> **NON-VOLATILE**
> > The structure is non-volatile.

cfname The name of the coupling facility in which the group buffer pool is allocated. If the group buffer pool is duplexed, this is the coupling facility name associated with the primary group buffer pool.

cflevel1 The coupling facility level from a DB2 functionality perspective for the coupling facility in which the secondary group buffer pool is allocated.

cflevel2 The coupling facility level for the coupling facility in which the secondary group buffer pool is allocated.

sdir# Actual number of directory entries allocated for the secondary group buffer pool.

sdata#   Actual number of data pages allocated for the secondary group buffer pool.

DSNB800I   DIRECTORY ENTRY TO DATA PAGE RATIO SET TO *value* **FOR** *gbpname*. **IT WILL TAKE EFFECT ON THE NEXT REALLOCATION**

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that requests a change to the directory entry to the data page ratio for a group buffer pool. It is also issued for the first ALTER GROUPBUFFERPOOL command for a group buffer pool, even when no ratio value is specified on the command (in this case, the default ratio value is being used). The new ratio takes effect the next time the coupling facility cache structure is allocated.

The directory entry to data page ratio determines the number of directory entries and the number of data pages that are allocated in the coupling facility cache structure for the group buffer pool. If the coupling facility structure is not allocated, the first connect from any DB2 causes allocation of the structure and the new directory entry to data page ratio to take effect.

If the structure size can be determined from the MVS CFRM administrative policy, message DSNB801I is issued to indicate an estimate of the number of directory and data entries that are allocated based on the new ratio and the current size of the coupling facility structure. Otherwise, message DSNB812I is issued.

System action: Processing continues.

System programmer response: Use the -DISPLAY GROUPBUFFERPOOL command to display the number of allocated directory and data entries.

DSNB801I   MAX STRUCTURE SIZE INDICATED IN MVS CFRM POLICY = *size* **KB ESTIMATED NUMBER OF DIRECTORY ENTRIES =** *dir#* **ESTIMATED NUMBER OF DATA PAGES =** *data#*

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command after the directory entry to data page ratio is successfully updated. It is also issued for the first ALTER GROUPBUFFERPOOL command for a group buffer pool, even when no ratio value is specified on the command (in this case, the default ratio value is used). The group buffer pool is identified in message DSNB800I, which immediately precedes this message.

Because the actual ratio change does not take affect until the group buffer pool is reallocated, this message shows the estimated number of directory entries and data pages that are allocated based on the new ratio and the maximum size of the coupling facility structure, as indicated in the active CFRM policy definition. The actual number of entries that are allocated varies based on the amount of coupling facility storage that is available at the time the allocation is attempted, and on the amount of storage overhead necessary for the coupling facility control structures. Because DB2 cannot determine these variables, the estimated number of directory and data entries displayed in this message assumes that the entire storage allocation for the coupling facility cache structure will be used for directory and data entries, which is not the case.

System action: Processing continues.

Related concepts:

➡ Tuning group buffer pools (DB2 Data Sharing Planning and Administration)

Related tasks:

➡ Changing the ratio of directory to data entries (DB2 Data Sharing Planning and Administration)

Related reference:

➡ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

➡ -ALTER GROUPBUFFERPOOL (DB2) (DB2 Commands)

➡ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

Related information:

"DSNB750I" on page 92

"DSNB800I"

DSNB802I   "*value*" IS NOT A VALID PARAMETER FOR 'RATIO'

Explanation: An illegal value for the RATIO parameter was specified for an ALTER GROUPBUFFERPOOL command. The RATIO value can be up to 6 characters, and must be a decimal number from 1.0 to 255 (inclusive). For values less than 25, any digits after the first decimal place are ignored. If the value for the RATIO parameter is greater than or equal to 25, any digits after the decimal point are ignored. Informational message DSNB803I is displayed if the system ignores any precision that exceeds the limits of the RATIO parameter.

System action: The request is rejected.

System programmer response: Reissue the ALTER GROUPBUFFERPOOL command, and specify an acceptable value for RATIO.

DSNB803I   RATIO IS TRUNCATED TO *value*

Explanation: The precision specified on the RATIO parameter on an ALTER GROUPBUFFERPOOL command was greater than allowed. For decimal numbers less than 25, any digits after the first decimal place are ignored: for example, 5.67 is treated as 5.6. For decimal numbers greater than or equal to 25, any digits after the decimal point are ignored: for example, 25.3 is treated as 25.

System action: Processing continues.

DSNB804I   CLASS CASTOUT THRESHOLD SET TO *class-threshold1*,*class-threshold2* **FOR** *gbp-name*

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that requests a change to the class castout threshold value for a group buffer pool. It is also issued for the first ALTER GROUPBUFFERPOOL command for a group buffer pool, even when no threshold value is specified on the command (in this case, the default class castout threshold value is used).

class-threshold1
> The threshold at which class castouts are to be initiated, expressed as a percentage of the group buffer pool size

class-threshold2
> The threshold at which class castouts are to be initiated, expressed as an absolute number of buffers. *class-threshold2* is used to determine the threshold only if *class-threshold1* is zero and *class-threshold2* is non-zero. Otherwise, *class-threshold1* is used to determine the threshold.

gbp-name
> The name of the group buffer pool.

System action: Processing continues.

User response: No action is required.

Related reference:

➥ -ALTER GROUPBUFFERPOOL (DB2) (DB2 Commands)

DSNB805I   *specified-value* IS NOT A VALID PARAMETER FOR 'CLASST'

Explanation: An illegal value was specified on the CLASST (*class-threshold1*,*class-threshold2*) keyword on an ALTER GROUPBUFFERPOOL command. The *class-threshold1* value must be an integer between 0 and 90, inclusive. The *class-threshold2* value must be an integer between 0 and 32767, inclusive.

specified-value
> The specified value for CLASST.

System action: The request is rejected.

System programmer response: Reissue the ALTER GROUPBUFFERPOOL command, specifying an acceptable value for CLASST.

Related reference:

➥ -ALTER GROUPBUFFERPOOL (DB2) (DB2 Commands)

DSNB806I   GROUP BUFFER POOL CASTOUT THRESHOLD SET TO *value*% FOR *gbpname*

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that successfully changes the group buffer pool castout threshold value for a group buffer pool. It is also issued for the first successful ALTER GROUPBUFFERPOOL command for a group buffer pool, even when no threshold value is specified on the command (in this case, the default group buffer pool castout threshold value is being used).

System action: Processing continues.

DSNB807I   *value* IS NOT A VALID PARAMETER FOR 'GBPOOLT'

Explanation: An ALTER GROUPBUFFERPOOL command was issued with an illegal value specified on the GBPOOLT (group buffer pool castout threshold) keyword. The GBPOOLT value must be an integer between 0 and 90, inclusive.

System action: The request is rejected.

System programmer response: Reissue the -ALTER GROUPBUFFERPOOL command, specifying an acceptable value for GBPOOLT.

DSNB808I   GROUP BUFFER POOL CHECKPOINT INTERVAL SET TO *value* MINUTES FOR *gbpname*

Explanation: This message is issued in response to an ALTER GROUPBUFFERPOOL command that successfully changes the group buffer pool checkpoint interval value for a group buffer pool. It is also issued for the first successful ALTER GROUPBUFFERPOOL command for a group buffer pool, even when no interval value is specified on the command (in this case, the default group buffer pool checkpoint interval value is being used).

System action: Processing continues.

DSNB809I   *value* IS NOT A VALID PARAMETER FOR 'GBPCHKPT'

Explanation: An ALTER GROUPBUFFERPOOL command was issued with an illegal value specified on the GBPCHKPT (group buffer pool checkpoint) keyword. The GBPCHKPT value must be an integer between 1 and 999999, inclusive.

System action: The request is rejected.

System programmer response: Reissue the -ALTER GROUPBUFFERPOOL command, specifying an acceptable value for GBPCHKPT.

DSNB812I   SIZE ESTIMATES ARE NOT AVAILABLE FOR THE GROUP BUFFER POOL MVS IXCQUERY RETURN CODE = *return code* MVS IXCQUERY REASON CODE = *reason code*

Explanation: The directory entries and data pages estimates for the group buffer pool, identified on the preceding DSNB800I message, could not be computed because IXCQUERY was unable to obtain the backing structure's size information from the MVS CFRM administrative policy definition. This can happen if an IXCQUERY processing error occurred, or if the structure was not defined in the CFRM policy. The backing structure for the group buffer pool must be defined in the CFRM policy before the group buffer pool can be allocated. The exact reason can be determined from the *return code* and the *reason code*. For an explanation of the IXCQUERY codes, see *MVS/ESA Programming: Sysplex Services Reference*.

System action: The command completes successfully, but without giving the estimates.

DSNB813I   *csect-name* ALTER NOTIFICATION FAILED

Explanation: An ALTER GROUPBUFFERPOOL command failed to propagate the request for castout threshold or checkpoint interval change to other DB2s in the data sharing group. This error is triggered by an error encountered by the IRLM notification process. As a result, different DB2s in the group might have different values for the castout threshold or checkpoint interval. The RATIO change does not use the notification process and is, therefore, not affected by such a failure.

System action: Processing continues.

System programmer response: Reissue the -ALTER GROUPBUFFERPOOL command with the same parameters. If the problem persists, it indicates a deeper problem, such as inadequate space for the locking structures.

DSNB814I   *csect-name* **NO CHANGES REQUESTED ON COMMAND**

Explanation: An ALTER GROUPBUFFERPOOL command without any parameters was issued for an existing group buffer pool. If an ALTER GROUPBUFFERPOOL command without any parameters is issued for a group buffer pool that does not yet exist, a group buffer pool definition with default parameters values is created.

System action: The command is successful and processing continues.

DSNB815I   **PAGE P-LOCK LOCK/UNLOCK, REQUESTS FOR SPACE MAP =** *nn* **REQUESTS FOR DATA PAGES =** *nn* **REQUESTS FOR INDEX LEAF =** *nn* **UNLOCK REQUESTS =** *nn*

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command. This message reports the physical lock (P-lock) status at the Group Buffer Pool level. The accounting of these P-locks is intended for use in monitoring and tuning database performance, especially in a data-sharing environment. For more information about the DISPLAY GROUPBUFFERPOOL command, please see the DB2 Command Reference.

System action: Processing Continues.

Related reference:

▶ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

▶ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

DSNB816I   **PAGE P-LOCK SUSPENSIONS/NEGOTIATIONS, SUSPENSIONS FOR SPACE MAP =** *nn* **P-LOCK SUSPENSIONS FOR DATA =** *nn* **P-LOCK SUSPENSIONS FOR INDEX LEAF =** *nn* **NEGOTIATIONS FOR SPACE MAP =** *nn* **P-LOCK NEGOTIATIONS FOR DATA PAGES =** *nn* **NEGITIATIONS FOR INDEX LEAF PAGES =** *nn*

Explanation: This message is issued in response to a DISPLAY GROUPBUFFERPOOL command. This message reports the physical lock (P-lock) status at the group buffer pool level. These P-lock statistics are intended for use in monitoring and tuning database performance, especially in a data-sharing environment.

System action: Processing continues.

User response: No action is required.

Related reference:

▶ Group buffer pool monitoring with the DISPLAY GROUPBUFFERPOOL command (DB2 Data Sharing Planning and Administration)

▶ -DISPLAY GROUPBUFFERPOOL (DB2) (DB2 Commands)

DSNB890I   *command-verb gbp-kwd* **COMMAND IS ONLY VALID FOR DATA SHARING**

Explanation: An attempt was made to issue a GROUPBUFFERPOOL command on a DB2 that was not initialized to participate in data sharing.

command-verb
        The name of the command attempted: ALTER, or DISPLAY

gbp-kwd
        The keyword GROUPBUFFERPOOL or GBPOOL

System action: The command request is rejected.

DSNB891I   *name* **IS NOT A VALID GROUP BUFFER POOL OR COUPLING FACILITY STRUCTURE NAME**

Explanation: An invalid name was specified on an ALTER GROUPBUFFERPOOL or DISPLAY GROUPBUFFERPOOL command.

name   Either the DB2 group buffer pool name or the backing coupling structure name

You must either specify the DB2 group buffer pool name (*gpbname*) or the backing coupling facility structure name (*strname*). Acceptable values for *gbpname* are GBP0, GBP1, ..., GBP49, GBP32K, GBP32K1, ... , GBP32K9. The *strname* has the following format:

```
groupname_gbpname
```

The *groupname* is the DB2 data sharing group name. The underscore (_) is used as a delimiter between the *groupname* and the *gbpname*.

System action: The request is rejected.

System programmer response: Reissue the command specifying a valid group buffer pool name.

DSNB892I   *grpname* **IS NOT A VALID GROUP NAME FOR THIS DB2**

Explanation: The group name specified as part of the full coupling facility cache structure name on an ALTER GROUPBUFFERPOOL or DISPLAY GROUPBUFFERPOOL command is incorrect. On the commands, you can specify either the correct 16-byte coupling facility cache structure name or the DB2 group buffer pool name. The coupling facility cache structure name is a 16-character name specified as follows:

```
groupname_gbpname
```

The *gbpname* can be GBP0, GBP1, ..., GBP49 GBP32K, GBP32K1, ..., GBP32K9. The *groupname* is the DB2 data sharing group name. The underscore (_) is used as a delimiter between the *groupname* and the *gbpname*. The coupling facility cache structure names are left-justified with a blank as the rightmost padding character. DB2 adds the padding blanks if they are not specified on the command.

System action: The request is rejected.

System programmer response: Reissue the command, specifying a valid group buffer pool name.

Chapter 3. DSNE messages

Message identifiers that begin with DSNE communicate information about the TSO attachment facility.

In the TSO attachment facility, the user can control whether message IDs are displayed. To see them, type PROFILE MSGID. To suppress them, type PROFILE NOMSGID.

Trace messages associated with the TSO attachment facility (including DSN and DB2I) are not listed in this documentation. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

All DSN command processor trace messages have the same message number: DSNET20I.

DSNET20I

Explanation: This is one of many possible trace messages bearing this message number. For information about TSO attachment facility trace messages that have this message number, see DB2 Diagnosis Guide and Reference.

DSNE004E   **PROBABLE SUBSYSTEM ERROR DETECTED IN CSECT csect-name, SUBSYSTEM FUNCTION service, RETCODE retcode, REASON CODE reason-code**

Explanation: A subsystem error has occurred. The function identified in the message as 'service' is either an MVS or a DB2 service routine. The error was discovered in the 'csect-name' indicated in the message. Refer to the appropriate MVS publications for a description of the return and reason codes.

System action: This message is sometimes followed by an intentional system abend. In other instances, the DSN command processor will terminate the current subcommand and attempt to continue by issuing another DSN prompt.

User response: Run the job again with a SYSUDUMP dump data set allocated and DSN tracing set to 123. Collect the DSN trace stream. If an abend occurs, collect the dump.

DSNE005A   **EXECUTION IS INTERRUPTED. ENTER C TO CANCEL, OR ANY OTHER REPLY TO RESUME THE subcmd SUBCOMMAND**

Explanation: The Attention key was pressed. Entering a 'C' will cancel the DSN subcommand that is currently executing. If the user enters a question mark, message DSNE006A will appear. Any other reply will cause execution of the subcommand that was in progress to continue.

If the DSN prompt was issued just before the Attention key was pressed, 'subcmd' may be either 'DSN' or the previously invoked subcommand.

User response: Enter a 'C' to cancel the DB2 subcommand currently executing. The DSN prompt will appear on the terminal. Enter anything else (except another attention) to resume execution.

DSNE006A   **C WILL TERMINATE THE SUBCOMMAND WITH UNCOMMITTED CHANGES BACKED OUT**

Explanation: Enter a 'C' to terminate the subcommand. No uncommitted changes are made to the database. Any other reply causes subcommand execution to continue.

User response: Enter a 'C' to terminate the subcommand. Control returns to DSN, and the DSN prompt appears on the terminal. This is a second-level help message associated with message DSNE005A.

DSNE008I   **REPLY WAS xxxxx, EXECUTION RESUMED**

Explanation: Execution was suspended by a user ATTENTION and then was resumed, because the string specified in the message as 'xxxxx' was entered.

System action: Control returns to the interrupted session.

DSNE009I   **REPLY WAS C, EXECUTION TERMINATED**

Explanation: The response to message DSNE005A was a C. This caused the current subcommand to terminate.

System action: The current subcommand is terminated. The DSN prompt appears on the terminal.

User response: You can now enter another DSN subcommand.

DSNE010I   **SUBCOMMAND subcmd CANCELED BY ATTENTION**

Explanation: The subcommand specified in the messages was canceled because the user pressed the attention key and then entered a 'C'.

System action: The current subcommand is terminated. The DSN prompt appears on the terminal.

User response: A new subcommand can be entered.

DSNE011I   **DSN COMMAND CANCELED BY ATTENTION**

Explanation: The user of the DSN command processor has pressed attention. Since this occurred before DSN was fully initialized, DSN terminates. If DSN had been fully initialized, it would have prompted the user to continue or cancel.

System action: The DSN command processor terminates. Control returns to TSO or the calling CLIST.

User response: Restart DSN.

DSNE100I   **ssid NOT OPERATIONAL, RETRY COUNT IS ZERO**

Explanation: The subsystem (specified in the message as 'ssid') was not active, and no retries remain to be executed. No more connection attempts are made. The operator should be notified to start this specific subsystem, and the DSN command should be resubmitted. Issue the DSN command with the RETRY parameter set greater than zero.

System action: Control returns to TSO.

Operator response: Start the subsystem with the 'ssid' specified in the message.

User response: Notify the operator to start the subsystem.

DSNE101I   **ssid NOT OPERATIONAL, WILL RETRY nnn MORE TIMES**

Explanation: The requested DB2 subsystem 'ssid' was not started or is not available. Connection to it will be attempted 'nnn' more times. This message is issued every 30 seconds (and each time 'nnn' is decreased by 1) until the connection is made or 'nnn' is zero.

System action: DSN repeats the connection attempt until the connection is made or until the retry count is decreased to zero.

User response: Notify the operator start DB2.

DSNE102I   **YOU ARE NOT AUTHORIZED TO USE DB2 SUBSYSTEM ssid**

Explanation: The authorization ID is not authorized to use subsystem 'ssid'.

System action: Control returns to TSO.

User response: Contact the system programmer, or system administrator to arrange for the required RACF or equivalent security system authorization.

DSNE103E   MAIN STORAGE REQUEST FROM csect-id FAILED

Explanation: A request for main storage failed.

System action: The TSO attachment facility is ready for more input.

User response: Increase the address space size. This can be done as part of the logging on process.

DSNE104E   RETRY COUNT OF number NOT VALID, SET TO 120

Explanation: An invalid 'RETRY' parameter was entered on the DSN command. It has been set to 120. Valid retry counts are integer numbers from 0 to 120.

System action: Retry count is modified, and DSN execution continues.

User response: The next time you invoke DSN, use a RETRY count of 0 to 120.

DSNE105E   SUBSYSTEM ssid NO LONGER OPERATIONAL

Explanation: The DB2 subsystem is down. It must be restarted before the user can continue.

System action: Control returns to TSO attachment facility.

User response: Before you can continue, DB2 must be restarted. You must also reissue the DSN command. ISPF users should restart their jobs.

DSNE106E   PLAN plan-id NOT AUTHORIZED FOR SUBSYSTEM ssid AND AUTH-ID auth-id

Explanation: The requested combination of plan-id, subsystem-id, and user-id is not currently authorized.

System action: The TSO attachment facility is ready for more input.

System programmer response: The system administrator for the DB2 subsystem can arrange the necessary authorization.

User response: Notify the system programmer or system administrator to obtain proper authorization.

DSNE107E   SUBSYSTEM RESOURCE NEEDED FOR PLAN plan-name, AUTH ID auth-id, AND SUBSYSTEM subsys IS NOT AVAILABLE

Explanation: The DB2 subsystem was unable to allocate a necessary resource for the requested combination of plan-id, user-id, and subsystem-id. This message is accompanied by DSNE108E. An attempt to establish connection with the DB2 subsystem has failed.

System action: The TSO attachment facility is ready for more input.

User response: Arrange to obtain the needed resource, and resubmit the subcommand. See the explanation for message DSNE108E, which accompanies this message.

DSNE108E   FEEDBACK - REASON CODE reason-code TYPE= type RESOURCE NAME - resource-name

Explanation: This message accompanies message DSNE107E or DSNE114E to explain the problem more clearly. RESOURCE NAME is the name of the unavailable resource. The reason code is in hexadecimal format.

System action: The TSO attachment facility is ready for more input.

User response: Arrange for the missing resource to be available, and resubmit your job.

Problem determination: The reason the resource is unavailable is identified by a reason code.

TYPE and RESOURCE NAME identify the resource that is currently unavailable. The TYPE code identifies the type, as well as the format and content, of the RESOURCE NAME.

Related concepts:

⏵ DB2 reason codes (DB2 Codes)

Related information:

Appendix C, "Resource types," on page 1153

"DSNE107E" on page 115

"DSNE114E" on page 117

DSNE109E   NO INPUT TO BIND, SUBCOMMAND TERMINATED

Explanation: No LIBRARY parameter was specified on the BIND subcommand. No DBRMLIB was allocated.

System action: The TSO attachment facility is ready for more input.

User response: Specify the LIBRARY parameter, and resubmit the BIND subcommand. Alternatively, you could issue an ALLOCATE command for the DDNAME DBRMLIB and then resubmit the BIND subcommand as originally written.

DSNE110E   *ssid* NOT VALID SUBSYSTEM ID, COMMAND TERMINATED

Explanation: There is no subsystem id with the name identified in the message as 'ssid'. The subsystem identifier you specify (or default to) on the DSN command must exist. This message is the result of specifying (or defaulting to) a nonexistent subsystem identifier.

System action: The DSN processor ends, and control returns to TSO.

User response: Reissue the command with a valid SYSTEM parameter.

Related concepts:

⏵ DSN command processor (DB2 Application programming and SQL)

Related reference:

⏵ DSN (TSO) (DB2 Commands)

DSNE111E   INSTALLATION EXIT REJECTED DSNAME data-set-name

Explanation: A local installation-supplied exit has rejected the data set name identified in the message as 'data-set-name'.

System action: The TSO attachment facility is ready for more input.

User response: Either correct the data set name specified, or arrange to authorize the original data set name. Resubmit the request.

DSNE112E   userid STILL SIGNED ON TO SUBSYSTEM ssid

Explanation: An attempt was made to sign on to DB2 when the user was already signed on.

System action: Control returns to TSO.

User response: Log off and log on again.

DSNE113E   NOT ABLE TO LINK TO PROGRAM *program-name* IN DSNAME *dataset-name*.

Explanation: A TSO attachment error occurred during an attempt to link to program *program-name* in data set *dataset-name*.

program-name
      The name of the program.

dataset-name
      The name of the data set.

System action: The TSO attachment facility is ready for more input.

User response: Check that the program exists and that its name is spelled correctly. Enter the correct program name.

DSNE114E   THE COMBINATION OF PLAN planid, AUTH ID auth-id, AND SUBSYSTEM ssid IS NOT VALID

Explanation: The resource in message DSNE108E, which will follow this message, is not a valid resource name.

System action: The TSO attachment facility is ready for more input.

User response: Correct the invalid combination, and resubmit the subcommand. The problem is probably caused by a user spelling error. If it is not, check with the system administrator to determine the availability of the needed resource. Your program may need to be bound. See the explanation for message DSNE108 (which accompanies this one) for more information.

DSNE117E   NOT ABLE TO LOCATE membername IN DSNAME data-set-name

Explanation: The 'data-set-name' specified does not contain the member the user wanted to run.

System action: Control remains with the DSN command processor, and the DSN prompt appears on the terminal. The DSN command processor is ready for another command.

User response: Correct the PROGRAM or LIBRARY parameter, and resubmit the subcommand.

DSNE118E   invalid-command NOT VALID COMMAND

Explanation: The user entered an invalid command. A typing error may be responsible. You may be trying to run a command processor that is either not available on your system or that is not part of your current LNKLST or STEPLIB concatenation.

System action: The TSO attachment facility is ready for more input.

User response: Either correct the spelling error, or rearrange your library concatenation so that you can access the required command processor. Resubmit the command.

DSNE119E   data-set-name NOT PARTITIONED DATA SET

Explanation: The user tried to allocate a sequential data set when a partitioned data set was required.

System action: The TSO attachment facility is ready for more input.

User response: Check the data-set-name. The data set you specified is sequential, but you specified a member name. Remove the member name, and resubmit the job.

DSNE120E   data-set-name NOT SEQUENTIAL DATA SET

Explanation: The user tried to allocate a partitioned data set when a sequential data set was required.

System action: The TSO attachment facility is ready for more input.

User response: Check the data-set-name. The data set you specified is partitioned, but you did not specify a member name. Add a member name, and resubmit the job.

DSNE121E   *csect-id* NOT ABLE TO OPEN DATA SET

Explanation: The DB2 subsystem was unable to open the requested data set.

System action: The TSO attachment facility is ready for more input.

User response: For information about why the data set could not be opened, read the explanations of the messages that accompany this message. Correct the condition, and resubmit the job.

DSNE122E   DATA SET ORGANIZATION MUST BE EITHER SEQUENTIAL OR PARTITIONED. data-set-name IS NOT

Explanation: The DB2 subsystem was unable to open data set 'data-set-name', because it is not a sequential or partitioned data set.

System action: The TSO attachment facility is ready for more input.

User response: Correct the data-set-name, or change the data set organization. Resubmit the job.

DSNE123I   *data-set-name* **NOT PARTITIONED DATA SET, IT IS IGNORED**

Explanation: The data-set-named in the message is required to be partitioned. It is not and, consequently, was ignored.

System action: DSN will attempt to continue BIND processing.

User response: Correct the subcommand, and resubmit it.

DSNE124E   **DATA SET** *data-set-name*, **MEMBER** *member-name* **IN USE, TRY LATER**

Explanation: The member contained in 'data-set-name' is currently in use. This message is normally issued when another user is updating the specified member.

System action: The TSO attachment facility is ready for more input.

User response: Wait until the member is available and reissue your DCLGEN request.

DSNE125E   **DATA SET** *data-set-name* **IN USE, TRY LATER**

Explanation: The sequential data set 'data-set-name' is currently in use. This message is normally issued when another user is updating the specified data set.

System action: The TSO attachment facility is ready for more input.

User response: Wait until the data set is available and reissue your DCLGEN request.

DSNE126E   **THE DB2 OPERATOR IS STOPPING THE SUBSYSTEM. PLEASE END YOUR DSN SESSION**

Explanation: The operator has requested that DB2 quiesce. The user should promptly finish the current task and terminate the DSN session so that this can take place. This message may be followed by message DSNE127I.

System action: No new users will be permitted to initiate DB2 sessions. Current users will be permitted to work until they terminate their connections to DB2.

User response: End your DSN session as soon as possible.

DSNE127I   **THE DB2 OPERATOR IS STOPPING THE SUBSYSTEM. DSN IS TERMINATING**

Explanation: The operator has requested that DB2 terminate.

System action: DB2 will shut down, interrupting user activities as necessary. DSN will terminate immediately and return the user to TSO.

User response: Reinitiate your session when DB2 returns.

DSNE128I   **DB2 IS ABNORMALLY ENDING. DSN IS TERMINATING**

Explanation: DB2 has encountered a problem from which it cannot recover.

System action: DB2 is abending. DSN terminates immediately and returns the user to TSO.

User response: Reinitiate your session when DB2 returns.

DSNE130I   **MAXIMUM NUMBER OF DB2 USERS REACHED. YOUR RETRY COUNT IS ZERO**

Explanation: The number of concurrent connections to DB2 is restricted. The number is specified in the installation parameters in member DSN6SYSP. For TSO foreground, see parameter IDFORE. For background jobs, see parameter IDBACK. The DSN request has been denied, because the current number of DB2 connections in the system has reached the installation-specified limit.

System action: Control returns to TSO.

User response: Resubmit the DSN command when activity using DB2 has diminished. If still unsuccessful, see the system programmer about the installation parameters.

DSNE131I   MAXIMUM NUMBER OF DB2 USERS REACHED. WILL RETRY *nnn* **MORE TIMES**

Explanation: The number of concurrent connections to DB2 is restricted. The number is specified in the installation parameters in member DSN6SYSP. For TSO foreground, see the value of the IDFORE subsystem parameter. For background jobs, see the value of the IDBACK subsystem parameter. The DSN request was denied because the current number of DB2 connections in the system reached the installation specified limit.

The DSN connection to DB2 is attempted *nnn* more times. The number of retries is determined by the RETRY parameter on the DSN command.

This message is issued every 30 seconds (and each time *nnn* is decreased by 1) until the connection is made or until *nnn* is decreased to zero.

System action: DSN repeats the connection attempt until the connection is made or the retry count is decreased to 0.

User response: If all retry attempts fail, resubmit the DSN command when DB2 activity diminishes. If still unsuccessful, see the system programmer about the installation parameters.

Related tasks:

⮕ Setting thread limits (DB2 Performance)

⮕ Setting TSO options for threads (DB2 Performance)

Related reference:

⮕ DSN (TSO) (DB2 Commands)

⮕ The DSN command and its subcommands (DB2 Commands)

⮕ MAX TSO CONNECT field (IDFORE subsystem parameter) (DB2 Installation and Migration)

⮕ MAX BATCH CONNECT field (IDBACK subsystem parameter) (DB2 Installation and Migration)

DSNE132I   DB2 IS OPERATING IN RESTRICTED ACCESS MODE. MAINTENANCE PERSONNEL ONLY

Explanation: DB2 is not available for normal operation. It is only available to service personnel.

System action: Control returns to the caller. Your request was not processed. The DSN command processor was unable to connect to DB2.

User response: Wait until DB2 is available for general use, then resubmit your request.

DSNE133E *csect-name* **RELEASE LEVELS NOT COMPATIBLE. DSN=**release_level**, DB2=**release_level

Explanation: The DSN command processor, perhaps running under the control of DB2I, has detected a release level incompatibility between itself and DB2. The current *release_level* of the DSN command processor code and the DB2 subsystem load modules is indicated as a string of three numeric characters as follows:

- Version
- Release
- Modification level

System action: The DSN command processor terminates.

Operator response: Notify the system programmer.

System programmer response: If you are unable to determine the problem, you can trace DSN execution through DB2I or the DSN command processor.

User response: Check that your method of invoking DSN or DB2I allocates the correct libraries for DSN. If you used JCL or a TSO logon procedure to execute DSN code, check your JOBLIB and STEPLIB allocations. If you used a CLIST to execute DSN, ensure that you obtained the correct version of the DSN code. See your system programmer if you invoke DSN using procedures supplied by your system programmer.

Problem determination: Ensure that coexistence is supported on both release levels (DSN and DB2). If coexistence is

supported, ensure that the lower level release has the correct SPE level code support that is required to coexist with the current higher level release.

For more information about coexistence and SPEs, refer to the program directory or preventative service planning (PSP) updates.

DSNE136I   DB2 IS OPERATING IN RESTART LIGHT MODE. INDOUBT RESOLUTION ONLY

Explanation: The DB2 member is not available for normal operation. It is only available to commit coordinators that have indoubt units of recovery that need to be resolved with this member.

System action: Control returns to the caller. Your request was not processed. The DSN command processor was unable to connect to DB2.

User response: Wait until this DB2 is available for general use, then resubmit your request. Or, request connection to a different DB2 member in the data sharing group (for example, by using the DB2 group attach name).

DSNE137I   YOU ARE NOT AUTHORIZED TO CONNECT TO DB2 ASUSER *userid***.**

Explanation: The authorization ID is not authorized to use DB2 with the ASUSER *userid*.

System action: The request was not processed.

User response: Contact the system administrator to arrange for the required trusted context authorization.

DSNE138E   NOT ABLE TO LINK TO PROGRAM *program-name* **IN THE STANDARD SEARCH ORDER. RETURN CODE** *return-code***. REASON CODE** *reason-code***.**

Explanation: A DB2 TSO attachment error occurred during an attempt to link to program *program-name* using the standard load module search order.

program-name
>   The name of the program.

return-code
>   A numeric value that indicates the reason for the failure.

reason-code
>   A numeric value that indicates the reason for the failure.

System action: The TSO attachment facility is ready for more input.

User response: Correct the program or execution environment, and resubmit the subcommand.

DSNE139E   NOT ABLE TO LINK TO MEMBER *member-name* **IN THE STANDARD SEARCH ORDER.**

Explanation: The member *member-name* was not found using the standard load module search order.

member-name
>   The member name of a particular DB2 subsystem in a data sharing group.

System action: Control remains with the DSN command processor, and the terminal displays the DSN command prompt. The DSN command processor is ready for another command.

User response: Correct the program or execution environment, and resubmit the subcommand.

DSNE231A   ONE NAME IS REQUIRED TO FREE A PLAN OR A PACKAGE

Explanation: The user has attempted to use the FREE PLAN or PACKAGE panel without specifying a plan or package name.

System action: The TSO attachment facility is ready for more input.

User response: Enter the name of a plan or package, or (*) to indicate 'all'.

DSNE232A   CONNECTION NAME REQUIRES ENABLE/DISABLE OF CONNECTION TYPE

Explanation: The user indicated that specific connection names were to be specified for ENABLE or DISABLE, but the associated connection type field was not updated. The ENABLE or DISABLE of connection names requires the corresponding connection type to also be specified for ENABLE or DISABLE.

System action: The TSO attachment facility is ready for more input.

User response: Enter Y (Yes) on the connection type for the associated connection names.

DSNE233A   MEMBER NAME MUST NOT BE SPECIFIED MORE THAN ONCE

Explanation: The LIBRARY field containing the data set name, also contains a member name.

System action: The TSO attachment facility is ready for more input.

User response: The member name must be specified only once, and must be specified on the MEMBER field.

DSNE234A   EITHER DBRM MEMBER OR INCLUDE PACKAGE LIST MUST BE SPECIFIED

Explanation: The BIND PLAN subcommand requires either a DBRM member name or a package list as input, neither was specified.

System action: The TSO attachment facility is ready for more input.

User response: Enter a DBRM member name(s), or specify that a package list is to be included. Both options are also allowed.

DSNE235A   *plan/package* NAME IS A DUPLICATE OF PRIMARY *plan/package* NAME

Explanation: In addition to the primary 'plan' or 'package' name specified on the REBIND or FREE subcommand panel, the user also specified that additional plans or packages were to be included in the list. The primary 'plan' or 'package' name was again specified to be included in the list.

System action: The duplicate of the primary 'plan' or 'package' name is automatically deleted from the list to avoid errors during subcommand processing. The TSO attachment facility is ready for more input.

User response: Complete the list of additional plans or packages.

DSNE236A   SPECIFY A CONNECTION TYPE TO ENABLE OR DISABLE

Explanation: The user indicated that individual connection types were to be ENABLED or DISABLED, but no specific connection types were specified.

System action: The TSO attachment facility is ready for more input.

User response: Enter Y (YES) for each individual connection type to ENABLE, or DISABLE.

DSNE237A   SPECIFY ENABLE ALL OR SPECIFIC CONNECTION TYPES

Explanation: In addition to requesting that all connection types be enabled, the user also requested that specific connection types be enabled. The enable of all connection types includes the enable of specific connection types.

System action: The TSO attachment facility is ready for more input.

User response: Specify specific connection types to be enabled, or specify that all connection types are to be enabled.

DSNE240A   ISPF RETURNED AN UNEXPECTED CODE WHILE CLOSING A TABLE

Explanation: An ISPF error return code was encountered when attempting to perform a TBCLOSE for an ISPF table while processing the current subcommand panel.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Verify that the ISPF table library allocated to the DSNETBLS DD statement has sufficient space or directory entries available. Also refer to the ISPF error log for additional diagnostic information.

DSNE241A • DSNE245A

User response: Notify the system programmer.

DSNE241A   INTERNAL PROCEDURE WAS PASSED AN INVALID NUMBER OF ARGUMENTS

Explanation: A TSO Interpreter error was encountered while executing a subroutine during the processing of the current subcommand.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Notify the IBM support center that an error was encountered in the TSO SYSCALL instruction.

User response: Notify the system programmer.

Problem determination: The TSO SYSCALL instruction is not passing all the parameters to a subroutine defined within the CLIST.

DSNE242A   ISPF RETURNED AN UNEXPECTED CODE WHILE SETTING SEARCH ARGUMENTS

Explanation: An ISPF table for the current function does not contain the expected column or table format.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Delete the ISPF table in error. The table is re-created automatically during the next invocation of the DB2I function.

User response: Notify the system programmer.

Problem determination: The format of the ISPF table was changed since the last successful use of the table. Refer to the DB2I ISPF table library allocated to the DSNETBLS DD statement, and examine the ISPF error log for additional diagnostic information.

DSNE243A   ISPF RETURNED AN UNEXPECTED CODE DURING A TABLE DISPLAY

Explanation: An ISPF error was encountered when attempting to display a DB2I table panel.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Ensure that the DB2I panel and panel library are established correctly. If the format of both is correct, use ISPF option 7.2 to test the panel for additional errors.

Examine the ISPF error log for additional error information and determine the name of the panel to be displayed.

User response: Notify the system programmer.

DSNE244A   CANNOT SAVE CREATED TABLE, VERIFY SPACE IN TABLE LIBRARY

Explanation: An ISPF TBSAVE function returned with a condition code greater than zero.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Verify that the ISPF table library allocated to the DSNETBLS DD statement has sufficient space and directory entries available. Also refer to the ISPF error log for additional diagnostic information.

User response: Notify the system programmer.

DSNE245A   ISPF RETURNED AN UNEXPECTED CODE WHILE ADDING A ROW TO A TABLE

Explanation: An ISPF TBADD function returned with a condition code greater than four.

System action: The TSO attachment facility is ready for more input.

System programmer response: Verify that the user has enough memory to perform ISPF functions. Also refer to the ISPF error log for additional diagnostic information.

User response: Enter the DB2I CANCEL command, and notify the system programmer.

DSNE246A   ISPF RETURNED AN UNEXPECTED CODE WHILE DELETING A ROW

Explanation: An ISPF TBDELETE function returned with a condition code greater than zero.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Verify that the user has enough memory to perform ISPF functions. Also refer to the ISPF error log for additional diagnostic information.

User response: Enter the DB2I CANCEL command, and notify the system programmer.

DSNE247A   ISPF RETURNED AN UNEXPECTED CODE WHILE UPDATING A ROW

Explanation: An ISPF TBPUT function returned with a condition code greater than four.

System action: TSO processing for the current subcommand is terminated.

System programmer response: Verify that the user has enough memory to perform ISPF functions. Recreate the user actions that lead to the failure. Also refer to the ISPF error log for additional diagnostic information.

User response: Enter the DB2I CANCEL command, and notify the system programmer.

DSNE248A   COMMAND ENTERED CONTAINS INVALID CHARACTERS, RESPECIFY

Explanation: The command entered at the command line contains an invalid character, the command must be reentered.

System action: The TSO attachment facility is ready for more input.

User response: Correct the invalid command entry.

DSNE250A   PLEASE ENTER THE COMMAND AT THE LINE WITH THE PLAN/PACKAGE NAME

Explanation: SELECT or RESET commands must be entered on the line where the plan or package name is shown. This line contains the name of the plan or package, while the secondary lines do not.

System action: The line command is rejected. The TSO attachment facility is ready for more input.

User response: Reenter the line command on the primary plan or packagelines.

DSNE252A   JCL WAS NOT CREATED, ERROR *errcode* **WHILE DOING** *function_name* **FUNCTION**

Explanation: Error code 'errcode' was received while doing a function 'function_name', the JCL was not created.

System action: Refer to *z/OS ISPF Dialog Developer's Guide* for the cause of this error.

User response: Contact your systems programmer.

Problem determination: Use the return code/function combination to find the causes for the problem in *z/OS ISPF Dialog Developer's Guide* and correct the problem.

Collect the following diagnostic items:

• Listing of the ISPF log for the session involved in the problem.

Severity: 20

DSNE283A   THE , (COMMA) OPTION IS INCOMPATIBLE WITH THE *hostlang* **LANGUAGE**

Explanation: The user tried to select the 'hostlang' APPLICATION LANGUAGE and the comma option of the DECIMAL POINT field on the DB2I defaults panel. This combination is not supported.

System action: The defaults panel is displayed again.

User response: Either change the APPLICATION LANGUAGE field or the DECIMAL POINT field. If you need the C (C/370™ Compiler), then the DECIMAL POINT field must contain a decimal point (.).

DSNE288A   YOU CANNOT SPECIFY (RUN)=YES WITH CICS OR IMS

Explanation: Application programs that make CICS or IMS calls cannot be run from the TSO environment. They can, however, be precompiled, bound, compiled, and linked under TSO. The user should specify N on the RUN panel when preparing a CICS or IMS application.

System action: The system is waiting for new input to the panel.

User response: Change the RUN option to N.

DSNE291A   ALL DB2 COMMANDS BEGIN WITH A DASH

Explanation: The user omitted the DB2 command recognition character that must precede all DB2 commands.

System action: The DB2 commands panel is displayed again so the user can resubmit the command.

User response: Add the DB2 command recognition character, and reenter the command.

DSNE292I   DATA SET *data-set-name* **NOW CONTAINS YOUR JCL**

Explanation: DB2I has just created some JCL for you. The JCL now resides in data set 'data-set-name'. You can now edit this data. The TSO SUBMIT command will start the job running in background.

System action: The system is waiting for your next request.

User response: You can start another DB2I job or leave the panel.

DSNE293A   MEMBER NAME IS REQUIRED. YOUR PROGRAM MUST BE IN A PDS

Explanation: The user either has attempted to run a program that is not in a partitioned data set or has forgotten to specify a member name.

System action: The RUN panel will be displayed so that the user can specify a member name and try the run again.

User response: Specify a member name, and rerun the program.

DSNE294I   SYSTEM RETCODE=*hex-code* **USER OR DSN RETCODE=***decimal-code*

Explanation: A DB2I function has just completed. The return codes for that function are returned in this message. The variable 'hex-code' is a 3-digit hexadecimal number; the variable 'decimal-code' is a 3-digit decimal number. Refer to the appropriate MVS publication for explanations of the return codes.

System action: DB2I is ready for another execution cycle unless other messages appeared indicating that DB2 is unavailable.

User response: You may have to change input parameters to correct unsatisfactory completion codes.

DSNE295A   YOU CANNOT DISPLAY AND RESTART A UTILITY AT THE SAME TIME

Explanation: You have set options on the DB2I Utilities panel that call for conflicting actions. You can either DISPLAY the status of a utility, or you can RESTART a utility, but you cannot do both.

System action: The system is waiting for new input to the panel.

User response: Either change the RESTART option to NO, or select a different FUNCTION.

DSNE296A   YOU CANNOT TERMINATE AND RESTART A UTILITY AT THE SAME TIME

Explanation: You have set conflicting options on the DB2I Utilities panel. You can either TERMINATE a utility, or you can RESTART a utility, but you cannot do both.

System action: The system is waiting for new input to the panel.

User response: Either change the RESTART option to NO, or select a different FUNCTION.

DSNE297A   ENTER DATA SET NAME: COPYDSN OR RCPYDSN1

Explanation: You have not specified a data set name for the COPY utility. A data set name for the local/current site (COPYDSN), and/or the recovery site (RCPYDSN1) is required.

System action: The system is waiting for new input to the panel.

User response: Enter a data set name for COPYDSN, and/or RCPYDSN1.

DSNE298A   ERROR DURING EDIT: CLIST *data set name*

Explanation: Errors were encountered while attempting to change the online book data set names in the DB2 CLIST member DSNEMC01. No data set names were changed.

System action: The system is waiting for new input to the panel.

System programmer response: If you are unable to determine the problem, you can trace the DB2 CLIST execution while displaying this panel.

User response: Ensure that the CLIST data set name is correct and contains the DSNEMC01 CLIST member to be changed. This CLIST data set must also be available to the TSO user ID in WRITE mode. This error will continue to be displayed on the panel until the error is resolved or the user exits the panel (END).

Problem determination: Examine the CLIST trace output for any TSO/ISPF nonzero return codes encountered during CLIST execution.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNE340I   INTERNAL ERROR - ILLEGAL TYPE IN MODULE *module-name*

Explanation: An internal DB2 problem has occurred. The named module has detected a nonexistent SQLTYPE.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Collect the SPUFI trace and ISPF log, and notify either the system programmer or the database administrator.

You may have to rerun the failing job after you have turned on trace. This will enable you to send trace messages to the ISPF log.

DSNE345I   WARNING: DB2 DATA CORRUPTION CAN RESULT FROM THIS SPUFI SESSION BECAUSE THE CCSID USED BY THE TERMINAL IS NOT THE SAME AS THE CCSID USED BY SPUFI. TERMINAL CCSID *terminal-ccsid*. **SPUFI CCSID:** *spufi-ccsid*. **NOTIFY THE DB2 SYSTEM ADMINISTRATOR.**

Explanation: The current SPUFI session was initiated on a TSO terminal that uses a coded character set identifier (CCSID) of *terminal-ccsid*. However, SPUFI was bound on the local DB2 server using a CCSID of *spufi-ccsid*. This discrepancy can lead to unexpected data conversion, affecting any characters that do not map to the same code point in the two CCSIDs.

If *terminal-ccsid* is blank, then the terminal code page and character set cannot be queried or they are not supported by ISPF. Possible reasons why the terminal code page and character set cannot be queried include:

- Hardware limitation such as use of older terminals
- Improper configuration of the terminal emulator software
- Improper BTAM or control unit setup

A blank *spufi-ccsid* indicates that SPUFI might have received an unexpected SQLCODE when it attempted to query the current application encoding scheme. Diagnostics for that SQLCODE appear directly after the DSNE345I message in the message warning panel.

System action: SPUFI waits for the user to indicate whether to continue or quit.

System programmer response: On the local DB2, SPUFI expects to process only EBCDIC character data encoded in

the CCSID of *spufi-ccsid*. Ensure that the user's TSO terminal configuration is updated to use a CCSID of *spufi-ccsid*. Verify the data integrity of all tables modified during the SPUFI session and follow standard recovery procedures. Contact IBM for guidance if widespread or long term data corruption is evident (for example, if many users report this message when using SPUFI).

Special considerations:

Blank CCSIDs

As noted previously, a blank *terminal-ccsid* setting occurs when the terminal code page and character set cannot be queried or they are not supported by ISPF. Check for the following situations:

- If the message occurs when using a terminal emulator, verify that the emulator software is configured with a CCSID setting.
- Check the VTAM® setup: The VTAM logmode entry used to define the session characteristics for the terminal must have the EDS (extended data stream) bit turned 'on' in the PSERVIC (that is, pservic=xx80xx..). For more information, consult *OS/390 ISPF Planning and Customization* and the *z/OS SNA Network Implementation Guide*.
- Older terminal hardware such as the IBM 327x series do not support the EDS, so a CCSID cannot be determined. You can instruct users of such devices to specify in the SPUFI defaults panel that the DSNE345I message should be suppressed. Note that suppression applies only when the terminal CCSID is blank. You cannot use this field to suppress DSNE345I when *terminal-ccsid* is non-blank.

A blank *spufi-ccsid* indicates that SPUFI probably received an unexpected SQLCODE when it attempted to query the current application encoding scheme. Check the diagnostics for the SQLCODE that is displayed directly after the DSNE345I message, and then respond to the diagnostics accordingly.

User response: Notify the system programmer. Depress the END key to quit or the ENTER key to continue.
Important: Continuing can result in corruption of data stored by DB2.

Related tasks:

Making SPUFI work with different terminal CCSIDs (DB2 Installation and Migration)

DSNE346I   WARNING: ALL SHIFT-IN AND SHIFT-OUT CHARACTERS IN THE 'FOR BIT DATA' COLUMNS IN THE OUTPUT ABOVE HAVE BEEN REPLACED BY THE SUBSTITUTION CHARACTER '.'

Explanation: The current SQL query returned one or more columns that have a data type of CHAR FOR BIT DATA, VARCHAR FOR BIT DATA, or both. Also, the current SPUFI session indicates that a graphic character set is being used. To prevent the possible corruption of outputted data elsewhere in the report, all shift-in (code point x'0E') and shift-out (code point x'0F') characters will be replaced with a substitution character of '.' (code point x'4B').

System action: SPUFI processing continues.

User response: To avoid the substitution character for shift-in and shift-out data, and in order to see all FOR BIT DATA in hexadecimal format, go to the SPUFI defaults panel and change the setting of the FOR BIT DATA field from ASIS to HEX.

DSNE351A   EDIT PROCESSING OPTION MUST BE Y (YES) OR N (NO)

Explanation: The user specified an edit option other than 'YES', 'NO', or the abbreviation 'Y' or 'N'. A valid user response determines whether SPUFI invokes the ISPF editor to allow the user to edit the input data set.

System action: SPUFI will wait until the user enters a valid response.

User response: Type in a 'Y' or an 'N', and press 'ENTER'.

DSNE352A   EXECUTE PROCESSING OPTION MUST BE Y (YES) OR N (NO)

Explanation: The user specified an execute option other than 'YES', 'NO', or the abbreviation 'Y' or 'N'. A valid user response determines whether SPUFI executes the SQL statements in the user input data set.

System action: SPUFI waits until the user enters a valid response.

User response: Type in a 'Y' or an 'N', and press 'ENTER'.

DSNE353A   BROWSE PROCESSING OPTION MUST BE Y (YES) OR N (NO)

Explanation: The user specified a browse option other than 'YES', 'NO', or the abbreviation 'Y' or 'N'. A valid user response determines whether SPUFI invokes ISPF BROWSE to allow the user to browse the output data set.

System action: SPUFI will wait until the user enters a valid response.

User response: Type in a 'Y' or an 'N', and press 'ENTER'.

DSNE354A   AUTOCOMMIT PROCESSING OPTION MUST BE Y (YES) OR N (NO)

Explanation: The user specified an AUTOCOMMIT option other than 'YES', 'NO', or the abbreviation 'Y' or 'N'. A valid user response determines whether SPUFI automatically commits all the changes to the database subsystem. This was encountered as a result of executing the user's SQL statement(s). The commit is done at the end of the processing of an input data set, if it is requested and if there has been no SQL execution error.

System action: SPUFI waits until the user enters a valid response.

User response: Type in a 'Y' or an 'N', and press 'ENTER'.

DSNE355A   CHANGE DEFAULTS PROCESSING OPTION MUST BE Y (YES) OR N (NO)

Explanation: The user specified a change-default option other than 'YES', 'NO', or the abbreviation 'Y' or 'N'. A valid user response determines whether SPUFI displays the current-session SPUFI defaults and allows the user to change the defaults.

System action: SPUFI waits until the user enters a valid response.

User response: Type in a 'Y' or an 'N', and press 'ENTER'.

DSNE359A   OUTPUT DATA SET NAME REQUIRED FOR EXECUTE OR BROWSE

Explanation: The user did not specify the name of the output data set.

System action: SPUFI waits until the user specifies an output data set.

User response: Type in the correct entry, and press 'ENTER'.

DSNE360A   THE OUTPUT DATA SET MUST RESIDE ON DASD

Explanation: The user has attempted to send the output of SPUFI to a SYSOUT data set. He must enter a generic device class that contains only DASD devices.

System action: SPUFI waits until the user specifies a valid output device class.

User response: Type in the correct entry, and press 'ENTER'.

DSNE361I   SPUFI PROCESSING COMPLETE

Explanation: This informational message is displayed after SPUFI successfully created the output data set. If errors are encountered before the completion of output data set processing, or during the subsequent BROUSE OUTPUT option for this data set, this message is replaced by either SPUFI or TSO/ISPF error messages.

System action: The system waits for the next user input.

User response: At this point you can change the SPUFI panel variable settings to initiate a new SPUFI request, or you can exit SPUFI.

DSNE363A   ONLY RR (REPEATABLE READ), CS (CURSOR STABILITY), OR UR (UNCOMMITTED READ) ALLOWED

Explanation: The user has entered an invalid isolation level. This value must be one of the following:
• RR
• CS

'Isolation level' here refers to the degree to which the SPUFI user wants to isolate SQL execution from the effects of other concurrently executing DB2 applications.

RR

> If the repeatable read (RR) level is specified and the AUTOCOMMIT option is 'YES', other applications can modify the selected database values as soon as the SPUFI EXECUTE phase is completed and as soon as the automatic COMMIT (or ROLLBACK, if an error was encountered) operation is performed. Therefore, it is possible that the retrieved data being browsed by the user has already been modified by another application.
>
> If the RR level is specified and the AUTOCOMMIT option is 'NO', then no other DB2 application executing concurrently can change any of the selected database values until the SPUFI user issues either a COMMIT or a ROLLBACK. Therefore, other applications might be in a normal wait state while the SPUFI user is browsing the output data set.

CS

> If the cursor stability (CS) level is specified, it does not make any difference whether the AUTOCOMMIT option is 'YES' or 'NO'. Other concurrently executing DB2 applications can change the selected database values as soon as the SELECT has completed.

UR

> If the uncommitted read (UR) level is specified, SPUFI queries can read data that has been changed, but has not yet been committed by another application. It does not make any difference whether the AUTOCOMMIT option is 'YES' or 'NO'. Other concurrently executing DB2 applications can change the selected database values without waiting for the SELECT to complete.

System action: SPUFI waits until the user specifies a valid isolation level.

User response: Type in the correct entry, and press 'ENTER'.

DSNE364A   *token* **IS INVALID AS AN SQL STATEMENT TERMINATOR**

Explanation: The user has entered an SQL statement terminator (token) that is not supported. The following values are not valid for use as a SQL statement terminator:
- blank
- comma
- single and double quotes
- underscore
- left and right parenthesis

System action: The statement cannot be executed.

User response: Specify a valid SQL statement terminator.

DSNE365A   **SQL STATEMENT TERMINATOR HAS BEEN CHANGED TO** *token*

Explanation: The SPUFI input data set contains a #SET statement that changes the SQL statement terminator to *token*.

System action: *token* is the SQL statement terminator until another #SET statement changes it or the end of the input data is reached.

DSNE366A   **ONLY THE LISTED RECFM TYPES ARE ALLOWED**

Explanation: The user has entered an invalid record format for the output data set.

System action: The system waits until the user enters one of the allowed RECFM types.

User response: Enter a RECFM of V, VB, VBA, F, FB, or FBA.

DSNE367A   ACTION MUST BE C (COMMIT) OR R (ROLLBACK) OR D (DEFER)

Explanation: This message requests the user to make a commit action decision.

User response: To commit all the changes made since the last commit point, enter a 'C'. To undo all the changes that have been made since the last commit point, enter an 'R'. To defer the decision, enter a blank.

DSNE368A   YES OR NO MUST BE ENTERED FOR EACH PROCESSING OPTION

Explanation: One of the processing options was left blank or set to blank by the user.

User response: Enter 'YES' or 'NO' for each processing option.

DSNE370A   INPUT DATA SET COULD NOT BE OPENED

Explanation: The input data set could not be opened because of invalid data set attributes. The logical record length (LRECL) of the input data set must be 80K. The record format (RECFM) must be either F or FB. The input data set may be either sequential or partitioned.

User response: Specify an input data set having valid attributes.

DSNE371A   OUTPUT DATA SET COULD NOT BE OPENED

Explanation: The output data set could not be opened because of invalid data set attributes. The logical record length (LRECL) must be at least 80KB and less than 32KB. The record format (RECFM) must be F, FB, FBA, V, VB, or VBA. The output data set must be sequential.

User response: Either specify an output data set having valid attributes, or enter correct attributes for this data set.

DSNE372A   OUTPUT DATA SET MUST BE SEQUENTIAL

Explanation: The user has requested that a nonsequential output data set contain the output from execution of SQL statements.

User response: Enter the data-set-name of a sequential data set for the output data set.

DSNE374A   OUTPUT DATA SET MUST BE SEQUENTIAL

Explanation: The output data set is of unrecognized organization. It must be sequential.

System action: Processing does not begin until the panel entries are correct and the user presses 'ENTER'.

User response: The output data set must be sequential. Enter a sequential data-set-name (no member) in the output data set field.

DSNE375A   ACCESS TO INPUT DATA SET NOT AUTHORIZED

Explanation: A local installation-supplied exit has rejected access to the input data set specified by the user.

User response: Either specify the name of a data set for which authorization will be allowed, or arrange to authorize the original data-set-name. Resubmit the request.

DSNE376A   ACCESS TO OUTPUT DATA SET NOT AUTHORIZED

Explanation: A local installation-supplied exit has rejected access to the output data set.

User response: Either specify the name of a data set for which authorization will be allowed, or arrange to authorize the original data-set-name. Resubmit the request.

DSNE377A   INPUT DATA SET RECFM MUST BE F OR FB WITH LRECL 79 OR 80

Explanation: The user specified an input data set that does not have the required attributes (F or FB with LRECL 79 or 80).

Note: If the input data set has an LRECL of 79, SPUFI assumes the data set was exported by QMF™. Therefore, it is recommended that you should not use an LRECL of 79 for a data set that is not exported from QMF.

User response: Correct the input data set attributes, and try again.

DSNE378A   ASA CONTROL CHARACTER NOT ALLOWED ON INPUT DATA SET RECFM

Explanation: The user attempted to use an input data set containing forms control characters or machine control characters. These characters are not allowed on the input data set.

User response: Remove the forms control characters or machine control characters from the input data set, and try again.

DSNE379A   text from DYNALLOC

Explanation: An attempt to allocate a data set has failed. This message contains the secondary explanation for the failure.

User response: Correct the condition, and resubmit the subcommand.

DSNE381A   UNEXPECTED ALLOCATION RETURN CODE return-code, reason-code

Explanation: An unexpected return code from SVC 99 was received. The return code received is the DYNALLOC (SVC 99) return code. Refer to the appropriate MVS publication for explanations of the return and reason codes.

User response: Correct the condition, and resubmit the subcommand.

DSNE383A   INVALID OUTPUT DATA SET RECORD FORMAT. OPEN RETURN CODE rcode

Explanation: The data set format was invalid. Acceptable record formats are: F, FB, FBA, V, VB, or VBA.

User response: Enter a valid value in the record format field.

DSNE384A   UNEXPECTED DEALLOCATION RETURN CODE return-code, reason-code

Explanation: A deallocation request has returned an unexpected return and reason code.

User response: Refer to the appropriate MVS publication for explanations of the return and reason codes.

DSNE385A   ENTER MEMBER NAME. INPUT DATA SET IS PARTITIONED

Explanation: The input data set is partitioned, and the user did not enter the member name on the SPUFI main menu.

System action: Processing does not begin until the panel entries are correct. If any of the processing options are changed, this message is ignored, and processing will proceed according to the new settings.

User response: Enter the member name.

DSNE386A   MEMBER member-name NOT PRESENT IN SPECIFIED INPUT DATA SET

Explanation: The named member is not in the input data set.

System action: Processing does not begin until the panel entries are correct.

User response: Enter a correct member name or a correct data-set-name.

DSNE387I   DSNTIAR ERROR. RETURN CODE = *return-code*

Explanation: An internal DB2 error has occurred. DSNTIAR attempted to format a message to explain a nonzero SQL code, but the attempt failed.

System programmer response: Print the ISPF log, and collect a dump of the TSO address space.

User response: Notify the system programmer.

Related tasks:

▶ Displaying SQLCA fields by calling DSNTIAR (DB2 Application programming and SQL)

Related reference:

▶ Possible return codes from DSNTIAR (DB2 Application programming and SQL)

DSNE388A   INPUT DATA SET MUST BE SEQUENTIAL OR PARTITIONED

Explanation: The input data set organization is not recognized.

System action: Processing does not begin until the panel entries are correct.

User response: Enter a data-set-name with sequential or partitioned organization. It may be necessary to allocate a sequential or partitioned data set.

DSNE389A   A MEMBER HAS BEEN SPECIFIED FOR A SEQUENTIAL DATA SET

Explanation: The user has specified a member as input to SPUFI. However, the input data set specified is sequential.

System action: Processing does not begin until the panel entries are correct.

User response: Enter the data-set-name without attempting to specify a member.

DSNE390A   THE OUTPUT DATA SET ATTRIBUTES ARE INCONSISTENT

Explanation: This message indicates that the user made one of three possible errors in specifying block size and logical record length (LRECL):

- The user specified a record format (RECFM) of F, but the block size is not equal to the logical record length (LRECL)
- The user specified a record format (RECFM) of F, FB, or FBA, but block size is not evenly divisible by logical record length (LRECL).
- The user specified a record format (RECFM) of VB, but the block size is less than logical record length (LRECL) + 4. The user must allow 4 bytes for the block header.

User response: Specify logical record length (LRECL) and block size appropriately on the DSNESP02 panel.

DSNE391I   FREEMAIN ERROR IN MODULE module-name

Explanation: SPUFI encountered a FREEMAIN error in the specified module.

System action: This message is logged in the ISPF log data set. No abend occurs.

System programmer response: Print the ISPF log, and collect a dump of the TSO address space. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer or system administrator.

DSNE392I   GETMAIN ERROR IN MODULE module-name

Explanation: SPUFI encountered a GETMAIN error in the specified module.

System action: This message is logged in the ISPF log data set. User abend X'00C50003' is issued.

System programmer response: Ensure that enough storage is allocated to the TSO user.

User response: Notify the system programmer or system administrator.

DSNE393I   **SPF VDELETE ERROR IN MODULE DSNESM90. ERROR CODE = flag**

Explanation: An internal SPUFI error has occurred. CSECT DSNESM90 tried to delete an ISPF dialog variable that was not previously defined.

System action: This message is logged in the ISPF log data set. No abend is issued. However, if the problem persists, notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Record the error code 'flag'. It has no user significance, but it can assist the IBM Support Center in resolving the problem.

DSNE395I   **UNEXPECTED RETURN CODE FROM SPF SERVICE**

Explanation: A return code of 4 from ISPF DISPLAY or a return code of 8 from the ISPF editor was encountered.

System action: The message is displayed on the SPUFI panel and logged in the ISPF log data set. No user abend is issued.

System programmer response: If a new release of ISPF is used, determine whether the above return codes are now assigned meaning by ISPF.

User response: Notify the system programmer.

Problem determination: This message may be issued if a previously unassigned (by ISPF) return code from the ISPF DISPLAY or EDIT function is encountered. If a new release of ISPF is used, determine whether the above return codes are now assigned meaning by ISPF.

DSNE396A   **YOU CANNOT SPECIFY DEFER WHEN LEAVING SPUFI**

Explanation: You have attempted to leave SPUFI with outstanding database changes not yet committed or rolled back. You can't leave SPUFI unless a decision has been made to commit or rollback.

System action: The system is waiting for new input to the panel.

User response: Specify COMMIT or ROLLBACK.

DSNE601I   **SQL STATEMENTS ASSUMED TO BE BETWEEN COLUMNS nn AND nn**

Explanation: This message indicates which record columns in the input data set are scanned for SQL statements. For data sets of type COBOL, columns 8 through 72 are scanned. For STANDARD data set types, if the LRECL is 79 then columns 1 through 71 are scanned, and if the LRECL is 80 then columns 1 through 72 are scanned.

System action: This message is written to the SPUFI output data set along with other summary messages.

DSNE603E   **INPUT STATEMENT HAS UNBALANCED APOSTROPHES**

Explanation: An opening apostrophe was detected in an input record, but the end of data set was reached before a closing apostrophe was encountered.

System action: SPUFI processing stops. Correct the error, and start again.

User response: A closing apostrophe was probably omitted from a literal value. Add the missing apostrophe, and execute the input file again.

DSNE604E   **THE SQL STATEMENT EXCEEDS THE MAXIMUM ALLOWED LENGTH**

Explanation: This message indicates that the preceding SQL statement in the output data set exceeds the maximum length allowed. An SQL statement may span up to a maximum of 29127 input records.

System action: SPUFI processing stops. Correct the error, and start again.

User response: Edit the SQL statement in question so that it resides on fewer input records.

DSNE605E   THE INPUT DATA SET IS EMPTY

Explanation: The user has specified an input data set that does not contain any valid SQL statements.

System action: SPUFI processing stops. Correct the error, and start again.

User response: Edit SQL statements into the specified data set, or specify another input data set that contains valid SQL statements.

DSNE606E   INVALID USE OF AN SQL STATEMENT TERMINATOR

Explanation: An SQL statement terminator has been used improperly in an SQL input statement. Either two successive statement terminators were encountered, or a series of blanks followed by a terminator was detected. The proper use of the SQL statement terminator is to delimit the end of a valid SQL statement.

System action: SPUFI processing stops. Correct the error, and start again.

User response: Remove the extraneous terminator from the input data set.

DSNE607E   PREVIOUS INPUT RECORD HAS UNBALANCED SO/SI CHARACTERS

Explanation: A shift out (SO) character was detected in the previous input record with no corresponding shift in (SI) character in the record. The SO/SI characters delimit a DBCS string in the SPUFI input data set. SPUFI requires that SO/SI characters be paired within an input record.

System action: SPUFI processing stops.

User response: A closing SI character was probably omitted from a literal value. Add the missing SI character, and re-execute the input file.

DSNE609E   INPUT STATEMENT HAS UNBALANCED QUOTES

Explanation: An opening quote (") was detected in an input record, but a corresponding quote was not found at the end. Two sets of quotes are used to delimit a statement.

System action: SPUFI processing stops.

User response: A closing quote was probably omitted from a literal value. Add the missing quote, and execute the input file again.

DSNE610I   NUMBER OF ROWS DISPLAYED IS nn

Explanation: Execution of an SQL SELECT statement causes one or more rows of data to be displayed. This message appears in the output data set following the returned data for a SELECT. It gives a count, 'nn', of the number of rows displayed.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

DSNE611I   COLUMN HEADER *name* FOR COLUMN NUMBER *nn* WAS TRUNCATED

Explanation: An SQL SELECT statement was executed, but the specified column name, identified by 'nn' in the message, was truncated. This truncation occurred either because the column name was longer than the remaining record width or because the name was longer than the user-specified maximum field length.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

User response: If the truncation is caused by insufficient record width, use a larger LRECL for the output data set or specify fewer data columns to be returned on the SELECT statement.

If the truncation is caused by the user-specified maximum field length setting, use a larger value for this setting.

DSNE612I   **DATA FOR COLUMN HEADER name COLUMN NUMBER nn WAS TRUNCATED**

Explanation: An SQL SELECT statement was executed, but data for the specified column name, identified by 'nn' in the message, was truncated. This truncation occurred either because the data was longer than the remaining record width or because the data was longer than the user-specified maximum field length.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

User response: If the truncation is because of insufficient record width, use a larger LRECL for the output data set or specify fewer data columns be returned on the SELECT statement.

If the truncation is because of the user-specified maximum field length setting, use a larger value for this setting.

DSNE613I   **OUTPUT LRECL IS TOO SMALL TO DISPLAY THE FOLLOWING COLUMNS:**

Explanation: An SQL SELECT statement was executed, but the logical record length of the output data set was not large enough to display the column names and data values of all returned columns. The columns specified following this message were not displayed because of insufficient remaining record width.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

User response: This condition can be avoided by using a larger LRECL for the output data set or by specifying that fewer data columns be returned on the SELECT statement.

DSNE614I   **AUTOCOMMIT IS NO, NO CHANGES COMMITTED**

Explanation: The user delayed committing or rolling back changes made during execution of the input file until later in the session. This is done by specifying an AUTOCOMMIT option of 'NO'.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

User response: Commit or roll back these changes before ending the session.

DSNE615I   **NUMBER OF ROWS AFFECTED IS** *nn*

Explanation: This message is written to the output data set following execution of an SQL data change statement, or REFRESH statement. *nn* is a count of the number of rows that qualified to be deleted, inserted, or updated by that statement, excluding rows affected by either triggers or referential integrity constraints. If the previous SQL statement is a multiple-row FETCH, *nn* indicates the number of rows fetched.

Note: In the case of a mass delete of a table in a segmented table space, or a delete from a view when neither the DELETE statement nor the definition of the view included selection criteria, this message will not be displayed. In these cases, DB2 does not access nor count the individual rows.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

DSNE616I   **STATEMENT EXECUTION WAS SUCCESSFUL, SQLCODE IS** *sql-code*

Explanation: This message is written to the output data set following the successful execution of an SQL statement from the input data set. The *sql-code* value specifies the SQL return code for the statement.

System action: Input data set processing is completed, and this message is included in the SPUFI output.
Related concepts:

➦ SQL codes (DB2 Codes)

DSNE617I   **COMMIT PERFORMED, SQLCODE IS** *sql-code*

Explanation: When an input file is executed successfully and the user specified an AUTOCOMMIT option of 'YES', this message is written to the output data set to indicate the status of the COMMIT that was performed. The *sql-code* value specifies the SQL return code for the COMMIT.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

Related concepts:

 ⮕ SQL codes (DB2 Codes)

DSNE618I   ROLLBACK PERFORMED, SQLCODE IS *sql-code*

Explanation: When an error is encountered while executing a SPUFI input file, and the user specified an AUTOCOMMIT option of 'YES', this message will be written to the output data set to indicate the status of the ROLLBACK that was performed. *sql-code* specifies the SQL return code for the statement.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

Related concepts:

 ⮕ SQL codes (DB2 Codes)

DSNE620I   NUMBER OF SQL STATEMENTS PROCESSED IS nn

Explanation: This message is written to the output data set after processing all SQL statements in the current input data set. The term 'nn' is a count of the number of SQL statements processed.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

DSNE621I   NUMBER OF INPUT RECORDS READ IS nn

Explanation: This message is written to the output data set after processing all SQL statements in the current input data set. The term 'nn' is a count of the total number of records read from the input data set.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

DSNE622I   NUMBER OF OUTPUT RECORDS WRITTEN IS nn

Explanation: This message is written to the output data set after processing all SQL statements in the current input data set. The term 'nn' is a count of the total number of records written to the output data set.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

DSNE623I   SYNTAX ERROR AT RECORD xx, OFFSET xx

Explanation: A syntax error was detected in the SQL statement preceding this message in the output data set. 'xx' identifies the erroneous line number in the SQL statement, and 'yy' identifies the offset (in characters) of the error within that record.

System action: Input data set processing is completed, and this message is included in the SPUFI output.

User response: Correct the specified syntax error in the input data set, and execute the job again.

DSNE624I   UNABLE TO LOAD ISPLINK MODULE. SPUFI PROCESSING TERMINATED

Explanation: During SPUFI initialization processing, an error was encountered attempting to LOAD the ISPF interface module, ISPLINK. This module could not be located in the LNKLST or STEPLIB library concatenation.

System action: SPUFI processing is terminated.

User response: Rearrange your library concatenation so that the ISPLINK module can be located. This module resides in the ISPF load library.

DSNE625I   CONNECT TO LOCATION (*location-name***) PERFORMED, SQLCODE IS** *sql-code*

Explanation: The SPUFI user has indicated that the SQL statements defined in the input file are to be processed by the server at the remote location specified in the CONNECT LOCATION field of the SPUFI panel.

This message appears in the output file and indicates that an SQL CONNECT request was issued by SPUFI to the specified location name. The The *sql-code* value specifies the SQL return code associated with the CONNECT request.

System action: Subsequent processing of the SQL statements in the input file is based upon the successful

completion of the CONNECT request. If the request is unsuccessful, the SQL error number and all associated error messages are placed in the output file.

User response: Respond as indicated by the SQL code returned.

Related concepts:

➡ SQL codes (DB2 Codes)

DSNE626I   MAXIMUM OUTPUT LINES FOR SELECT STATEMENT REACHED (number), PROCESSING FOR CURRENT SELECT STATEMENT TERMINATED

Explanation: The maximum number of output lines to be displayed for a SELECT statement was reached. Processing of the current SELECT statement is terminated. The maximum number to display is specified by the SPUFI user on the CURRENT SPUFI DEFAULTS panel.

This message appears in the SPUFI output file.

System action: Processing of subsequent SQL statements in the input file continues.

User response: If the number of lines displayed is insufficient, you can increase the maximum number of lines to be displayed for SELECT statements on the CURRENT SPUFI DEFAULTS panel and re-execute the SELECT statement.

DSNE627I   *csect-id* **FOR SQLCODE =** *sql-code*, **USER REQUESTED:** *action*

Explanation: During PREPARE processing of the SQL statement, SPUFI received an SQLCODE (+495) and requested you to respond with the action to be taken by SPUFI (CONTINUE or BYPASS).

SPUFI does not consider this to be an SQL statement error and places this message in the SPUFI output file to document the SQLCODE received and the action you requested. *sql-code* specifies the SQL return code associated with the PREPARE request.

System action: SPUFI processing for this statement is based on *action* documented in the message:
* CONTINUE
  – Processing for the current SQL statement continues normally.
* BYPASS
  – Processing for the current SQL statement is terminated without error.

User response: Respond as indicated by the SQL code returned.

Related concepts:

➡ SQL codes (DB2 Codes)

Related information:

➡ +495 (DB2 Codes)

DSNE800A   NO DEFAULT VALUES WERE CHANGED. PRESS ENTER TO CONTINUE

Explanation: The user left panel DSNESP02 without changing it. This panel contains all the defaults that can be modified by the user.

System action: This message is presented on the SPUFI panel DSNESP01. SPUFI waits for the next user input.

User response: Press 'ENTER' if you want to continue.

DSNE803A   INPUT FILE WAS NOT CHANGED. PRESS ENTER TO CONTINUE

Explanation: The user has exited from the ISPF editor without modifying the input data set.

System action: The system waits for the next user input.

User response: Press 'ENTER' if you want to continue. The SPUFI panel variable settings can be changed at this point.

DSNE804A   **TO CONTINUE, C (COMMIT) OR R (ROLLBACK) MUST BE SPECIFIED**

Explanation: The user has specified the AUTOCOMMIT=NO option and now wants either to exit or to switch to another DB2 subsystem with outstanding work uncommitted on the current DB2 subsystem.

System action: DB2 waits for the user to specify either the COMMIT (C) or the ROLLBACK (R) option.

User response: Enter 'C' or 'R' to continue.

DSNE808A   **EDIT SESSION HAS COMPLETED. PRESS ENTER TO CONTINUE.**

Explanation: The user has exited from the ISPF editor.

System action: SPUFI waits form the next user input.

User response: Press 'Enter' if you want to continue. Press 'END' if you want to terminate your SPUFI session. The SPUFI panel variable settings can be changed at this point.

DSNE901I   **EXECUTION COMPLETE**

Explanation: Execution of a subcommand has successfully completed.

System action: The TSO attachment facility is ready for more input.

DSNE903I   **WARNING, DECLARATION HAS SAME NAME AS TABLE** table-name

Explanation: A DCLGEN subcommand has a 'STRUCTURE' parameter that is the same as the 'table name' specified in the message. The requested declaration has been created, but this could cause conflicts with internal names.

System action: DCLGEN execution continues.

User response: Choose a different 'STRUCTURE' parameter.

DSNE904I   **EXECUTION COMPLETE, MEMBER member-name REPLACED**

Explanation: A DCLGEN subcommand executed successfully, replacing the 'member-name' specified in the message.

System action: The TSO attachment facility is ready for more input.

DSNE905I   **EXECUTION COMPLETE, MEMBER member-name ADDED**

Explanation: A DCLGEN subcommand executed successfully, adding the 'member-name' specified in the message to the data set specified in the DCLGEN subcommand.

System action: The TSO attachment facility is ready for more input.

DSNE908E   *csect-name* **INVALID USE OF** *parameter* **IN PARAMETER LIST OF KEYWORD** *keyword*.

Explanation: An invalid parameter was specified in the parameter list of the specified keyword.

System action: Subcommand processing is terminated. The TSO attachment facility is ready for more input.

User response: Correct the parameter list in error, and resubmit the job.

Severity: 8

DSNE909E   *csectname* **WARNING, DBCSSYMBOL KEYWORD IS INVALID WITH LANGUAGE** *language*
                  KEYWORD IGNORED

Explanation: You specified the DBCSSYMBOL keyword in your DCLGEN invocation, but your LANGUAGE keyword value was not COB2. DBCSSYMBOL only applies to COB2 PICTURE clauses. DCLGEN will ignore this keyword.

System action: DCLGEN processing continues.

User response: In the future, do not specify the DBCSSYMBOL keyword unless the LANGUAGE keyword value is COB2.

DSNE910E   *csectname* **WARNING, THE OWNER KEYWORD WAS SPECIFIED WITH A QUALIFIED TABLE NAME. OWNER KEYWORD IGNORED**

Explanation: You specified both the OWNER keyword and a qualified table name for the TABLE keyword in your DCLGEN invocation. The table qualifier will be used as the table owner, and the OWNER keyword will be ignored.

System action: DCLGEN processing continues.

User response: In the future, specify either a qualified table name for the TABLE keyword, or an unqualified table name for the TABLE keyword along with the OWNER keyword.

DSNE920E   **member-name ALREADY EXISTS, SPECIFY ACTION(REPLACE) TO REPLACE**

Explanation: The DCLGEN user tried to overwrite an old declaration with a new one without specifying REPLACE on the 'ACTION' parameter.

System action: The TSO attachment facility is ready for more input.

User response: Change the 'ACTION' parameter to REPLACE (if that's actually what you want to do), and resubmit the command.

DSNE921E   **REQUIRED PARAMETER table-parameter IS MISSING**

Explanation: Your DCLGEN subcommand was missing the required TABLE parameter.

System action: Your DCLGEN subcommand was not processed. The DSN command processor issued a return code of 8. DSN is ready for more input.

User response: Resubmit the DCLGEN subcommand with a TABLE parameter.

DSNE922E   **TABLE NAME QUALIFIER OF qualifier IS TOO LONG. CANNOT EXCEED 8 CHARACTERS IN LENGTH**

Explanation: DCLGEN detected an invalid TABLE parameter. The table name qualifier (the left portion of the table name) was too long.

System action: Your DCLGEN subcommand was not processed. The DSN command processor issued a return code of 8. DSN is ready for more input.

User response: Resubmit the DCLGEN subcommand with a shorter table name qualifier.

DSNE923E   **CSECT** *csect-name* **FOUND AN UNRECOGNIZED LANGUAGE** *unknown*

Explanation: This is a DCLGEN subsystem error, not a user error. The language parameter *unknown* is not known to the subsystem.

System action: The TSO attachment facility is ready for more input.

System programmer response: Re-create the problem with DSN tracing turned on. Collect the trace stream in the DSNTRACE data set. Use the ABEND subcommand of the DSN command processor to collect a dump. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Report the problem to the system programmer.

DSNE924E   **PARSE FAILED IN CSECT** *csect-name* **PARAMETER parm, FLAG** *flag*

Explanation: This is a DB2 subsystem error, not a user error. The 'PARM' parameter was not correctly parsed. The flag specified was detected as an inappropriate value in a parse results field.

System action: The TSO attachment facility is ready for more input.

System programmer response: Re-create the problem with DSN tracing turned on. Collect the trace stream in the DSNTRACE data set. Use the ABEND subcommand of the DSN command processor to collect a dump. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Report the problem to the system programmer.

DSNE925E **BUFFER OVERFLOW IN CSECT csect-name WHILE WRITING string**

Explanation: This is a DCLGEN subsystem error. CSECT 'csect-name' detected a buffer overflow condition while attempting to write 'string'.

System action: The TSO attachment facility is ready for more input.

System programmer response: Re-create the problem with DSN tracing turned on. Collect the trace stream in the DSNTRACE data set. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

DSNE926E **COMMAND REJECTED -** *command* **NOT SUPPORTED UNDER** *command_processor*

Explanation: The user entered a TSO TIME or TEST command. These are not supported by DSN or DB2I.

System action: The user should leave DSN or DB2I, return to TSO, and retry the command. DSN can be run under TSO TEST. The TSO attachment facility is ready for more input.

User response: Enter the DSN subcommand, or leave DSN to execute the requested command.

DSNE927E **RUN SUBCOMMANDS REQUIRE (1)PROGRAM PARM OR (2)PLAN AND CP PARMS**

Explanation: RUN subcommands may contain (1) the 'PROGRAM' parameter and, optionally, the 'PLAN' parameter or (2) the 'PLAN' and command processor ('CP') parameters. Combinations of (1) and (2) are not allowed.

System action: The TSO attachment facility is ready for more input.

User response: Correct the subcommand, and resubmit the job.

DSNE928E **RUN SUBCOMMANDS CANNOT HAVE A PROGRAM PARM AND A CP PARM**

Explanation: A RUN subcommand cannot contain both the 'PROGRAM' and command processor ('CP') parameters. The subcommand should be resubmitted with only one of the parameters.

System action: The TSO attachment facility is ready for more input.

User response: Correct the subcommand, and resubmit the job.

DSNE929E **TABLE NAME LOCATION OF (location-name) IS TOO LONG. CANNOT EXCEED 16 CHARACTERS IN LENGTH.**

Explanation: DCLGEN detected an invalid AT(location) keyword parameter. The specified location name is too long.

System action: Your DCLGEN subcommand was not processed. The DSN command processor issues a return code of 8. The DSN is ready for more input.

User response: Resubmit the DCLGEN subcommand specifying a location name that does not exceed 16 characters.

DSNE930E **TABLE NAME OF table-name MUST BE QUALIFIED WHEN SPECIFIED WITH LOCATION OF location-name**

Explanation: DCLGEN detected an invalid TABLE keyword parameter. You must specify a qualified table name with the AT (location) keyword.

System action: Your DCLGEN subcommand was not processed. The DSN command processor issues a return code of 8. DSN is ready for more input.

User response: Specify a qualified table name and resubmit the DCLGEN command.

DSNE931E   *csect-name keyword-name (parameter_value)* **CONTAINS AN INVALID SYMBOL (symbol)**

Explanation: An invalid 'symbol' (symbol) was specified in the 'parameter_value' associated with the specified 'keyword-name'.

System action: Subcommand processing is terminated. The TSO attachment facility is ready for more input.

User response: Correct the keyword 'parameter_value' in error, and resubmit the job.

Severity: 8

DSNE932I   **WARNING, ONLY IBM SUPPLIED names SHOULD BEGIN WITH DSN**

Explanation: The user has issued a BIND, REBIND, or FREE subcommand, using a name or id that begins with DSN.

System action: Processing continues.

User response: In the future, do not use plan-names, collection-ids, or package -ids that begin with DSN. It would be a good idea to FREE the plan or package you just created and to BIND again with a new plan-name, collection-id, or package-id.

DSNE935I   **... IS THE DCLGEN COMMAND THAT MADE THE FOLLOWING STATEMENTS**

Explanation: This message is a part of normal DCLGEN output. It identifies the DCLGEN command that produced the declaration of which it is a part.

System action: This message is included in the DCLGEN output.

DSNE936I   **THE NUMBER OF COLUMNS DESCRIBED BY THIS DECLARATION IS nnn**

Explanation: This message is a part of normal DCLGEN output. It tells how many columns were in the table that was named in the DCLGEN command.

System action: This message is included in the DCLGEN output.

DSNE937I   **COBOL DECLARATION FOR TABLE table-name**

Explanation: This message is a part of normal DCLGEN output. It tells the name of the table for which the declaration was created.

System action: This message is included in the DCLGEN output.

DSNE938I   **PLI DECLARATION FOR TABLE table-name**

Explanation: This message is a part of normal DCLGEN output. It tells the name of the table for which the declaration was created.

System action: This message is included in the DCLGEN output.

DSNE939E   *csect-name* **THE COMBINATION ACQUIRE(ALLOCATE) AND RELEASE(COMMIT) IS NOT ALLOWED**

Explanation: The BIND or REBIND subcommand was submitted with conflicting parameters.You cannot specify ACQUIRE(ALLOCATE) and RELEASE(COMMIT) on the same subcommand.

System action: The subcommand is terminated and the DSN command processor reissues theDSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Related tasks:

➜ Choosing a RELEASE option (DB2 Performance)

Related reference:

➜ BIND PACKAGE (DSN) (DB2 Commands)

➜ BIND PLAN (DSN) (DB2 Commands)

➜ REBIND PACKAGE (DSN) (DB2 Commands)

➜ REBIND PLAN (DSN) (DB2 Commands)

➜ ACQUIRE bind option (DB2 Commands)

➜ RELEASE bind option (DB2 Commands)

DSNE940E   DEFER(PREPARE) INVALID WITH NODEFER(PREPARE)

Explanation: The BIND or REBIND subcommand was submitted with conflicting parameters. You cannot specify DEFER(PREPARE) and NODEFER(PREPARE) on the same subcommand because these actions conflict.

System action: The subcommand is terminated and the DSN command processor reissues the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Related reference:

➜ BIND PACKAGE (DSN) (DB2 Commands)

➜ BIND PLAN (DSN) (DB2 Commands)

➜ REBIND PACKAGE (DSN) (DB2 Commands)

➜ REBIND PLAN (DSN) (DB2 Commands)

➜ DEFER AND NODEFER bind options (DB2 Commands)

DSNE941I   DSNBIND IS A RESERVED PLAN NAME

Explanation: This message is generated when the plan name is 'DSNBIND' in a BIND statement.

System action: Subcommand processing is terminated. The TSO attachment facility is ready for more input.

User response: Change the plan-name and resubmit the job.

DSNE942E   *csect-name* REMOTE CONNECTION NAME *(parameter-value)* HAS INVALID FORMAT

Explanation: The REMOTE connection name *(parameter-value)* contains a < or> character in the starting or ending position of the parameter, but the parameter value is not bracketed by both characters. This message is also generated if the first character of the connection name is not an alphabetic character.

If the LU name format of the REMOTE keyword is to be used, the name must be specified as *<LU-name>*.

System action: Subcommand processing is terminated. DSN command processor reissues the DSN prompt to allow another subcommand to be entered.

User response: Correct the keyword *(parameter-value)* in error, and resubmit the job.

Severity: 8

Related reference:

➜ DSNH (TSO CLIST) (DB2 Commands)

DSNE943E   *csect-name* **REMOTE CONNECTION NAME** *(parameter_value)* **CANNOT EXCEED** *(length_value)* **CHARACTERS IN LENGTH**

Explanation: The REMOTE connection name *(parameter_value)* specified is greater than the length *(length_value)* permitted for the REMOTE keyword.

This message is issued if the REMOTE location name format exceeds 16 characters, or the LU name format (<LU name>) exceeds 10 characters in length.

System action: Subcommand processing is terminated. DSN command processor reissues the DSN prompt to allow another subcommand to be entered.

User response: Correct the keyword *(parameter_value)* in error and resubmit the job.

Severity: 8

Related reference:

➥ DSNH (TSO CLIST) (DB2 Commands)

DSNE944I   **WARNING, BECAUSE YOUR TABLE OR COLUMN NAMES CONTAIN LOWER CASE OR NON-ALPHABETIC CHARACTERS, YOU WILL NEED TO USE THE SQL (NOT PL/I) INCLUDE STATEMENT TO INCLUDE DCLGEN OUTPUT INTO YOUR PROGRAM**

Explanation: DCLGEN had to wrap your table or column name(s) in quotes so DB2 could process them. Quotes are not recognized by the PL/I compiler. To avoid compilation errors, use the SQL INCLUDE statement ('EXEC SQL INCLUDE *member-name*') to merge the DCLGEN output with your source program.

System action: DCLGEN processing continues.

User response: If you have not coded the SQL INCLUDE, do so before the next compile.

Related reference:

➥ DCLGEN (DECLARATIONS GENERATOR) (DSN) (DB2 Commands)

➥ INCLUDE (DB2 SQL)

DSNE947I   **QUOTE PARAMETER IS ONLY VALID WITH COBOL. IT HAS BEEN CHANGED TO APOST**

Explanation: You specified the DCLGEN QUOTE parameter and a non-COBOL language parameter. This combination is not valid. The QUOTE parameter was specified as either a parameter on the DCLGEN subcommand or a default set for your site at DB2 installation time.

The DCLGEN processor changed the QUOTE setting (for this run) to APOST.

System action: DCLGEN processing continues and should produce valid output.

User response: If you used QUOTE on the DCLGEN command, you may want to change it to APOST before submitting it again.

DSNE948I   *csect-id* **FURTHER** *subcommand* **PROCESSING WILL NOT BE ATTEMPTED**

Explanation: A DSN command processor subcommand (indicated in the message) has encountered an error which stopped DSN from processing this subcommand. The subcommand will not complete normally. This message is accompanied by other messages explaining the exact nature of the problem.

System action: Subcommand processing terminates. The DSN prompt is reissued, indicating DSN is ready for another subcommand.

User response: Read the accompanying messages for help in deciding what to do next.

DSNE949E   **COMMIT PROCESSING ABNORMAL TERMINATION, REASON CODE reason_code**

Explanation: The DSN command processor has initiated a 'prepare to commit' request in DB2. The request is rejected when DB2 cannot commit the work performed

System action: Commit processing is terminated.

User response: Examine the reason code associated with this message, and respond as indicated before resubmitting the job.

Problem determination: Reference the associated reason code to determine why the 'prepare to commit' was not successful.

DSNE950E   DCLGEN DOES NOT SUPPORT THE lang LANGUAGE

Explanation: A DCLGEN user attempted to specify a LANGUAGE parameter, 'lang', that DCLGEN does not support.

System action: DCLGEN processing terminates.

User response: Correct the LANGUAGE parameter and resubmit the DCLGEN subcommand.

DSNE952I   C DECLARATION FOR TABLE table-name

Explanation: This message is a part of normal DCLGEN output. It tells the name of the table for which the declaration was created.

System action: This message is included in the DCLGEN output.

**DSNE953I   csect-name PRECISION OF DECIMAL COLUMN colname EXCEEDS THE MAXIMUM PRECISION
              FOR THE lang LANGUAGE.**

Explanation: DCLGEN was building a COBOL or PL/I language declaration for a table when it encountered a decimal column with a precision greater than the precision supported by the language.

System action: DCLGEN set the return code to 4 and processed the rest of the table.

User response: This is a warning. Be aware that the declaration for column 'colname' causes a compiler error. You must edit the declaration before including it in an application program.

**DSNE954I   csect-name USE OF THE DCLGEN COLSUFFIX(YES) OPTION WILL RESULT IN A MIXED FIELD
              NAME FOR COLUMN colname. THE lang LANGUAGE DOES NOT SUPPORT MIXED FIELD
              NAMES.**

Explanation: The COLSUFFIX(YES) option and a double-byte NAMES prefix were specified in a DCLGEN invocation, but column colname contains single-byte characters. This combination would result in a generated field name with a mixture of double-byte and single-byte characters, but the lang language does not support mixed field names.

System action: DCLGEN set the return code to 4 and generated the field name as requested.

User response: This is a warning. Be aware that the declaration for column colname causes a compiler error. You must edit the declaration before including it in an application program.

**DSNE955I   csect-name COLSUFFIX(YES) WAS SPECIFIED WITHOUT A NAMES PREFIX. COLUMN NAMES
              WILL BE USED FOR FIELD NAMES**

Explanation: When COLSUFFIX(YES) is specified in a DCLGEN invocation, DCLGEN generates field names by concatenating the NAMES prefix with the column names. However, no NAMES prefix was specified.

System action: DCLGEN set the return code to 4 and used the column names as field names.

User response: This is a warning. Specify a NAMES prefix when you specify COLSUFFIX(YES).

DSNE956A   A FIELD NAME PREFIX MUST BE SPECIFIED WHEN COLUMN SUFFIX VALUE IS YES.

Explanation: When YES is specified for the COLUMN SUFFIX field on the DCLGEN panel, DCLGEN generates field names by concatenating the value in the FIELD NAME PREFIX field with the column names. However, no value was specified for FIELD NAME PREFIX.

System action: The DCLGEN panel generated this error message.

User response: Specify a FIELD NAME PREFIX value, or change the COLUMN SUFFIX value to NO.

DSNE957I	**THE LENGTH OF ONE OR MORE NAMES EXCEEDS THE STANDARD MAXIMUM ALLOWED BY THE** *language* **LANGUAGE**

Explanation: DCLGEN was building a COBOL or PL/I language declaration for a table when it generated a host variable name that is longer than the language permits.

language
        The language that the declaration uses: COBOL or PL/I

The new declaration might cause a compiler error.

System action: DCLGEN issues return code to 4 and processes the rest of the table.

User response: If you want to use this declaration in an application, edit the declaration so that the length of the host name variable is within the permitted range for the language.

DSNE958I	**DCLGEN DOES NOT SUPPORT THE SQL DATA TYPE** *data-type* **(SQLTYPE** *sql-type***) FOR THE** *language* **LANGUAGE**

Explanation: The specified table or view contains a column with a data type that is either not supported by or not recognized by DCLGEN.

data-type
        The data type of the column.

sql-type Numeric representation of the data type.

language
        The programming language that was used for the data declaration: COBOL, PL/I, C, or C++.

System action: DCLGEN issues return code 8 and finishes processing the table. A declaration for the table or view is created, but it is incomplete and will not compile successfully without modifications.

User response: If the reported data type is UNKNOWN, contact IBM Software Support and provide the SQLTYPE value and programming language from the message.

For all other data types, verify that the data type that you want to use is supported by DCLGEN. If the data type that you originally chose is not supported, you might be able to identify an alternate data type. If you want to use the table or view declaration that DCLGEN created, update the generated declaration by either removing the column or changing the data type.

DSNE960E	**PARAMETER STRING EXCEEDS ALLOWABLE LIMIT OF 100 (DECIMAL) CHARACTERS**

Explanation: The PARMS option parameter-string of the RUN subcommand cannot exceed 100 (decimal) characters in length.

System action: The TSO attachment facility is ready for more input.

User response: Correct the PARMS option and resubmit the job.

DSNE964E	*csect-name* **INVALID SYNTAX FOR** *keyword* **KEYWORD PARAMETER** *parameter*

Explanation: A parameter that does not follow DB2 naming conventions was specified for *keyword*.

System action: Subcommand processing is terminated. The DSN command processor reissues the DSN prompt to allow you to enter another subcommand.

User response: Correct the keyword parameter value in error, and resubmit the command.

Related reference:

➡ BIND and REBIND options (DB2 Commands)

➡ BIND PLAN (DSN) (DB2 Commands)

➡ BIND PACKAGE (DSN) (DB2 Commands)

➡ REBIND PLAN (DSN) (DB2 Commands)

➡ REBIND PACKAGE (DSN) (DB2 Commands)

DSNE965E   *parameter-value* **IS NOT A VALID** *keyword* **KEYWORD PARAMETER**

Explanation: The *parameter-value* is not a valid parameter value for the keyword.

System action: Subcommand processing terminates. The DSN command processor reissues the DSN prompt to allow you to enter another subcommand.

User response: Correct and reissue the subcommand.

Related reference:

➡ BIND and REBIND options (DB2 Commands)

➡ DYNAMICRULES bind option (DB2 Commands)

➡ BIND PACKAGE (DSN) (DB2 Commands)

➡ BIND PLAN (DSN) (DB2 Commands)

➡ REBIND PACKAGE (DSN) (DB2 Commands)

➡ REBIND PLAN (DSN) (DB2 Commands)

DSNE966E   *connection-type* **KEYWORD OMITTED FROM THE ENABLE|DISABLE DECLARATION**

Explanation: The *connection-type* keyword was not specified in either the ENABLE or DISABLE keywords and is required if connection-names are specified for the *connection-type*.

System action: Subcommand processing is terminated. DSN command processor reissues the DSN prompt to allow you to enter another subcommand.

User response: Correct the connection type parameter with values that are supported on the ENABLE and DISABLE keywords and reissue the subcommand.

Related reference:

➡ ENABLE and DISABLE bind options (DB2 Commands)

DSNE967E   *keyword-name* **KEYWORD CANNOT BE SPECIFIED MORE THAN ONCE PER RUN**

Explanation: Either the BIND PACKAGE MEMBER or LIBRARY keywords specified more than one parameter value. Only one member or library name is permitted.

System action: Subcommand processing is terminated. The DSN command processor reissues the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Severity: 8

Related reference:

➡ BIND PACKAGE (DSN) (DB2 Commands)

➡ BIND PLAN (DSN) (DB2 Commands)

➡ MEMBER bind option (DB2 Commands)

➡ LIBRARY bind option (DB2 Commands)

DSNE969E   *csect-name keyword1* **IS MUTUALLY EXCLUSIVE WITH** *keyword2*

Explanation: Both the PKLIST and NOPKLIST keywords, cannot be specified on one invocation of the REBIND PLAN subcommand. Both the ENABLE and DISABLE keywords cannot be specified on one invocation of the BIND/REBIND PLAN/PACKAGE subcommand.

System action: Subcommand processing is terminated. DSN command processor will reissue the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Severity: 8

Related reference:

➡ PKLIST and NOPKLIST bind options (DB2 Commands)

➡ ENABLE and DISABLE bind options (DB2 Commands)

➡ BIND PLAN (DSN) (DB2 Commands)

➡ BIND PACKAGE (DSN) (DB2 Commands)

➡ REBIND PLAN (DSN) (DB2 Commands)

➡ REBIND PACKAGE (DSN) (DB2 Commands)

DSNE970E   **EITHER MEMBER OR PKLIST MUST BE SPECIFIED, BOTH ARE ALSO ALLOWED**

Explanation: The BIND PLAN subcommand requires either the MEMBER or PKLIST keyword, and neither was specified. Either keyword or both may be specified.

System action: Subcommand processing is terminated. DSN command processor will reissue the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Severity: 8

Related reference:

➡ BIND PLAN (DSN) (DB2 Commands)

➡ MEMBER bind option (DB2 Commands)

➡ PKLIST and NOPKLIST bind options (DB2 Commands)

DSNE971E   **EITHER MEMBER OR COPY BUT NOT BOTH MUST BE SPECIFIED**

Explanation: The BIND PACKAGE subcommand requires either the MEMBER or COPY keyword; both are not allowed on the same subcommand invocation.

System action: Subcommand processing is terminated and the DSN command processor will reissue the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand

Severity: 8

Related reference:

➦ BIND PACKAGE (DSN) (DB2 Commands)

DSNE974E *csect-name keyword1* **KEYWORD IS INVALID WITH** *keyword2* **KEYWORD.**

Explanation: The *keyword2* cannot be specified with *keyword1*.

System action: Subcommand processing is terminated. The DSN command processor reissues the DSN prompt to allow another subcommand to be entered.

User response: Modify one of the keywords on the command and resubmit the job.

Related reference:

➦ BIND PACKAGE (DSN) (DB2 Commands)

➦ BIND PLAN (DSN) (DB2 Commands)

DSNE975E *csect-name keyword* **KEYWORD IS ONLY VALID WHEN COPY KEYWORD IS PRESENT**

Explanation: A keyword was issued that may only be specified in conjunction with the COPY keyword.

keyword
> COPYVER or OPTIONS

System action: Subcommand processing is terminated. The DSN command processor will reissue the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Severity: 8

Related reference:

➦ BIND PACKAGE (DSN) (DB2 Commands)

DSNE976E **LIBRARY KEYWORD IS ONLY VALID WHEN MEMBER KEYWORD IS PRESENT**

Explanation: The LIBRARY keyword may only be specified in conjunction with the MEMBER keyword.

System action: Subcommand processing is terminated. The DSN command processor will reissue the DSN prompt to allow another subcommand to be entered.

User response: Enter the corrected subcommand.

Severity: 8

Related reference:

➦ BIND PACKAGE (DSN) (DB2 Commands)

Chapter 4. DSNF messages

Message identifiers that begin with DSNF communicate information about the message generator.

DSNF001I   MESSAGE GENERATOR INITIALIZATION PARAMETERS NOT FOUND. DEFAULTS ASSUMED

Explanation: The message generator was unable to access the routing code initialization parameter defined by the DSN6SYSP macro. Default values defined by that macro are assumed.

System action: Subsystem initialization continues.

Operator response: Notify the system programmer.

System programmer response: It might be necessary to change the DSN6SYSP macro and replace the DSN6SYSP member of the DSNIPARM directory.

Related tasks:

➡ Defining DB2 initialization parameters: DSNTIJUZ (DB2 Installation and Migration)

DSNF700I   *csect-name message-id* HAS BEEN ISSUED

Explanation: An unsuccessful attempt has been made to issue the message 'message-id'. This message is issued only if the requested message could not be found in the DB2 message directory.

System action: The system proceeds as though the requested message had been issued.

Operator response: Notify the system programmer.

System programmer response: Refer to this manual for actual message text. You need the message number ('message-id'). If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Chapter 5. DSNG messages

Message identifiers that begin with `DSNG` communicate information about the database descriptor manager.

DSNG001I   **EDM** *stype* **CAN NOT BE DECREASED BELOW THE VALUE** *isize* **SPECIFIED AT DB2 STARTUP**

Explanation: The EDM *stype* storage cannot be decreased below the size that was specified at DB2 startup.

System action: Processing continues.

DSNG002I   **EDM** *storage-type* **HAS AN INITIAL SIZE** *initial-size*, **REQUESTED SIZE** *requested-size*, **AND AN ALLOCATED SIZE** *allocated-size*

Explanation: The size of an EDM storage pool has been changed.

storage-type
> The type of EDM storage pool that was updated. Possible values are:

Value	Description
DBD CACHE	EDM DBD pool
RDS ABOVE	EDM RDS pool above
RDS BELOW	EDM RDS pool below
SKELETON	EDM statement pool
STMT POOL	EDM skeleton pool

initial-size
> The storage size prior to the request for a change.

requested-size
> The requested storage size.

allocated-size
> The storage size in effect immediately after the request for a change.

If the request was for an increase to the EDM pool size, the *allocated-size* value is the storage available to satisfy the request.

If the request was for a decrease to the EDM pool size, the *allocated-size* value is the *initial-size* value reduced by the amount that could be released immediately. When the *allocated-size* value is larger than the *requested-size* value, the difference is marked to be released when it is no longer referenced.

System action: Processing continues.

User response: No action is required.

Related concepts:

➥ EDM storage (DB2 Performance)

DSNG003I   **INSUFFICIENT VIRTUAL STORAGE TO EXPAND EDM** *stype* **STORAGE**

Explanation: Insufficient virtual storage exists to expand the EDM *stype* storage as specified.

System action: The EDM *stype* storage is expanded only to the size allowed by the available virtual storage.

DSNG004I   EDM *stype* **HAS A FINAL ALLOCATED SIZE** *asize*

Explanation: This message is issued in response to a request to decrease the EDM *stype* storage. The message is issued when the requested new level is reached.

stype   The EDM pool that is being reduced. Possible values are Pool, DBD Pool, or Statement Pool.

asize   The EDM *stype* storage that is available.

System action: Processing continues.

DSNG005I   **DBD IS INCONSISTENT WITH THE CATALOG. UNABLE TO FIND A ROW IN SYSTABLES WITH DBID** *dbid* **AND OBID** *obid*.

Explanation: This message is issued in response to a request to load a database with DBID *dbid*. This message is issued when the row is not found in SYSTABLES.

dbid   DBID of the database that is being loaded.

obid   OBID of a table in the database.

System action: DBD could not be loaded.

Operator response: Based on *dbid* and *obid* from the message, verify that the row does not exist in the Catalog. REPAIR DBD TEST or DIAGNOSE can be run to gather more diagnostic information. Once you have determined that the Catalog contains the correct information, REPAIR DBD REBUILD can be run to rebuild the DBD.

Severity: 8 (error).

DSNG006I   **DBD IS INCONSISTENT WITH THE CATALOG. UNABLE TO FIND A ROW IN SYSCOLUMNS WITH DBID** *dbid*, **OBID** *obid*, **AND COLUMN NUMBER** *colno*.

Explanation: This message is issued in response to a request to load a database with DBID *dbid*. This message is issued when the row is not found in SYSCOLUMNS.

dbid   DBID of the database that is being loaded.

obid   OBID of a table in the database.

colno   Column number in the table.

System action: DBD could not be loaded.

Operator response: Based on *dbid*/*obid*/*colno* from the message, verify that the row does not exist in the Catalog. REPAIR DBD TEST or DIAGNOSE can be run to gather more diagnostic information. Once you have determined that the Catalog contains the correct information, REPAIR DBD REBUILD can be run to rebuild the DBD.

Severity: 8 (error).

DSNG007I   **DB2 CATALOG LEVEL (**catalog-level**) CODE LEVEL (**code-level**) MODE (**system-mode**)**

Explanation: This message is issued when the catalog is loaded. It displays the DB2 catalog and code release levels and a mode indicator.

catalog-level
　　　　Three-character indicator of the DB2 catalog release level.

　　　　810　　　Version 8

　　　　910　　　Version 9

　　　　A10　　　Version 10

code-level
　　　　Three-character indicator of the DB2 code release level.

　　　　810　　　Version 8

　　　　910　　　Version 9

　　　　A10　　　Version 10

system-mode
> Two- or three-character indicator of the current mode of the system:

> **C***n*   Conversion mode

> **C***n**   Conversion mode, but the system detects that it has been to a higher level

> **E***n*   Enabling-new-function mode

> **E***n**   Enabling-new-function mode, but the system detects that it has been to a higher level

> **N**   New-function mode

System action: Processing continues.

User response: No action is required.

DSNG008I   **EDM RDS BELOW STORAGE HAS AN INITIAL LIMIT** *initial-limit* **AND A NEW LIMIT** *new-limit***.**

Explanation: The limit for EDM RDS BELOW storage has been changed. The new limit takes effect immediately.

initial-limit
> The limit prior to the request for a change.

new-limit
> The limit in effect after the request for a change.

A `resource unavailable` condition can occur when the limit is decreased and the amount of storage in use was already at or near the new limit.

System action: Processing continues.

User response: No action is required.

Related tasks:

⏩ Designing EDM storage space for performance (DB2 Performance)

DSNG009I   *csect-name* **TABLE SPACE DSNDB01.SPT01 IS IN REORG PENDING STATE**

Explanation: The change to the inline length of the SPTSEC_DATA column of SPT01 is complete, and the new length is shorter than the current length.

csect-name
> The name of the control section that issued the message.

System action: The SPT01 table space is placed in restrictive REORG-pending (REORP) state.

User response: Reorganize the SPT01 table spaces to remove the REORP state. When the SPT01 table space is in REORP state binds cannot be processed.

Related reference:

⏩ SPT01 INLINE LENGTH field (SPT01_INLINE_LENGTH subsystem parameter) (DB2 Installation and Migration)

DSNG010I   *csect-name* **TABLE SPACE DSNDB01.SPT01 IS IN REORG PENDING STATE**

Explanation: The SPT01 inline length was successfully changed.

csect-name
> The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Related reference:

⏩ SPT01 INLINE LENGTH field (SPT01_INLINE_LENGTH subsystem parameter) (DB2 Installation and Migration)

DSNG011I

DSNG011I   *csect-name* **SPT01 INLINE LENGTH CHANGE UNSUCCESSFUL, LENGTH** *size*

Explanation: The SPT01 inline length was not changed.

csect-name
>   The name of the control section that issued the message.

size   The size of the inline length in bytes.

System action: Processing continues.

User response: Check the console for a DSNT500I message.

Related reference:

➡ SPT01 INLINE LENGTH field (SPT01_INLINE_LENGTH subsystem parameter) (DB2 Installation and Migration)

Chapter 6. DSNH messages

Message numbers that begin with DSNH communicate information about the DB2 precompiler or the DB2 coprocessor.

References to the DB2 precompiler will also apply to the DB2 coprocessor, unless otherwise mentioned. The DB2 coprocessor is invoked through the compiler.

Message line numbers

The line number *nnnn* in DSNH messages is generated by assigning sequential numbers to the original source. The numbers are printed on the SQL DB2 precompiler source listing (if the SOURCE option is specified to the precompiler). If you are using the DB2 coprocessor, the error messages and their line numbers, if applied, are printed on the compiler output listing. The line number points to the beginning of the SQL statement. The column number, if it can be determined, is a relative number from the beginning of the SQL statement that is in error.

'COL *cc*' in the messages identifies the source column in which the condition was detected. The indicated column, '*cc*', usually specifies the location of the column containing the first character of the associated source token. However, if the precompiler cannot determine this location, it is the beginning of the statement. Because the detected error might be similar to a valid, but different, construction, the SQL parsing might indicate the wrong location for the error.

Message severity codes

One of the characters I, W, E, S, or U is shown at the beginning of each precompiler message as an aid to the programmer. These characters indicate the severity level as follows:

I	0	Informational
W	4	Warning
E	8	Error
S	12	Severe error
U	16	Unrecoverable error

When unsupported non-DB2 statements are detected during SQL(ALL) precompilation, the return code will be forced to 4 if the return code was 8, and no host-related errors were detected. This is done so that the application can proceed with the compilation and link-edit steps. This allows the syntax checking to be delayed until bind. If a severity 8 precompiler error message is not a host-related error, the precompiler treats the error as a warning and place a W instead of an E in the precompiler output listing. In this instance, the precompiler also issues DSNH049I to indicate that the actual return code 8 from the precompiler was changed to 4.

The following precompiler messages with severity levels of 8 are not changed from errors to warnings when SQL(ALL) is in effect:
- DSNH051I
- DSNH080I

- DSNH290I
- DSNH291I
- DSNH292I
- DSNH312I
- DSNH313I
- DSNH314I
- DSNH504I

Source tokens

Tokens from your source program do not always appear in messages exactly as they appear in your program. For example, leading zeros are removed from integer tokens.

DSNH003I   **S** *csectname* **FILE** *ddname* **PARAMETERS INCORRECT**

Explanation: The file *ddname* has invalid or incorrect parameters, such as logical record length.

System action: The precompiler is terminated.

User response: Correct the data set or library parameters.

Severity: 12 (severe error)

DSNH004I   **S** *csectname type* **COULD NOT BE FOUND**

Explanation: The precompiler could not find the indicated type of statement in the application program source. This message is only issued for HOST(COBOL) and HOST(COB2). The *type* is DATA DIVISION, WORKING-STORAGE SECTION, or PROCEDURE DIVISION. This error can cause other errors.

System action: Processing continues. Additional errors might result.

User response: Ensure that the host language statement in your application program is correct and that the statement is present within the proper margins.

Severity: 12 (severe error)

DSNH005I   **U** *csectname* **FILE** *ddname* **NO MEMBER NAME SPECIFIED**

Explanation: The *ddname* must refer to a partitioned data set, and the member name must be specified in the allocation. This helps to prevent accidental erasure of the DBRM library.

System action: The precompiler is terminated.

User response: Correct the JCL or TSO allocate command to specify a member name.

Severity: 16 (unrecoverable error)

DSNH006I   **U** *csectname* **FILE** *ddname* **COULD NOT BE OPENED**

Explanation: The file identified in the message is required, but it could not be opened.

System action: The precompiler is terminated.

System programmer response: Ensure that the SYSIN data is provided. Also ensure that SYSLIB is defined if the application program contains SQL INCLUDE statements.

User response: Ensure that the file is present in your JCL.

Severity: 16 (unrecoverable error)

DSNH007I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **ILLEGAL CHARACTER IGNORED: '***c***' ('***hh***'X)**

Explanation: The character identified in the message as *c* is not permitted. The hexadecimal representation for the character is *hh*.

System action: The character is ignored.

User response: Correct the character. You might need to place the character within quotation marks or apostrophes. Check to ensure that the CCSID in which the source is written matches the CCSID option for the Precompiler. For example, if the CCSID 1047 codepoints for left and right square brackets (hexadecimal 'AD' and 'BD') are used in a C source program, ensure that the Precompiler CCSID option is CCSID(1047).

Severity: 8 (error)

DSNH008I   **S** *csectname* **LINE** *nnnn* **COL** *cc* **NESTED INCLUDE NOT SUPPORTED**

Explanation: While processing an INCLUDE, another nested INCLUDE statement was encountered.

System action: The nested INCLUDE statement is ignored.

User response: Remove the nested INCLUDE statements; they are not allowed in an included file. The DB2 precompiler permits only one level of INCLUDE statements.

Severity: 12 (severe error)

DSNH009I   **S** *csectname* **LINE** *nnnn* **COL** *cc* **MEMBER** *name* **COULD NOT BE INCLUDED**

Explanation: The identified member name was not found or could not be read.

System action: Processing continues without the data from that member.

User response: Ensure that the member is in the library and that its name is spelled correctly. Check and correct any JCL errors. They are likely to cause other errors. Ensure that the data set and library parameters are correct.

Severity: 12 (severe error)

DSNH010I   **S** *csectname* **LINE** *nnnn* **COL** *cc* **UNTERMINATED STRING CONSTANT BEGINNING** *text*

Explanation: The constant identified in the message as '*text*' is not terminated properly.

System action: The string is treated as if it were terminated by the end of file.

User response: Terminate the string constant. Check for and correct any missing or extra quotation marks and apostrophes. These errors are likely to cause others. Some statements were probably missed because of missing or extra string delimiters.

Severity: 12 (severe error)

DSNH011I   **S** *csectname* **LINE** *nnnn* **COL** *cc* **UNTERMINATED COMMENT**

Explanation: A comment is not terminated properly. This message is only issued for HOST(PLI) and HOST(C).

System action: The string is treated as if it were terminated by the end of file.

User response: Ensure that the comment terminator is typed properly and that it is within the specified margins. This error is likely to cause others. Some statements were probably missed because of the missing comment terminator.

Severity: 12 (severe error)

DSNH012I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **INVALID CONTINUATION**

Explanation: The continuation does not follow the host language rules.

System action: Processing continues. Additional errors might result.

User response: Follow the host language continuation rules and embedded SQL continuation rules. Check the margin option. This error is most common in assembler language in cases where a continuation begins before the

continuation column (the continuation column is column 16 by default).

Severity: 4 (warning)

DSNH013I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **SQL STATEMENT FOUND OTHER THAN IN THE FIRST PROGRAM (OR MAIN PROGRAM)**

Explanation: This message is for VS COBOL II only. If any SQL statement is found in the source after the first program in a BATCH COMPILE or the end of the main program of a nested program, this message is issued.

System action: The precompiler is terminated.

User response: Move up the SQL statement and corresponding host variable declaration to the main program or to the first program and resubmit the job.

Severity: 8 (error)

DSNH014I   **S** *csectname* **LINE** *nnnn* **COL** *cc* **SQL STATEMENT IS FOUND IN AN INAPPROPRIATE PLACE**

Explanation: If any SQL statement is found in the COBOL DECLARATIVES section, this message is issued.

System action: Processing continues. Additional errors might result.

User response: Move the SQL statement out of the COBOL DECLARATIVES section and resubmit the job.

Severity: 12 (severe error)

DSNH016I   **E** *csectname* **LINE** *nnnn* **COL** *cc token* **REQUIRED**

Explanation: The required character or keyword identified as *token* is missing.

System action: The token is assumed to be present in order to continue checking the statement syntax.

User response: Correct the SQL statement. Determine if you skipped either this keyword or another keyword earlier in the SQL statement. The assumed token is printed in the SYSTEM error message. This token might not be what you intended, even though it is the only symbol allowed in the clause at this point.

Severity: 8 (error)

DSNH019I   **W** *csectname* **OPTION: INVALID OPTION** *option*

Explanation: The identified option was requested, but it is not a precompiler option.

System action: The option is ignored.

User response: Check the valid precompiler options to determine if your spelling is correct for this option. This problem might cause additional errors if the APOST, APOSTSQL, COMMA, HOST, MARGINS, QUOTESQL, or TWOPASS option is needed but not specified. The other options are either defaults or affect only the listing. Correct the problem and run the precompile again.

Severity: 4 (warning)

DSNH021I   **W** *csectname* **OPTION: SUBOPTION NOT PERMITTED WITH OPTION** *option*

Explanation: The identified option cannot have suboptions, but one or more were specified.

System action: The suboptions are ignored.

User response: None is required. However, you should remove the suboption.

Severity: 4 (warning)

DSNH022I   **W** *csectname* **OPTION: UNMATCHED ')' TAKEN AS ','**

Explanation: An unmatched ending parenthesis was found before the end of the option string.

System action: The parenthesis is assumed to be a comma.

User response: Balance the parentheses.

Severity: 4 (warning)

DSNH023I   **W** *csectname* **OPTION: SUBOPTION MISSING FOR OPTION** *option*

Explanation: The specified option requires a suboption, but one was not provided.

System action: The option is ignored.

User response: Provide a suboption or remove the option. This problem might cause additional errors if the HOST or MARGINS option is needed but not specified. The other options are either defaults or affect only the listing. Correct the problem and run the precompile again.

Severity: 4 (warning)

DSNH024I   **W** *csectname* **OPTION: SUBOPTION** *suboption* **INVALID FOR OPTION** *option*

Explanation: The identified suboption is not allowed for the option specified.

System action: The entire option is usually ignored.

User response: Refer to the 'options used' list in the precompiler output listing to see if the option was used. Remove the suboption.

Severity: 4 (warning)

DSNH025I   **W** *csectname* **OPTION: EXCESSIVE SUBOPTIONS FOR OPTION** *option*

Explanation: Too many suboptions were given for the identified option.

System action: The excess suboptions are ignored.

User response: Remove excess suboptions.

Severity: 4 (warning)

DSNH026I   **W** *csectname* **OPTION: INVALID MARGINS FOR HOST LANGUAGE**

Explanation: The host language does not permit the specified margins.

System action: The MARGINS option is ignored.

User response: Ensure that the source statements are within the proper margins and that the margins are specified correctly. The margins for COBOL must be 8, 72. The margins for FORTRAN must be 1, 72.

Severity: 4 (warning)

DSNH027I   **W** *csectname* **SUBOPTION** *suboption* **TOO LONG, TRUNCATED AFTER** *numofchar* **CHARS**

Explanation: The version ID is greater than *numofchar* bytes long. The maximum length for the version ID is 64 bytes.

System action: The option is honored, but the version ID is truncated after 64 bytes.

User response: Refer to the 'options used' list in the precompiler output to see what version ID was used.

Severity: 4 (warning)

DSNH028I   **W** *csectname* **ATTACH SUBOPTION** *suboption* **NOT PERMITTED WITH HOST (FORTRAN) SPECIFIED.**

Explanation: The identified suboption for precompiler option ATTACH cannot take effect when the application is a FORTRAN application because the interface code for FORTRAN is always DSNHFT.

System action: The option is ignored.

User response: None is required. However, you can either code ATTACH(TSO) or specify no ATTACH precompiler option

Severity: 4 (warning)

DSNH029I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **INTO CLAUSE REQUIRED**

Explanation: SELECT statements embedded in an application program must have an INTO clause to denote where the results of the SELECT are to be placed. Dynamic SELECT statements do not permit the INTO clause.

System action: The statement is replaced by host language statements that, if executed, set an error code in the SQLCA.

User response: Add the INTO clause to the SELECT statement, and precompile the application program again.

Severity: 8 (error)

DSNH030I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT CONTAINS WRONG NUMBER OF 'INTO' VARIABLES**

Explanation: The number of variables specified in an INTO clause is not the same as the number of select-list elements (taking into account expansion of host structures and '*' or 'table.*' select entries according to the declarations supplied).

System action: Processing continues.

User response: Ensure the included table declarations are correct. If the declarations are correct, this statement causes an error when the application program is bound. If the declarations are incorrect, you can proceed, but the declarations should be corrected. Otherwise, you need to correct the SQL statement and precompile again.

Severity: 4 (warning)

DSNH031I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT CREATES A COLUMN WITH NAME** *name1* **WHICH DOES NOT MATCH DECLARED NAME** *name2*

Explanation: The column name specified in a CREATE statement (*name1*) is not the same as the name of the corresponding column in the DECLARE TABLE statement (*name2*).

System action: Processing continues.

User response: Ensure that the DECLARE TABLE matches the CREATE TABLE. You can use the cross-reference and source listings to help find the two statements.

Severity: 4 (warning)

DSNH032I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT CREATES A COLUMN** *name* **WHOSE ATTRIBUTES DO NOT MATCH THE DECLARATION**

Explanation: The column *name* is specified in a CREATE statement with a different data type or length than was specified for the corresponding column in the DECLARE TABLE statement.

System action: Processing continues.

User response: Correct the DECLARE TABLE or CREATE TABLE statement. If the DECLARE TABLE statement is incorrect, some warnings might also be incorrect or might not be generated when needed. If the CREATE TABLE statement is incorrect, you must correct it.

Severity: 4 (warning)

DSNH033I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT CREATES MORE OR LESS COLUMNS THAN WERE DECLARED**

Explanation: The number of columns specified in a CREATE statement is different from the number of columns specified in the DECLARE TABLE statement.

System action: Processing continues.

User response: Ensure that the DECLARE TABLE matches the CREATE TABLE. If the DECLARE TABLE statement is incorrect, some warnings might also be incorrect or might not be generated when needed. If the CREATE TABLE statement is incorrect, you must correct it.

Severity: 4 (warning)

DSNH034I   W *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT INSERTS MORE OR LESS VALUES THAN THE NUMBER OF COLUMNS DECLARED**

Explanation: The number of values specified in an INSERT operation is different from the number of columns specified in the DECLARE TABLE statement for the receiving table.

System action: Processing continues.

User response: Insert the correct number of values. If the DECLARE TABLE statement is incorrect, this warning might also be incorrect. If the INSERT operation is incorrect, you must correct it.

Severity: 4 (warning)

DSNH038I   S *csectname* **LINE** *nnnn* **COL** *cc* **PROGRAM OR SUBPROGRAM CONTAINS MORE THAN 32767 SQL STATEMENTS**

Explanation: The number of SQL statements in an application program or a FORTRAN subprogram is greater than 32767.

System action: The precompiler terminates.

User response: Divide the application program or subprogram into smaller programs or reduce the SQL statements in each program unit so that the maximum number of SQL statements is not exceeded.

Severity: 12 (severe error)

DSNH041I   W *csectname* **OPTION: MISSING HOST OPTION**

Explanation: The host language was not specified in the OPTION field.

System action: HOST(COBOL) is assumed.

User response: Specify the host language if it is not COBOL. It is probably best to specify the host language and remove all warnings.

Severity: 4 (warning)

DSNH042I   W *csectname* **OPTION:** *option* **OPTION INVALID FOR HOST LANGUAGE**

Explanation: The option identified in the message is not valid. The QUOTE and COMMA options are not permitted with HOST(ASM), HOST(PLI), or HOST(C). The TWOPASS option is not permitted with HOST(COBOL).

System action: The specified option is ignored.

User response: Remove the invalid option, or correct the host specification.

Severity: 4 (warning)

DSNH049I   W *csectname* **RETURN CODE IS FORCED TO 4 BECAUSE OPTION SQL(ALL) IS IN EFFECT, ORIGINAL RETURN CODE WAS** *nnn*

Explanation: One or more SQL-related error messages were detected by the precompiler, but because of option SQL(ALL), the return code is forced to 4 in order to let the application proceed with compilation.

The SQL syntax checking is deferred until bind time because the source might contain non-DB2 SQL syntax that the DB2 precompiler does not recognize and flags as an error. The DBRM produced by this precompilation cannot be bound successfully by DB2 if there are SQL syntax errors or non-DB2 SQL syntax.

The return code remains untouched if there are host-related syntax or semantic errors in the source.

System action: Processing continues.

User response: Ensure that the intention of this precompilation is for executing it in a non-DB2 system.

Severity: 4 (warning)

DSNH050I   I *csectname* **WARNINGS HAVE BEEN SUPPRESSED DUE TO LACK OF TABLE DECLARATIONS**

Explanation: One or more warning messages were suppressed because no table declarations were found before the first SQL statement that referenced a table or column.

System action: Processing continues. Suppression is discontinued if a table declaration is found.

User response: Insert (or INCLUDE) table declarations before they are referenced in order to permit table and column checking.

Severity: 0

DSNH051I   E *csectname* **LINE** *nnnn* **COL** *cc sqltype name* **PREVIOUSLY DECLARED OR REFERENCED**

Explanation: The variable *sqltype* is TABLE, CURSOR, or STATEMENT. The identifier *name* was already defined with the given *sqltype*. The definition might have occurred implicitly. For example, *name* might have been found in the FROM clause of a SELECT, causing it to be defined as a table.

System action: The name is installed in the symbol table as a duplicate.

User response: Remove duplicate definitions, or correct the references. Avoid implicit definitions by inserting (or including) table declarations before they are referenced. Check the number of passes used. Use the cross-reference and source listings to help resolve this type of error.

Severity: 8 (error)

DSNH052I   E *csectname* **LINE** *nnnn* **COL** *cc* **COLUMN** *name* **IS ALREADY DEFINED IN TABLE** *table-name*

Explanation: The identified name is already defined as a column in the table named *table-name*. The definition might have occurred implicitly. For example, *table-name* might have been found in a select-list, causing *name* to be defined as a column within *table-name* if *table-name* was not declared as a host variable.

System action: The name is installed in the symbol table as a duplicate.

User response: Remove duplicate definitions, or correct the references. Avoid implicit definitions by inserting (or including) table declarations before they are referenced.

Severity: 8 (error)

DSNH053I   W *csectname* **NO SQL STATEMENTS WERE FOUND**

Explanation: No SQL statements were found in the precompiler input file. An empty DBRM member was written.

System action: Processing continues.

User response: If you embedded SQL statements in your application program and none were found, make sure your column margins are properly defined for your application program.

Severity: 4 (warning)

DSNH057I   E *csect-name* **LINE** *line-number* **COL** *column number* **RETURN STATEMENT IN AN SQL FUNCTION MUST RETURN A VALUE**

Explanation: A RETURN statement is specified without a value in an SQL function. The return value must be specified.

csect-name
>   The name of the control section that issued the message.

line-number **and** *column number*
>   The location of the error in the source program.

System action: The statement can not be processed.

User response: Add a return value to the statement, and try the request again.

Severity: 8 (error)

DSNH060I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **INVALID** *type* **SPECIFICATION:** *spec*

Explanation: The *type* is either SCALE or LENGTH. The *spec* is the given SCALE or LENGTH specification in an SQL declaration. Length and scale specifications must be unsigned integers. The scale specification, where permitted, cannot exceed the length. The length specification cannot be 0.

System action: Processing continues.

User response: Correct the length or scaling.

Severity: 8 (error)

DSNH078I   **E** *csect-name* **LINE** *line-number* **COL** *column number* **PARAMETER NAMES MUST BE SPECIFIED FOR ROUTINE** *routine-name*

Explanation: The routine does not have parameter names specified for all parameters.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

routine-name
> The name of the routine.

When the SQL routine is defined with LANGUAGE SQL, a parameter name is required for each parameter that is defined. For native SQL procedures, LANGUAGE SQL is optional, but for external SQL procedures, LANGUAGE SQL is required.

System action: The statement cannot be processed.

Programmer response: Ensure that each defined parameter includes a parameter name.

Severity: 8 (error)

DSNH080I   **E** *csectname* **LINE** *nnnn* **COL** *cc* *usage* **VARIABLE** *name* **IS NOT** *type* **TYPE**

Explanation: The *usage* is INDICATOR or STRING. The *type* is SMALLINT, VARCHAR, or CHARACTER. The data type of the variable *name* is not acceptable. All indicator variables must be small integers. PREPARE and EXECUTE strings must be variable character strings, except for FORTRAN, which requires character type.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Specify a host variable of the proper type.

Severity: 8 (error)

DSNH081I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT DOES NOT CONFORM TO ANS SYNTAX STANDARD - REASON CODE** *ccc*

Explanation: When the precompiler option STDSQL(YES) is in effect, the SQL syntax is checked to see if it conforms to the ANSI/ISO SQL standard of 1992. Reason codes are:

000   Aggregate function with DISTINCT as an arithmetic-operation operand does not conform to the ANSI/ISO SQL standard of 1992.

001   FOR UPDATE OF as part of a DECLARE CURSOR statement does not conform to the ANSI/ISO SQL standard of 1992.

System action: Processing continues.

System programmer response: Check the SQL statement.

User response: You have the option of changing or not changing the syntax to conform to the ANS syntax standard.

Severity: 4 (warning)

DSNH083I   I *csectname* **LINE** *nnnn* **COL** *cc* **IBM SQL RESERVED WORD** *ibmkeywd* **SPECIFIED AS AN IDENTIFIER NAME**

Explanation: When the precompiler option SQL(ALL) is in effect, IBM SQL reserved words should not be used as identifier names. Using IBM SQL reserved words limits the portability of the statement.

System action: Processing continues.

User response: If possible, change the identifier name to a non-IBM SQL reserved word so that the statement conforms to the IBM SQL standard. Not changing the identifier name means that the program is less portable within the IBM SQL environment.

Severity: 0 (informational)

DSNH084I   W *csectname* **LINE** *nnnn* **COL** *cc* **UNACCEPTABLE SQL STATEMENT**

Explanation: The referenced SQL statement is not allowed in this situation, but it will not cause an error. Depending on the option specified at precompile time, one of the following occurred.

- The BEGIN DECLARE SECTION(*), END DECLARE SECTION(*), and UPDATE STATISTICS statements are flagged as not compatible.

  * For C applications, BEGIN and END DECLARE SECTION are not flagged.

  * For other languages, the statement is only allowed with precompiler option STDSQL(YES).

- Using the STDSQL(YES) precompiler option, the INCLUDE SQLCA statement is flagged as a warning.

System action: The BEGIN DECLARE SECTION, END DECLARE SECTION, and INCLUDE SQLCA statements are ignored. The other statements are replaced by host language statements that will, if executed, set a warning code or SQLCODE -84 in the SQLCA.

User response: Unless this application program is used in a portable manner, remove the statement.

Severity: 4 (warning)

DSNH088I   W *csectname* **LINE** *nnnn* **COL** *cc* **THIS STATEMENT WILL** *action* **AN ENTIRE TABLE**

Explanation: The *action* is either DELETE or UPDATE. The referenced SQL statement has no WHERE clause and will delete or update all rows of the specified table. This message is printed as a warning to the programmer.

System action: Processing continues.

User response: Be sure that you intend to delete or update all rows, or add a suitable WHERE clause.

Severity: 4 (warning)

DSNH090I   S *csectname* **LINE** *nnnn* **COL** *cc* **HOST VARIABLE NOT PERMITTED HERE:** *name*

Explanation: Host variable references are not permitted in this type of SQL statement. The *name* is invalid.

System action: The statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Specify a literal or a column name rather than a host variable. The colon (:) indicates that the name that follows is a host variable. Remove it to specify a column name.

Severity: 12 (severe error)

DSNH091I   I *csectname* **LINE** *nnnn* **COL** *cc* **EBCDIC CCSID used for string-expression.**

Explanation: In a PL/1 application program, a string-expression was specified for a PREPARE or EXECUTE IMMEDIATE statement, and the precompiler-generated structures for this string-expression specify an EBCDIC CCSID.

System action: Processing continues.

User response: If you intended that the string-expression should be treated as a string expression, then no action is required.

If the string-expression consists of just one host variable, and you intended that the string-expression should be treated as a host variable, then the host variable must be preceded by a colon, and at least one DECLARE VARIABLE statement must be contained in your program source.

If you add a DECLARE VARIABLE for any host variable in your program source, and re-precompile your application, the variable previously treated as a string-expression by the precompiler will be treated as a host variable and this message will not be issued.

Severity: 0 (informational)

DSNH097I   **E** *csectname* **LINE** *nnnn* **COL** *cc*. **THE USE OF LONG VARCHAR OR LONG VARGRAPHIC IS NOT ALLOWED IN THIS CONTEXT.**

Explanation: The statement attempted to use the LONG VARCHAR or LONG VARGRAPHIC syntax. This syntax cannot be used for the following statements:
- CAST syntax
- CREATE DISTINCT TYPE
- CREATE FUNCTION
- CREATE PROCEDURE
- ALTER FUNCTION
- COMMENT ON FUNCTION
- GRANT EXECUTE ON FUNCTION
- REVOKE EXECUTE ON FUNCTION

System action: The statement cannot be executed.

User response: Correct and reissue the failing statement.

Severity: 8 (error)

DSNH101I   **S** *csectname* **THE STATEMENT IS TOO LONG OR TOO COMPLEX.**

Explanation: The statement cannot be executed because it exceeds the system limits for either length or complexity.

System action: The statement cannot be executed.

User response: Break the statement into shorter or less complex SQL statements.

Severity: 16

DSNH102I   **E** *csectname* **STRING CONSTANT IS TOO LONG. STRING BEGINS** *string*

Explanation: The string constant beginning with *string* has a length greater than the maximum allowed length. Most strings have these limits:
- For a hexadecimal constant (X, GX, or UX), the number of hexadecimal digits must not exceed 32704.
- Any other character string constant must be short enough so that its UTF-8 representation requires no more than 32704 bytes.
- Any other graphic string constant must be short enough so that its UTF-8 representation requires no more than 32704 bytes.
- The sum of the lengths of the UTF-8 representations of the constant values in an ENDING AT clause must not exceed 765 bytes. This clause applies to CREATE TABLE, ALTER TABLE, CREATE INDEX, or ALTER INDEX.

In some contexts, however, the limit is smaller.

For special registers, the allowable length depends on the particular special register. A string that exceeds the limit can be specified only through assignment from a host variable. If this release of DB2 increases the maximum allowed length of a string constant, then an attempt to exploit the new length before using the New Function Mode produces this SQL return code.

Two consecutive string delimiters are used to represent one string delimiter within the character string, but these count as 2 bytes when calculating the lengths of character string constants.

System action: If the SQL statement containing the string is a declaration, it is discarded. If the SQL statement containing the string is executable, it is replaced by host language statements that, if executed, set an error code in the SQLCA.

User response: Correct the string. If a shorter string is acceptable, reduce the length of the string constant. If a long string is required, a host variable is required.

Severity: 8 (error)

Related concepts:

➦ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

➦ Limits in DB2 for z/OS (DB2 SQL)

DSNH103I **E** *csectname constant* **IS AN INVALID NUMERIC CONSTANT**

Explanation: The *token* was found. It begins with a digit, but it is not a valid integer, decimal, or float literal. Identifiers cannot begin with a digit except for the WHENEVER statement with HOST(FORTRAN), and the identifiers in COBOL.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Check the indicated token for validity. For a character literal, use apostrophes or quotation marks. For a number, remove the invalid characters.

Severity: 8 (error)

DSNH104I **E** *csectname* **LINE** *nnnn* **COL** *cc* **ILLEGAL SYMBOL "***token***". SOME SYMBOLS THAT MIGHT BE LEGAL ARE:** *token-list*

Explanation: A syntax error was detected where the symbol "*token*" occurs in the SQL statement. The list of symbols that might be legal shows some alternate symbols that could possibly be correct at that point, if the preceding part of the statement is entirely correct.

However, the preceding part of the statement might be incorrect. For example, if an important keyword is omitted, DB2 detects the error later, and not always immediately after the point where the keyword should appear. The list of alternate symbols are only suggestions.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the statement and precompile the program again.

Severity: 8 (error)

DSNH105I **E** *csectname* **LINE** *nnnn* **COL** *cc***, INVALID STRING CONSTANTS**

Explanation: Incorrect graphic or mixed data literals were found on line *nnnn*, column *cc*.

System action: The literal is ignored and the SQL statement containing the problem is discarded.

System programmer response: Check for a quote, an apostrophe, shift-out (so) or shift-in (si) characters, G or N characters, or the number of characters within the literal.

User response: Specify the correct format of literals.

Severity: 8 (error)

DSNH107I **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE NAME** *name* **IS TOO LONG. MAXIMUM ALLOWABLE SIZE IS** *size*

Explanation: The identified *name* values exceeded the maximum supported length.

If this release of DB2 increases the maximum allowed length of a name, then an attempt to exploit the new length when the value of the NEWFUN option is NO will produce this message.

System action: If the SQL statement containing the problem is a declaration, it is discarded. If the SQL statement containing the problem is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the name length.

Severity: 8 (error)

Related concepts:

➡ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

➡ Limits in DB2 for z/OS (DB2 SQL)

DSNH109I   **E** *csect-name* **LINE** *nnnn* **COL** *cc clause-type* **CLAUSE NOT PERMITTED**

Explanation: A clause was used where it is not permitted.

csect-name
>The name of the control section that issued the message.

nnnn **and** *cc*
>The location of the clause, stated as a line number and column number.

clause-type
>The type of clause.

Possible causes for this message can include:
- A SELECT INTO statement cannot include a FOR UPDATE OF clause, because position is not retained.
- SELECT clauses used in cursor declarations or subselects cannot have an INTO clause because the INTO clause is on the FETCH statement.
- A SELECT INTO statement cannot specify a KEEP UPDATE LOCKS clause, except in a USE AND KEEP UPDATE LOCKS clause.
- CREATE VIEW statements cannot include any of the following clauses:
  - INTO
  - ORDER BY
  - FOR UPDATE
  - USING
  - LARGE
  - CORRELATION NAME
- The UNION operator is not allowed in an embedded SELECT clause.
- A CREATE TABLESPACE statement cannot specify LOCKPART without the NUMPARTS option.
- An ALTER TABLESPACE statement cannot specify LOCKPART for a non-partitioned table space.
- A table space with LOCKPART YES cannot be altered to LOCKSIZE TABLESPACE.
- A table space with LOCKSIZE TABLESPACE cannot be altered to LOCKPART YES.
- The LARGE clause without the NUMPARTS option is not allowed in a CREATE TABLESPACE statement.
- DESCRIBE INPUT statement can not have USING clause.
- The table being updated in a POSITIONED UPDATE statement can not be assigned a correlation name.
- On a CREATE INDEX statement, the INCLUDE clause is not allowed for any of the following indexes:
  - A non-unique index
  - A partitioning index, when index-controlled partitioning is being used
  - An XML index
  - An extended index

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Remove the clause. You might need to use a cursor to perform the processing. The ORDER and FOR

UPDATE clauses are accepted on the DECLARE CURSOR statement. The INTO clause is accepted on the FETCH statement. The LARGE clause is accepted on a CREATE TABLESPACE statement with the NUMPARTS option.

Severity: 8 (error)

Related reference:

⇨ Statements (DB2 SQL)

DSNH110I   **E** *csectname* **INVALID HEXADECIMAL CONSTANT BEGINNING** *constant*

Explanation: The string constant that begins with the specified *string* is invalid because it has at least one of the following errors:

- the string constant contains one or more characters that are not valid hexadecimal digits
- the string constant does not have an even number of digits
- for a UX or GX string constant, the number of digits is not a multiple of 4

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the string constant.

Severity: 8 (error)

DSNH111W   **W** *csectname* **LINE** *nnnn* **COL** *cc* **THE SUBPAGES OPTION IS NOT SUPPORTED FOR TYPE 2 INDEXES**

Explanation: You cannot use the SUBPAGES option for type 2 indexes.

System action: The option is ignored; processing continues.

Programmer response: Remove the SUBPAGES option to get rid of the warning.

Severity: 4 (warning)

DSNH113I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **INVALID CHARACTER FOUND IN** *string*, **REASON CODE -** *nnn*

Explanation: The string contains an invalid character. It can be an SQL ordinary identifier name, a host variable name, or a DBCS comment.

Certain SQL identifiers (for example names of buffer pools, databases, and plans) must contain only uppercase alphabetic or national characters and numerics; the first character must be alphabetic or national.

The following reason codes apply to SBCS identifiers:

000   An invalid character was found in the SBCS identifier (including the case in which a DBCS identifier was used where only an SBCS identifier is allowed).

The following reason codes apply to DBCS identifiers or comments:

101   An odd number of bytes exists between the shift-out and the shift-in character.

102   Either a shift-in or shift-out character is missing.

103   DBCS blanks X'4040' are not allowed.

104   There are no characters between the shift-out and the shift-in characters.

105   Shift-out cannot be the first byte of the DBCS character between the shift-out and the shift-in characters.

System action: Processing is terminated.

User response: Correct the name.

Severity: 8 (error)

DSNH115I   E *csectname* **LINE** *nnnn* **COL** *cc* **ILLEGAL USE OF COMPARISON OPERATOR** *operator*

Explanation: The syntax is not correct for this type of comparison. A simple comparison such as '>' must not be followed by a list of items. The ANY and ALL comparisons must precede a subselect, rather than an expression or a list of items.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax.

Severity: 8 (error)

Related reference:

➡ Statements (DB2 SQL)

DSNH117I   E *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT INSERTS THE WRONG NUMBER OF VALUES**

Explanation: The number of entries in a value-list is not the same as the number of columns specified in an INSERT statement.

System action: Processing continues.

User response: Correct the syntax.

Severity: 8 (error)

Related reference:

➡ Statements (DB2 SQL)

DSNH120I   E *csectname* **LINE** *nnnn* **COL** *cc* **FUNCTION USAGE NOT PERMITTED:** *fname*

Explanation: A WHERE clause or SET clause contains an aggregate function, possibly as a result of a reference to a column that is derived from an aggregate function in a view definition. An aggregate function is allowed in a WHERE clause only if that clause appears within a subquery of a HAVING clause.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax. A subselect clause might be needed to produce the function result.

Severity: 8 (error)

Related concepts:

➡ Queries (DB2 SQL)

Related reference:

➡ subselect (DB2 SQL)

DSNH125I   E *csectname* **LINE** *nnnn* **COL** *cc* **INVALID ORDER BY COLUMN NUMBER:** *number*

Explanation: The identified column number is either less than 1 or greater than the number of items in the select-list.

System action: The cursor declaration is discarded.

User response: Correct the column number.

Severity: 8 (error)

Related concepts:

➡ Queries (DB2 SQL)

DSNH126I   E *csectname* **LINE** *nnnn* **COL** *cc* **ORDER BY CANNOT BE USED WITH A FOR UPDATE OF CLAUSE**

Explanation: The ORDER BY clause makes the cursor incapable of being updated.

System action: The cursor declaration is discarded.

User response: This is a DB2 restriction. You can have either an ordered cursor or a cursor that can be updated. You cannot specify both attributes for the same cursor. You might need to change your application logic.

Severity: 8 (error)

DSNH129I   E *csectname* **LINE** *nnnn* **COL** *cc* **TOO MANY TABLES SPECIFIED IN SQL STATEMENT**

Explanation: In a subselect, (including all subqueries), there can be no more than 225 references to table names.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Reduce the number of table references in each subselect. Each reference to a single table counts separately. The count only applies to table/view references in the FROM list. Each view is only counted one time.

Severity: 8 (error)

Related reference:

➥ subselect (DB2 SQL)

➥ from-clause (DB2 SQL)

➥ Limits in DB2 for z/OS (DB2 SQL)

DSNH130I   E *csectname* **LINE** *nnnn* **COL** *cc* **THE ESCAPE CLAUSE '*x*' CONSISTS OF MORE THAN ONE CHARACTER**

Explanation: The literal *x* used in the ESCAPE clause is too long. Only one SBCS or one DBCS character is allowed.

System action: The corresponding SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the string. The literal string for the ESCAPE clause can only be one SBCS or DBCS character.

Severity: 8 (error)

DSNH131I   E *csectname* **LINE** *nnnn* **STATEMENT WITH LIKE PREDICATE HAS INCOMPATIBLE DATA TYPES**

Explanation: If the column name at the left of LIKE or NOT LIKE is a character type, the expression at the right must be a character type. If the column name is a graphic type, the expression at the right must be a graphic type.

System action: The statement is ignored.

User response: Check the data type of every operand.

Severity: 8 (error)

DSNH132I   E *csectname* **LINE** *nnnn* **COL** *cc* **INVALID LIKE PREDICATE**

Explanation: The value following LIKE must be an alphanumeric host variable or literal, graphic host variable or literal, or special character literal.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax of the LIKE predicate.

Severity: 8 (error)

Related concepts:

➡️ Language elements (DB2 SQL)

Related reference:

➡️ LIKE predicate (DB2 SQL)

DSNH142I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE SQL STATEMENT IS NOT SUPPORTED**

Explanation: The *token* was found. An SQL statement was detected that is not supported by the database. The statement might be valid for other IBM relational database products or it might be valid in another context. For example, statements such as VALUES and SIGNAL SQLSTATE can appear only inside a trigger.

System action: The SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Change the syntax of the SQL statement or remove the statement from the program.

Severity: 8 (error)

DSNH170I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **NUMBER OF ARGUMENTS SPECIFIED FOR** *function-name* **IS INVALID**

Explanation: An SQL statement specified the scalar function *function-name* with either too many or too few arguments.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that, if executed, sets an error code in the SQLCA.

User response: Correct the function call.

Severity: 8 (error)

Related reference:

➡️ Scalar functions (DB2 SQL)

DSNH171I   **E** *csect-name* **LINE** *nnnn* **COL** *cc* **THE DATATYPE, LENGTH, OR VALUE OF ARGUMENT** *nn* **OF** *function-name* **IS INVALID**

Explanation: The data type, length, or value of the argument *nn* of a *function-name* scalar function is incorrect.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that, if executed, sets an error code in the SQLCA.

User response: Correct the argument.

Severity: 8 (error)

Related reference:

➡️ Scalar functions (DB2 SQL)

DSNH182I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **AN ARITHMETIC EXPRESSION WITH A DATETIME VALUE IS INVALID**

Explanation: The specified arithmetic expression contains an improperly used date/time value or labeled duration.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the indicated expression.

Severity: 8 (error)

Related concepts:

➡ Date arithmetic (DB2 SQL)

➡ Time arithmetic (DB2 SQL)

➡ Timestamp arithmetic (DB2 SQL)

DSNH184I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **AN ARITHMETIC EXPRESSION WITH A DATETIME VALUE CONTAINS A PARAMETER MARKER**

Explanation: The specified expression contains a parameter marker improperly used with a date/time value.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the indicated expression.

Severity: 8 (error)

Related concepts:

➡ Date arithmetic (DB2 SQL)

➡ Time arithmetic (DB2 SQL)

➡ Timestamp arithmetic (DB2 SQL)

DSNH198I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **EMPTY SQL STATEMENT IGNORED**

Explanation: The referenced SQL statement has no text between the EXEC SQL and the statement terminator.

System action: The statement is ignored.

User response: Correct the statement, or remove it.

Severity: 8 (error)

DSNH199I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **INVALID KEYWORD** *keyword*; **VALID SYMBOLS ARE:** *token list*

Explanation: A syntax error was detected at the keyword specified in the message. As an aid to the programmer, a partial list of valid tokens is provided in the message. Only those tokens that fit are listed. The error might be earlier in the statement, but the syntax of the statement appears to be valid up to this point.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Check the SQL statement in the area of the indicated keyword. A colon or SQL delimiters might be missing. Correct the SQL statement.

Severity: 8 (error)

DSNH203I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT REFERENCES COLUMN** *column name*, **WHICH IS DECLARED IN MORE THAN ONE OF THE SPECIFIED TABLES**

Explanation: The column *column name* was referenced without sufficient qualification. Type checking cannot be done.

System action: Processing continues.

User response: If the table declares are correct and the column name is not unique, this creates an error in the BIND step. Correct the DECLARE TABLE statement, or use the correct table name to qualify the column name.

Severity: 4 (warning)

DSNH204I   W *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT REFERENCES UNDECLARED TABLE** *table-name*

Explanation: The table definition was not provided. It is optional, but checking cannot be done without the table information. (BIND checks against the actual table during later processing.)

System action: Processing continues.

Severity: 4 (warning)

DSNH205I   W *csectname* **LINE** *nnnn* **COL** *cc* **COLUMN** *column-name* **WAS NOT DECLARED IN TABLE** *table-name*

Explanation: The column name *column-name* is not contained in the table declarations. The name specified in a DISTINCT function must reference a column of the table. The value expression of an ALL function must reference a column of the table.

System action: Processing continues.

User response: If the table declares are correct and the column name is not in the table, this creates an error in the BIND step. Correct the DECLARE TABLE statement or the column name.

Severity: 4 (warning)

DSNH206I   W *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT REFERENCES COLUMN** *column name* **WHICH IS NOT DECLARED IN THE SPECIFIED TABLE(S)**

Explanation: The column *column name* is not contained in any TABLE declaration for the tables specified in the statement.

System action: Processing continues.

System programmer response: Check the spelling of the column name. Ensure that all the tables needed were specified in the statement.

User response: If the table declares are correct and the column name is not in the tables, this creates an error in the BIND step. Correct the DECLARE TABLE statements or the column name.

Severity: 4 (warning)

DSNH207I   E *csectname* **LINE** *nnnn* **COL** *cc* **ILLEGAL ORDERBY SPECIFICATION**

Explanation: If the select expression involves a UNION operator, the ORDER BY list must use numeric constants rather than variables.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax.

Severity: 8 (error)

Related concepts:

Queries (DB2 SQL)

Related reference:

order-by-clause (DB2 SQL)

fullselect (DB2 SQL)

DSNH251I   E *csectname* **LINE** *nnnn* **COL** *cc* **TOKEN** *name* **IS NOT VALID**

Explanation: The location name cannot contain alphabetic extenders for national languages (that is, #, @, and $ in the United States).

System action: Processing continues.

User response: Correct the content of the location name.

Severity: 8 (error)

DSNH270I   FUNCTION NOT SUPPORTED

Explanation: The statement cannot be processed because it violates one or more of the following restrictions:

- DESC is not allowed in CREATE INDEX if the index is an XML index or expression-based index.
- RANDOM is not allowed in CREATE INDEX if the index is an XML index or expression-based index.
- STACKED is not allowed on GET DIAGNOSTICS for external SQL procedures
- A FETCH statement cannot include WITH CONTINUE or CURRENT CONTINUE within an SQL routine.
- The only options allowed for ALTER FUNCTION ALL VERSIONS are SECURED and NOT SECURED. Any other option is flagged with an error.

System action: The statement cannot be processed.

User response: Correct the violation and issue the statement again.

Severity: 8 (error)

DSNH290I   E *csectname* **LINE** *nnnn* **COL** *cc* **NO VALID 'END DECLARE SECTION' FOUND**

Explanation: The user's source program contains a BEGIN DECLARE SECTION, but no corresponding END DECLARE SECTION was found.

System action: Processing continues.

User response: Check the application program and ensure that a valid END DECLARE SECTION statement is present.

Severity: 8 (error)

DSNH291I   E *csectname* **LINE** *nnnn* **COL** *cc* **NO VALID 'BEGIN DECLARE SECTION' FOUND**

Explanation: The user's source program contains an END DECLARE SECTION, but no corresponding BEGIN DECLARE SECTION was found.

System action: Processing continues.

User response: Check the application program and ensure that a valid BEGIN DECLARE SECTION is present.

Severity: 8 (error)

DSNH292I   E *csectname* **LINE** *nnnn* **COL** *cc* **HOST NAME** *name* **IS TOO LONG; MAXIMUM IS** *size* **CHARACTERS**

Explanation: Fully qualified C host variable names cannot contain more than 256 characters.

System action: If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Reduce the length of names or the level of qualification of the host variable name in error.

Severity: 8 (error)

DSNH303I   W *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT SELECTS INCOMPATIBLE DATA TYPE INTO** *hostvar*

Explanation: The data type of host variable *hostvar* is not compatible with the apparent type of the corresponding select expression. One of them is numeric and the other is character, or one of them is character and the other is graphic. If the select expression is a column reference, the data type is obtained from its declaration.

System action: Processing continues.

System programmer response: Ensure that the table definitions are current, and that the host variable has the proper data type.

User response: Ensure that the table definitions are current and that the host variable has the proper data type.

Severity: 4 (warning)

DSNH310I   **W** *csectname* **LINE** *nnnn* **COL** *cc language* **HOST VARIABLE** *name* **WAS DECLARED IN** *location*

Explanation: Host variable *name* should not be declared in this *location* of the source program. This message is issued only at reference time.

System action: Processing continues.

User response: If this host variable is used, the user assumes responsibility for possible exposure of incorrect output.

Declaring a COBOL host variable in the File Section might lead to incorrect output from SQL statements or abend 04E with RC=00E7000C.

Severity: 4 (warning)

Related concepts:

⯈ Host variables (DB2 Application programming and SQL)

Related tasks:

⯈ Declaring host variables and indicator variables (DB2 Application programming and SQL)

Related information:

⯈ 00E7000C (DB2 Codes)

DSNH312I   **E** *csect-name* **LINE** *line-number* **COL** *column-number* **UNDEFINED OR UNUSABLE HOST VARIABLE** *variable-name*

Explanation: A problem occurred with the specified variable in an SQL statement. Either no declaration was found in the application program before the statement, or the attributes of the variable are improper for the intended use.

csect-name
>   The name of the control section that issued the message.

line-number
>   The line number in the source code of the application program.

column-number
>   The numeric representation of the column. The first column is identified as column 1, the second as column 2, and so on.

variable-name
>   The name of the variable that caused the error.

This message is issued only at reference time.

System action: If the SQL statement that contains the variable is a declaration, it is discarded.

If the SQL statement that contains the variable is executable, it is replaced by host language statements that, if executed, set an error code in the SQLCA.

User response: Check the declaration and the attributes of the named host variable. If you intended the name to refer to a column, the colon (:) must not be present. If you intended a literal, enclose the value in string delimiters (apostrophes or quotation marks).

If you intended to use a host variable, only a subset of the possible variable declarations are valid for use as a DB2 host variable. See the information about rules for host variables in an SQL statement.

To determine the source of the problem, use the cross-reference listing and the source listing to determine whether any of the following situations apply:
- If you use the STDSQL(YES) SQL processing option, make sure that all of the host variable declarations are declared within the BEGIN DECLARE SECTION and the END DECLARE SECTION.
- If the cross-reference listing does not contain the host variable name, the declaration is missing. If the source for a declaration is present, a missing terminator from a comment or from a string might cause this type of error. The margin settings might also cause this type of error.
- If the name is present, but the data type is not listed, the host variable is not usable; it does not follow the rules for valid host variable definitions.

- If several definitions exist for the name, the first reference might occur before the definition. In this situation, the TWOPASS SQL processing option helps by processing DECLARE statements before the other types of statements. However, host variable declarations and cursor declarations are processed in the same pass. The host variables that are referenced in cursor declarations must be defined before the cursor is defined in the application program. This rule applies regardless of whether the ONEPASS or TWOPASS SQL processing option is used.
- A parameter marker is expected rather than a host variable in an EXECUTE IMMEDIATE statement.
- The data type of a host variable that is passed in a CALL statement is not CHAR or VARCHAR.
- The length attribute of a CHAR or VARCHAR host variable that is passed in a CALL statement exceeds 255 characters.
- If you are using the precompiler for COBOL, check whether the named host variable is declared in a structure that contains FILLER elements. The FILLER elements might invalidate the variables in the structure.

Severity: 8 (error)

Related concepts:

➡ Rules for host variables in an SQL statement (DB2 Application programming and SQL)

➡ Output from the DB2 precompiler (DB2 Application programming and SQL)

Related tasks:

➡ Collecting data for DB2 precompiler and DB2 coprocessor problems (Collecting data)

Related reference:

➡ Descriptions of SQL processing options (DB2 Application programming and SQL)

➡ EXECUTE IMMEDIATE (DB2 SQL)

➡ DECLARE CURSOR (DB2 SQL)

➡ CALL (DB2 SQL)

➡ Host structures in COBOL (DB2 Application programming and SQL)

DSNH313I   **E** *csect-name* **LINE** *nnnn* **COL** *cc* **INVALID TYPE OR LENGTH OF HOST VARIABLE '***variable-name***'**

Explanation: The host variable *variable-name* is used in an SQL statement, but either the type or the length is improper for this use.

System action: The corresponding SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Check the declaration and the attributes of the named host variable.

Severity: 8 (error)

Related concepts:

➡ Compatibility of SQL and language data types (DB2 Application programming and SQL)

Related tasks:

➡ Declaring host variables and indicator variables (DB2 Application programming and SQL)

DSNH314I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **REFERENCE TO HOST VARIABLE** *name* **IS AMBIGUOUS**

Explanation: The host variable *name* was defined more than once in this application program. The precompiler cannot distinguish which host variable definition should be used here. The following table indicates some possible combinations. DB2 supports only a single adjacent qualifier for a host variable name.

Case	Ref	Definition	Result	Suggested Change
1	a	x.a	Valid	None
2	x.a	x.a, y.a	Valid	None

Case	Ref	Definition	Result	Suggested Change
3	a	x.a, y.a	DSNH314I	Reference x.a or y.a
4	a	x.a, a	Valid	None
5	x.a	q.x.a, v.x.a	No precompiler error, but host language compiler will issue an error.	Variable name changes
6	a	a, a	No precompiler error, but host language compiler will issue an error.	Variable name changes

System action: If the SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Make the host variable name unique or use qualification to indicate which host variable definition is to be used, as suggested in the above table.

Severity: 8 (error)

DSNH330I   E *csect-name* **LINE** *line-number* **COL** *column-number* **TEXT COULD NOT BE CONVERTED TO CCSID 1208, REASON** *reason-code*

Explanation: DB2 parses SQL statements in Unicode UTF-8 (CCSID 1208). The DB2 precompiler or coprocessor will convert the application program internally (not externally) from the application source EBCDIC CCSID to UTF-8 for processing. This message indicates that text from the application program could not be converted.

csect-name
> The name of the control section that issued the message.

reason-code
> A numeric value that indicates the reason for the failure. A numeric value that indicates the context of this error:

> **012**   A character to be converted is an invalid code-point.

> **016**   The string to be converted does not conform to the rules for valid data. For example, this could be a mismatch between shift-in and shift-out, or by the absence of half of a DBCS character.

System action: The statement is discarded.

Programmer response: Take the appropriate action for the *reason-code* value:

012   Change the text in the application program to a valid code point.

016   Change the text in the application program to conform to the rules for well-formed mixed data.

Severity: 8 (error)

DSNH331I   W *csectname* **LINE** *nnnn* **COL** *cc*. **SUBSTITUTION OCCURRED ON CONVERSION FROM CCSID** *from-ccsid* **TO** *to-ccsid*

Explanation: During conversion of the source text into UTF-8 (CCSID1208), a character was encountered that resulted in a substitution character of either '3F'x (in EBCDIC code pages), or '1A'x (in UTF-8).

System action: Processing of the statement continues.

User response: Review the character that was converted to determine if the substitution character '3F'x is correct. If it is correct, the message can be ignored. Otherwise, change the character to the correct code point.

Severity: 4 (warning)

DSNH332I **U** *csect-name* **CHARACTER CONVERSION FROM CCSID** *ccsid-1* **TO CCSID** *ccsid-2* **IS NOT SUPPORTED, REASON** *reason-code*

Explanation: DB2 parses SQL statements in Unicode UTF-8 (CCSID 1208). The DB2 precompiler or coprocessor will convert the application program internally (not externally) from the application source EBCDIC CCSID to UTF-8 for processing. In some cases, DB2 also converts portions from UTF-8 back to the source EBCDIC CCSID. This message indicates that the conversion could not take place.

csect-name
      The name of the control section that issued the message.

ccsid-1   The original CCSID.

ccsid-2   The target CCSID.

reason-code
      A numeric value that indicates the reason for the failure. A numeric value that indicates the context of this error:

      000   An invalid ccsid was specified on the CCSID option.

      001   No conversion is defined from the original CCSID to the target CCSID

System action: The precompiler or coprocessor is terminated.

System programmer response: Make sure that the Unicode System Services conversion image includes the required conversions.

User response: Check that the value of the CCSID option is valid. Run the precompilation again with a valid CCSID if you do not want to use the EBCDIC system CCSID.

Severity: 16 (unrecoverable error)

Related concepts:

➡ z/OS: Unicode Services User's Guide and Reference

DSNH350I **E** *csect-name* *column-name* **WAS IMPLICITLY OR EXPLICITLY REFERENCED IN A CONTEXT IN WHICH IT CANNOT BE USED**

Explanation: The ALTER, CREATE, or DECLARE TEMPORARY TABLE statement is invalid for one of the following reasons:

- A LOB, DECFLOAT, or XML column cannot be specified as a column of the primary key in a PRIMARY KEY clause.
- A LOB, DECFLOAT, or XML column cannot be specified as a column of a UNIQUE KEY.
- A LOB, DECFLOAT, XML, or security label column cannot be specified as a column of a foreign key in a *referential-constraint* clause.
- A LOB, DECFLOAT, XML, or security label column cannot be specified as a column of a parent key in a REFERENCES clause.
- A LOB, DECFLOAT, or XML column cannot be specified as a column of a partitioning key for range partitioning.
- A LOB or XML column cannot be defined in a temporary table (created global temporary table or declared global temporary table).
- A LOB, DECFLOAT, or XML column cannot be defined in a table that has an EDITPROC.
- A LOB, DECFLOAT, or XML column cannot be specified as a column of an index key, and the result data type, including the intermediate result, of a key expression for an index cannot be a LOB, DECFLOAT, XML, or a distinct type based on a LOB or DECFLOAT data type. **Exception:** For an XML index, only XML columns can be specified, and the result type of the index key can be DECFLOAT.
- A hidden column or security label column cannot be specified in a key expression for an index.
- A ROWID, LOB, or XML data type cannot be the data type of a column in the result table of an *as-result-table* clause of a CREATE TABLE or DECLARE GLOBAL TEMPORARY TABLE statement.
- A security label column *column-name* exists in the result table of the *as-result-table* clause; however, it is not the only security label column in that result table. A table can be defined only with a single security label column.

- A LOB or XML column cannot be defined in a table that has an edit procedure that is defined with row attribute sensitivity.

System action: The statement cannot be processed.

User response: Correct the syntax, and reissue the statement.

Severity: 8 (error)

DSNH372I **E** *csectname* **ONLY ONE ROWID, ROW CHANGE TIMESTAMP, SECURITY LABEL OR IDENTITY COLUMN IS ALLOWED IN A TABLE**

Explanation: An attempt was made to do one of the following actions:
- Create a table with more than one ROWID column.
- Add a ROWID column to a table that already has one.
- Create a table with more than one identity column.
- Add an identity column to a table that already has one.
- Add a security label column to a table that already has one.
- Create a table with more than one security label column.
- Create a table with more than one row change timestamp column.
- Add a row change timestamp column to a table that already has one.
- Create a table without a security label in the Common Criteria environment which requires a security label column on every table created.

System action: The statement cannot be executed.

User response: For a CREATE TABLE statement, select only one column to have the row ID data type, the IDENTITY attribute, the SECURITY LABEL attribute, or the row change timestamp attribute. For an ALTER TABLE statement, a ROWID column, identity column, or security label column already exists for the table. Do not attempt to add another column with the data type row ID, with the AS IDENTITY attribute or the AS SECURITY LABEL to the table. If the Common Criteria environment is set (DSN6SPRM COMCRIT= YES), then every table created must have a security label. Designate a column AS SECURITY LABEL or change the subsystem parameter.

Severity: 8 (error)

DSNH385I **ASSIGNMENT TO AN SQLSTATE OR SQLCODE VARIABLE IN AN SQL PROCEDURE** *routine-name* **MAY BE OVER-WRITTEN AND DOES NOT ACTIVATE ANY HANDLER**

Explanation: SQL routine *routine-name* was created, but it contains an assignment statement that specifies the SQLCODE or SQLSTATE variable as the target. Assignment to an SQLSTATE or SQLCODE variable in an SQL routine only changes the value in the variable, and does not activate any handler. The SQLSTATE and SQLCODE variables are assigned values by processing of SQL statements in the SQL routine. Therefore, any value assigned might be over-written as a result of SQL statement processing.

System action: The routine definition is processed successfully.

Programmer response: No recovery is necessary. If the intent was to signal an error or warning, specify the SIGNAL or RESIGNAL statement instead of the assignment. To prevent this warning, remove any assignments to the SQLSTATE or SQLCODE special variables.

Severity: 4 (warning)

DSNH401I **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT SPECIFIES INCOMPATIBLE OPERANDS FOR COMPARISON OPERATOR** *name*

Explanation: The comparison operation *name* is used with a mixture of incompatible data type operands.

System action: Processing continues.

User response: Make both operands compatible character data types. Ensure that the table declaration is correct.

Severity: 4 (warning)

DSNH402I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT APPLIES** *name* **TO CHARACTER DATA**

Explanation: The numeric operation or function *name* is used with a non-numeric operand.

System action: Processing continues.

User response: Make the operand numeric or use a character function. Ensure that the table declaration is correct.

Severity: 4 (warning)

DSNH405I   **E** *csectname* **THE NUMERIC CONSTANT** *constant* **CANNOT BE USED AS SPECIFIED BECAUSE IT IS OUT OF RANGE**

Explanation: The numeric *constant* is not in the proper range.

The proper ranges for SQL values are as follows:
- 5.4E-79 to 7.2E+75 for FLOAT values
- $-(10^{31} -1)$ to $+(10^{31} -1)$ for DECIMAL values
- -2147483648 to 2147483647 for INTEGER values
- -32768 to +32767 for small integer (SMALLINT) values
- $-9.999999999999999 \times 10^{-383}$ to $9.999999999999999 \times 10^{384}$ for DECFLOAT(16) values
- $-9.999999999999999999999999999999999 \times 10^{-6143}$ to $9.999999999999999999999999999999999 \times 10^{6144}$ for DECFLOAT(34) values
- - 9223372036854775808 to +9223372036854775807 for big integer (BIGINT) values

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the number.

Severity: 8 (error)

DSNH408I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **STATEMENT INSERTS OR UPDATES COLUMN** *column-name* **WITH INCOMPATIBLE DATA TYPE**

Explanation: The data type of the value to be inserted into a column by an INSERT or UPDATE statement is incompatible with the declared data type of the column.

csect-name
> The name of the control section that issued the message.

nnnn
> The line number that is generated for DSNH messages. For more information about these line numbers, see "Message line numbers" on page 155.

cc
> The location of the source column. For more information about the *cc* value, see "Message line numbers" on page 155.

column-name
> The name of the column into which you are trying in insert data.

System action: Processing continues.

Programmer response: Ensure that the table definitions are current. Also, ensure that host variables and literals have the proper data types for the target column.

Severity: 4 (warning)

Related concepts:

Compatibility of SQL and language data types (DB2 Application programming and SQL)

Related tasks:

Declaring table and view definitions (DB2 Application programming and SQL)

DSNH414I   E *csectname* **LINE** *nnnn* **COL** *cc* **THE NUMERIC OR DATETIME COLUMN** *column-name* **IS SPECIFIED IN A LIKE PREDICATE**

Explanation: Column *column-name* with a numeric data type was specified in a LIKE predicate. Columns specified in a LIKE predicate must be character or graphic string columns.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the statement.

Severity: 8 (error)

DSNH434I   **OPTION** *keyword* **IS A DEPRECATED FEATURE**

Explanation: *keyword* is a deprecated feature that will not be supported in releases following DB2 Version 11. It is accepted, but we recommend that you discontinue the use of this keyword.

For indexes, we recommend that you use type 2 indexes rather than type 1 indexes.

System action: Processing continues normally.

User response: No change is required for the current release. However, we recommend that you change your SQL statement to remove this feature to prepare for future releases when this feature is not supported.

Severity: 4 (warning)

DSNH435I   E *csectname* **AN INVALID SQLSTATE** *sqlstate* **IS SPECIFIED IN A RAISE_ERROR FUNCTION, RESIGNAL STATEMENT, OR SIGNAL STATEMENT**

Explanation: The SQLSTATE specified in the RAISE_ERROR function, RESIGNAL statement, or SIGNAL statement of a trigger or a routine does not conform to the rules for a valid SQLSTATE.

System action: The statement cannot be processed.

User response: Correct the SQLSTATE specified in the RAISE_ERROR function or SIGNAL or RESIGNAL statement. The SQLSTATE must be a character string containing exactly 5 characters. It must be of type CHAR defined with a length of 5, or a type VARCHAR defined with a length of 5 or greater. The SQLSTATE value must follow the rules for application-defined SQLSTATEs as follows:

- Each character must be from the set of digits ('0' through '9') or non-accented upper case letters ('A' through 'Z').
- The SQLSTATE class (first two characters) cannot be '00' since this represents successful completion.

Severity: 8 (error)

DSNH449I   W *csectname* **LINE** *nnnn* **COL** *cc* **CREATE FUNCTION OR PROCEDURE FOR** *routine-name* **CONTAINS AN INVALID FORMAT OF THE EXTERNAL NAME CLAUSE OR IS MISSING THE EXTERNAL NAME CLAUSE**

Explanation: An error was found in the EXTERNAL NAME clause of the CREATE FUNCTION or CREATE PROCEDURE statement for *name*, or the clause is needed but was not specified. See SQLCODE -449 for specific information about rules for LANGUAGE JAVA.

System action: The corresponding SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax of the CREATE or ALTER statement for the function or procedure.

Severity: 8 (error)

Related reference:

➡ ALTER FUNCTION (external) (DB2 SQL)

➡ ALTER PROCEDURE (external) (DB2 SQL)

➡ CREATE FUNCTION (external scalar) (DB2 SQL)

➡ CREATE PROCEDURE (SQL - external) (DB2 SQL)

Related information:

➡ -449 (DB2 Codes)

DSNH451I **THE** *data-item* **DEFINITION IN THE CREATE OR ALTER STATEMENT FOR** *routine-name*
CONTAINS DATA TYPE *type* **WHICH IS NOT SUPPORTED FOR THE TYPE AND LANGUAGE**
OF THE ROUTINE

Explanation: An error was found in the *data-item* part of the CREATE or ALTER statement for *routine-name*. The
statement contained an invalid data type, or it contained a distinct type which is based on the invalid data type.

System action: The statement cannot be processed.

Programmer response: Correct the specification of the routine and reissue the CREATE statement.

Severity: 8 (error)

DSNH457I **E** *csectname* **LINE** *nnnn* **COL** *cc*. **A FUNCTION OR DISTINCT TYPE CANNOT BE CALLED** *name*
SINCE IT IS RESERVED FOR SYSTEM USE.

Explanation: The user-defined function or distinct type cannot be created or referenced because the name selected is
reserved for use by the system.

A number of names used as keywords are reserved for system use. These names may not be used as user-defined
functions or distinct-type-names, **even if they are delimited identifiers**. These names are:

```
=        <        >        >=       <=
<>       ¬=       ¬>       ¬>
ALL      AND      ANY      BETWEEN  DISTINCT
EXCEPT   EXISTS   FALSE    FOR      FROM
IN       IS       LIKE     MATCH    NOT
NULL     ONLY     OR       OVERLAPS SIMILAR
SOME     TABLE    TRUE     TYPE     UNIQUE
UNKNOWN
```

System action: The statement cannot be executed.

User response: Select a name for the function or distinct type that is not reserved for system use.

Severity: 8 (error)

DSNH473I **E** *csectname* **LINE** *nnnn* **COL** *cc*. **A USER DEFINED DATA TYPE CANNOT BE CALLED THE SAME**
NAME AS A SYSTEM PREDEFINED TYPE (BUILT-IN TYPE).

Explanation: The name of a data type to be created has an unqualified name that is the same as a
system-predefined data type. This is not allowed. Adding delimiters does not make the name valid. The following
names are restricted:

BIGINT	DECIMAL	REF
BINARY	DECFLOAT	ROWID
BLOB	DOUBLE	SMALLINT
BOOLEAN	DOUBLE PRECISION	TIME
CHAR	FLOAT	TIMESTAMP
CHAR VARYING	GRAPHIC	VARBINARY
CHARACTER	INT	VARCHAR
CHARACTER VARYING	INTEGER	VARGRAPHIC

CLOB	LONG VARCHAR	XML
DATALINK	LONG VARGRAPHIC	
DATE	NUMERIC	
DBCLOB	REAL	
DEC	ROWID	

System action: The statement cannot be executed.

User response: Correct the statement to use another identifier for the name of the new user-defined type.

Severity: 8 (error)

DSNH487I   **E** *csectname* **LINE** *nnnn* **COL** *cc object-type object-name* **ATTEMPTED TO EXECUTE AN SQL STATEMENT WHEN THE DEFINITION OF THE FUNCTION OR PROCEDURE DID NOT SPECIFY THIS ACTION**

Explanation: SQL procedure *object-name* contains SQL statements, but the SQL procedure definition specifies the NO SQL option.

System action: The SQL statement cannot be executed.

Programmer response: Change the definition of the SQL procedure to allow SQL statements, or remove the failing SQL statement from the SQL procedure.

DSNH490I   **E** *csectname* **LINE: MV.NNNN COL** *cc* **NUMBER** *number* **DIRECTLY SPECIFIED IN A SQL STATEMENT IS OUTSIDE THE RANGE OF ALLOWABLE VALUES IN THIS CONTEXT (***minval***,** *maxval***)**

Explanation: A number (*number*) was specified that is not valid in the context in which it was specified. The minimum allowed value in this context is *minval*. The maximum allowed value in this context is *maxval*. *n* must be within the range specified by *minval* and *maxval* (*minval =< n =< maxval*).

System action: The statement was not executed.

User response: Change the value *n* to a valid value in the statement.

Severity: 8 (error)

DSNH491I   **E** *csectname* **LINE** *nnnn* **COL** *cc*. **CREATE STATEMENT FOR USER-DEFINED FUNCTION** *function-name* **MUST HAVE A RETURNS CLAUSE, AND EITHER THE EXTERNAL CLAUSE (WITH OTHER REQUIRED KEYWORDS) OR THE SOURCE CLAUSE.**

Explanation: A required clause is missing in the CREATE for function *function-name*. If EXTERNAL was specified, the following must also be specified:
- LANGUAGE
- PARAMETER STYLE

System action: The statement cannot be executed.

User response: Correct and reissue the failing statement.

Severity: 8 (error)

DSNH504I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **CURSOR** *name* **WAS NOT DECLARED**

Explanation: Identifier *name* is used as a cursor, but was not declared. The declaration might have been discarded because of errors.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct or add the required cursor declaration.

Severity: 8 (error)

DSNH506I   E *csectname* **LINE** *nnnn* **COL** *cc* **DECLARE CURSOR FOR STATEMENT NAME MUST PRECEDE THE PREPARE STATEMENT FOR THAT STATEMENT NAME**

Explanation: This DECLARE *name* CURSOR statement is preceded by a PREPARE *name* statement. In DB2, declarations must precede action statements. This is not the same as in SQL/DS.

System action: The declaration is noted as being in error and discarded.

User response: Move the DECLARE or PREPARE statement so that they are in the proper order, or specify the TWOPASS option if the host language is PLI, ASM, or C. The TWOPASS option has the precompiler check for declarations on the first pass and avoids this problem.

Severity: 8 (error)

DSNH520I   E *csectname* **THE OPEN STATEMENT FOR CURSOR cursor-name IS INVALID BECAUSE THE CURSOR WAS DEFINED BY AN ALLOCATE CURSOR STATEMENT**

Explanation: The cursor was not successfully declared, so it was assumed to be defined by an ALLOCATE CURSOR statement. A cursor defined by an ALLOCATE CURSOR statement is open after successful completion of the statement. Attempting to OPEN an allocated cursor is an error.

System action: The statement cannot be executed.

User response: Add or fix the declaration of the cursor. If the cursor is supposed to be defined by an ALLOCATE CURSOR statement then remove the OPEN statement.

Severity: 8 (error)

DSNH521I   W *csectname* **THE statement FOR CURSOR cursor-name IS ASSUMED TO APPLY TO A DYNAMIC ALLOCATE CURSOR STATEMENT.**

Explanation: A FETCH, CLOSE, or DESCRIBE for an undeclared cursor was issued. The cursor is assumed to be defined using a dynamic ALLOCATE CURSOR statement.

System action: The cursor is assumed to be allocated dynamically.

User response: Ensure that the cursor in the FETCH, CLOSE, or DESCRIBE statement is dynamically allocated.

Severity: 4 (warning).

DSNH522I   E *csectname* **DESCRIBE CURSOR cursor-name IS INVALID BECAUSE THE CURSOR WAS DEFINED BY A DECLARE CURSOR STATEMENT.**

Explanation: DESCRIBE CURSOR is allowed for cursors defined using the ALLOCATE CURSOR statement.

System action: The statement cannot be executed.

User response: Remove the DESCRIBE CURSOR statement. Issue a DESCRIBE of the statement used to define the declared cursor.

Severity: 8 (error).

DSNH527I   **WARNING: THE PRECOMPILER OR DB2 COPROCESSOR ATTEMPTED TO USE THE DB2-SUPPLIED DSNHDECP MODULE.**

Explanation: The DSNHDECP module that is supplied by DB2 is being used to run the DB2 precompiler or coprocessor. The precompiler or coprocessor requires site-dependent settings that are specified in the DSNHDECP module that was created when DB2 was installed. The installation-generated DSNHDECP module typically resides in the SDSNEXIT library.

System action: Processing halts.

System programmer response: The following actions are required to enable the DSNH CLIST to load your user-supplied DSNHDECP module rather than the DB2 DSNHDECP module that was supplied by DB2:

1. The JOBLIB or STEPLIB concatenation of any job or TSO userid that calls DSNH needs to allocate the library where the user-supplied DSNHDECP module resides (usually prefix.SDSNEXIT) ahead of the library where the DB2-supplied DSNHDECP module resides (prefix.SDSNLOAD).

2. The DSNH call should include the PCLOAD parameter, and it should be coded as shown here: PCLOAD('*(DSNHPC)').

User response: Check the STEPLIB, JOBLIB, or system link list concatenation used in the DB2 address spaces. To ensure that the proper DSNHDECP module is used, verify that the library that contains the DSNHDECP module that is generated during installation precedes the SDSNLOAD library that contains the DSNHDECP module that was supplied by DB2.

DSNH557I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **INCONSISTENT GRANT/REVOKE KEYWORD** *keyword*; **PERMITTED KEYWORDS ARE** *keyword-list*

Explanation: The identified *keyword* is not permitted. The *keyword-list* is a list of the keywords that are permitted in this context. The combination of privileges requested on a GRANT or REVOKE statement is not allowed. The authorities must all be of one type and consistent with the form of GRANT or REVOKE. REVOKE UPDATE (column list) is not allowed; only REVOKE UPDATE is permitted.

System action: The containing SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax.

Severity: 8 (error)

Related reference:

➡ GRANT (DB2 SQL)

➡ REVOKE (DB2 SQL)

➡ Statements (DB2 SQL)

DSNH558I   **E** *csect-name* **LINE** *nnnn* **COL** *ccc* **INVALID CLAUSE OR COMBINATION OF CLAUSES ON A GRANT OR REVOKE**

Explanation: "PUBLIC AT ALL LOCATIONS" is invalid for non-table privilege GRANT or REVOKE statements. It is also invalid for table privilege GRANT or REVOKE statements that specify the ALTER or INDEX privilege.

System action: The statement cannot be executed.

User response: Correct the syntax.

Severity: 8 (error)

Related reference:

➡ GRANT (DB2 SQL)

➡ REVOKE (DB2 SQL)

➡ Statements (DB2 SQL)

DSNH562I   **THE SPECIFIED PRIVILEGES CANNOT BE GRANTED TO PUBLIC.**

Explanation: The GRANT statement attempted to grant one of the following privileges to the reserved authorization ID PUBLIC (public):
- System DBADM
- DATAACCESS
- ACCESSCTRL
- CREATE_SECURE_OBJECT

These privileges cannot be granted to PUBLIC (public).

System action: The statement is not processed.

Programmer response: Change the statement to specify a privilege that can be granted to PUBLIC, or specify a grantee other than PUBLIC.

Related reference:

↪ GRANT (DB2 SQL)

DSNH577I **E** *csectname* **LINE** *nnnn* **COL** *cc object-type object-name* **ATTEMPTED TO MODIFY DATA WHEN THE DEFINITION OF THE FUNCTION OR PROCEDURE DID NOT SPECIFY THIS ACTION**

Explanation: An SQL procedure contains SQL statements that modify data, but the SQL procedure is defined with the NO SQL, READS SQL DATA, or CONTAINS SQL option.

System action: The SQL statement cannot be executed.

User response: Change the definition of the SQL procedure to allow SQL statements that modify data, or remove the failing SQL statement from the SQL procedure.

Severity: 8 (error)

DSNH580I **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE RESULT-EXPRESSIONS OF A CASE EXPRESSION CANNOT ALL BE NULL**

Explanation: There is a CASE expression in the statement that has all the *result-expressions* (expressions following the THEN and ELSE keywords) coded with the keyword NULL.

System action: The statement was not processed.

Programmer response: Change the CASE expression to include at least one *result-expression* with a keyword other than NULL.

Severity: 8 (error)

DSNH581I **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE DATA TYPES OF THE RESULT-EXPRESSIONS OF A CASE EXPRESSION ARE NOT COMPATIBLE**

Explanation: A CASE expression in the statement has at least one *result-expression* (expressions following the THEN and ELSE keywords) whose data type does not match that of one of the other result-expressions.

System action: The statement was not processed.

Programmer response: Change the CASE expression so that the data types of all of the *result-expressions* are compatible.

Severity: 8 (error)

DSNH582I **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE SEARCH-CONDITION IN A SEARCHED-WHEN-CLAUSE CANNOT BE A QUANTIFIED PREDICATE, IN PREDICATE, OR AN EXISTS PREDICATE**

Explanation: The *search-condition* in a *searched-when-clause* specifies a quantified predicate, an IN predicate or an EXISTS predicate, but is not allowed.

System action: The statement was not processed.

Programmer response: Correct the *search-condition*.

Severity: 8 (error)

DSNH585I **W** *csectname* **LINE** *nnnn* **COL** *cc* **DUPLICATE** *schema name* **SCHEMA NAME**

Explanation: The identified schema name was specified twice in the statement.

System action: The duplicated schema name is ignore.

User response: Remove/correct the duplicate specification.

Severity: 4 (error)

DSNH586I   **E** *csectname* **LINE** *nnnn* **COL** *cc.* **THE TOTAL LENGTH OF THE CURRENT PATH SPECIAL REGISTER CANNOT EXCEED 254 CHARACTERS.**

Explanation: The CURRENT PATH special register is defined as a VARCHAR(254). The content of the string includes each schema name delimited with double quotes and separated from the next schema name by a comma. The total length of the string of all schema names in the CURRENT PATH cannot exceed 254 characters. A SET CURRENT PATH statement causing this message would exceed this limit.

System action: The statement cannot be executed.

User response: Remove schema names to reduce the total length to fit the 254 character maximum length. If all the schema names are required, it may be necessary to consolidate some user-defined functions so that fewer schema names are required for the CURRENT PATH.

Severity: 8 (error)

DSNH587I   **A list of item-references are not in the same family.**

Explanation: Each item-reference in the SET assignment statement is either a host-variable or a transition-variable. The list of item-references must be of the same family, meaning if one of the item-references is a transition-variable, then all of the item-references in the list must be a transition-variable. If the statement is used in the triggered action of a CREATE TRIGGER statement, each item-reference must identify a transition variable. If the statement is used in any other context, each item-reference must identify a host variable.

System action: The corresponding SQL statement is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the syntax of the SQL statement and precompile the program again. Refer to the DB2 SQL Reference to information on the SET assignment statement.

Programmer response: Correct the statement and execute it again.

Severity: 8 (error)

DSNH590I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **NAME** *name* **IS NOT UNIQUE IN ROUTINE** *routine-name*

Explanation: Name *name* is a parameter, SQL variable, condition name, or label in SQL procedure *routine-name*. *name* is not unique.

System action: The statement cannot be processed.

User response: Change the name to make it unique within the routine.

Severity: 8 (error)

DSNH599I   **W** *csectname* **LINE** *nnnn* **COL** *cc.* **COMPARISON FUNCTIONS ARE NOT CREATED FOR A DISTINCT TYPE BASED ON A LONG STRING DATA TYPE.**

Explanation: Comparison functions are not created for a distinct type based on a long string data type (BLOB, CLOB, DBCLOB, LONG VARCHAR, LONG VARGRAPHIC) since the corresponding function are not available for these built-in data types.

System action: Processing continues.

User response: No action is required.

Severity: 4 (warning)

DSNH612I   **E** *csectname* **LINE** *nnnn* **COL** *cc* *column-name* **IS A DUPLICATE COLUMN NAME**

Explanation: The CREATE INDEX, CREATE TABLE, CREATE VIEW or ALTER TABLE statement specifies the same 'column-name' for two (or more) columns of the index, table, view, or the UPDATE OF clause of a trigger definition specifies the same column name more than once. Column names must be unique within an index, a table, a view, or in the UPDATE OF clause of a trigger definition. A column cannot be specified in more than one ALTER TABLE clause except if it is specified in an ALTER COLUMN clause and ADD CHECK CONSTRAINT clause.

System action: The statement cannot be executed.

DSNH628I

User response: Correct the CREATE statement to specify unique names for each of the columns of the index, table, view, or the columns in the UPDATE OF clause of a trigger definition. Correct the ALTER statement to specify unique names for each of the ALTER COLUMN clauses.

This error can also occur on CREATE TABLE when a column list of a PRIMARY KEY, FOREIGN KEY, or UNIQUE clause contains two or more occurrences of the same column name.

Severity: 8 (error)

DSNH628I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **MUTUALLY EXCLUSIVE CLAUSES HAVE BEEN SPECIFIED.**

Explanation: Mutually exclusive clauses were specified in one or more ways.

- A *column-definition* value contains both NOT NULL and DEFAULT NULL clauses.
- A *column-definition* value contains both FIELDPROC and DEFAULT clauses.
- A *select-statement* value contains both the *update-clause* and the FOR FETCH ONLY clause.

ALTER FUNCTION or CREATE FUNCTION statements
- A CREATE FUNCTION statement contains both a CAST FROM clause and a SOURCE clause.
- A CREATE FUNCTION statement contains both a SOURCE clause and a RETURNS TABLE clause.
- A CREATE FUNCTION statement contains both a SOURCE clause and a clause used to define an external function (EXTERNAL, LANGUAGE, NO SQL).
- A CREATE FUNCTION or ALTER FUNCTION statement contains a clause other than the CAST FROM clause for external scalar functions.

ALTER INDEX or CREATE INDEX statements
- A CREATE INDEX statement contains both GENERATE KEY USING and a column specified as ASC or DESC.
- A CREATE INDEX statement contains both a GENERATE KEY USING clause, and a column specified as ASC or DESC.
- A CREATE INDEX or ALTER INDEX statement attempted to specify either CLUSTER, PARTITION BY, or both, for an XML index or an extended index.
- A CREATE INDEX or ALTER INDEX statement attempted to specify either CLUSTER, PARTITION BY, or both, for an XML index or an extended index.

ALTER PROCEDURE or CREATE PROCEDURE statements
- A CREATE PROCEDURE or ALTER PROCEDURE statement attempts to use the NO WLM ENVIRONMENT and PROGRAM TYPE SUB options. When NO WLM ENVIRONMENT is used, then PROGRAM TYPE MAIN must also be used.
- A CREATE PROCEDURE or ALTER PROCEDURE statement attempts to use both NO WLM ENVIRONMENT and either USER or DEFINER for EXTERNAL SECURITY. When NO WLM ENVIRONMENT is used, then PARAMETER STYLE must be either GENERAL or SIMPLE CALL WITH NULLS.
- A CREATE PROCEDURE or ALTER PROCEDURE statement contains both a LANGUAGE REXX clause, and a PARAMETER STYLE DB2SQL or PARAMETER STYLE JAVA clause.

ALTER TABLE statements
- An ALTER TABLE statement contains both a DROP CONSTRAINT clause and either a DROP FOREIGN KEY clause or a DROP CHECK clause.
- An ALTER TABLE statement contains both an ALTER COLUMN clause and a VALIDPROC clause.
- An ALTER TABLE statement contains both an ALTER COLUMN clause and a clause other than the check constraint clause.
- An ALTER TABLE statement contains a clause and one of the following clauses: ADD CLONE, DROP CLONE, RENAME COLUMN, ADD ORGANIZED BY HASH, ALTER ORGANIZATION, or DROP HASH ORGANIZATION. When one of these clauses is specified, no other clause is allowed on the ALTER TABLE statement.

ALTER TABLESPACE or CREATE TABLESPACE statements

- A CREATE TABLESPACE or ALTER TABLESPACE contains both the LOCKPART YES and LOCKSIZE TABLESPACE
- A CREATE TABLESPACE or ALTER TABLESPACE contains both the LOCKPART YES and LOCKSIZE TABLESPACE
- A CREATE TABLESPACE statement contains both the SEGSIZE and NUMPARTS clauses
- A CREATE TABLESPACE statement contains both the SEGSIZE and LARGE clauses.
- A CREATE TABLESPACE statement contains both the SEGSIZE and MEMBER CLUSTER clauses.

CREATE TRIGGER statements

- A CREATE TRIGGER statement specifies more than one correlation name for OLD, NEW, OLD_TABLE, or NEW_TABLE. Each of these correlation specific can appear no more than once in the CREATE TRIGGER statement.

DECLARE CURSOR statements

- A DECLARE CURSOR statement specifies both the WITH RETURN TO CLIENT option and one the following options: WITH RETURN (TO CALLER) or WITHOUT RETURN.

System action: The statement was not processed.

Programmer response: Change the options specified in the statement and reissue the statement.

Severity: 8 (error)

DSNH630E E *csectname* **LINE** *nnnn* **COL** *cc* **THE WHERE NOT NULL SPECIFICATION IS INVALID FOR TYPE 1 INDEXES**

Explanation: Type 1 indexes cannot be created with the WHERE NOT NULL specification.

System action: The statement cannot be executed.

Programmer response: Either the index must be created as a type 2 index, or the WHERE NOT NULL specification must not be used.

Severity: 8 (error)

DSNH637I E *csectname* **LINE** *nnnn* **COL** *cc* **DUPLICATE** *keyword* **KEYWORD**

Explanation: The identified keyword was specified twice in the statement.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Remove the duplicate specification.

Severity: 8 (error)

DSNH643I **A CHECK CONSTRAINT OR THE VALUE OF AN EXPRESSION FOR A COLUMN OF AN INDEX EXCEEDS THE MAXIMUM ALLOWABLE LENGTH**

Explanation: The definition of a check constraint or expression of an index column exceeds the maximum allowable length for one of the following reasons:

- A check constraint exceeded 3800 UTF-8 bytes. Redundant blank spaces are excluded from this limit.
- A value for an expression or an index exceeded 4000 UTF-8 bytes.

System action: The CREATE TABLE or ALTER TABLE statement failed. The statement cannot be executed.

User response: Change the constraint or expression so that it does not exceed the maximum allowable limit. If i is a check constraint, rewrite the check constraint definition so that it is less than 3800 UTF-8 bytes. You might need to divide the check constraint into two or more smaller check constraints.

Severity: 8 (error)

DSNH644I E *csectname* **INVALID VALUE SPECIFIED FOR KEYWORD** *keyword-name* **IN** *statement-type*
STATEMENT

Explanation: The value specified for the *keyword-name* parameter in the *statement-type* SQL statement is not a permitted value.

System action: If the SQL statement containing the problem is a declaration, it is discarded. If the SQL statement containing the problem is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

User response: Correct the statement.

Severity: 8 (error)

Related reference:

↪ Statements (DB2 SQL)

DSNH683I E *csectname* **LINE** *nnnn* **COL** *cc*. **THE SPECIFICATION FOR COLUMN, DISTINCT TYPE,**
FUNCTION, OR PROCEDURE *data-item* **CONTAINS INCOMPATIBLE CLAUSES**

Explanation: There is an error in the data item specification in a statement. Incompatible specifications are present such as "INTEGER and FOR BIT DATA", or "INTEGER AS LOCATOR". The location of the error is given by *data-item* as follows:

- For a CREATE or ALTER TABLE statement, *data-item* gives the name of the column containing the error. The error could be an invalid specification of FOR BIT DATA, FOR SBCS DATA, FOR MIXED DATA, or FIELDPROC for column data-item.
- For a CREATE FUNCTION or CREATE PROCEDURE statement, *data-item* is a token that identifies the area of the problem in the statement. For example, "PARAMETER 3", "RETURNS", or "CAST FROM".
- For a CREATE DISTINCT TYPE statement, *data-item* gives the name of the type being defined
- Otherwise, *data-item* is a token that identifies the area of the problem in a parameter list for a function. For example, "PARAMETER 5".

System action: The statement cannot be processed.

User response: Correct and reissue the failing statement.

Severity: 8 (error)

DSNH684I E *csectname* **LINE** *nnnn* **COL** *cc* **LENGTH OF LITERAL LIST BEGINNING** *string* **IS TOO LONG**

Explanation: The length of the literal list beginning with *string*, excluding insignificant blanks and delimiting parenthesis, is greater than 255.

System action: The SQL statement cannot be executed.

System programmer response: Correct the SQL statement.

Severity: 8 (error)

Related concepts:

↪ Constants (DB2 SQL)

↪ Language elements (DB2 SQL)

DSNH739I E *csectname* **LINE** *nnnn* **COL** *cc*. **CREATE FUNCTION** *function-name* **FAILED BECAUSE FUNCTIONS**
CANNOT MODIFY DATA WHEN THEY ARE PROCESSED IN PARALLEL.

Explanation: The function cannot be created because ALLOW PARALLEL and MODIFIES SQL DATA were both specified explicitly or implicitly. A function cannot be parallelized if it modifies data.

System action: The statement cannot be executed.

User response: Specify DISALLOW PARALLEL or change the MODIFIES SQL DATA to NO SQL, CONTAINS SQL or READS SQL DATA.

Severity: 8 (error)

DSNH740I   *xxxxxx phase* **FINISHED, RC =** *retcode*, **LISTING IN** *data-set*

Explanation: The DSNH INST phase completed with return code *retcode*. The *phase* is one of the following:
- PLI Macro Pass
- Precompiler
- CICS Command Translator
- Bind Package
- Bind Plan
- Compile/assembly
- PLI
- Prelink (unique to C/370 Compiler)
- Link
- Run

If the CLIST puts the listing into a data set, the name of the data set is shown as *data-set*. If the PRINT parameter is LEAVE, NONE, or TERM, the 'LISTING IN *data-set*' part of this message is omitted.

Refer to message DSNH756I for an explanation of the print data set names assigned during processing.

This message is issued by the following CLIST: DSNH

System action: Processing continues as requested.

User response: This phase is complete. If there are errors, other messages are produced. The next phase requested begins. The order of the phases is:
1. PLI Macro Pass
2. Precompiler
3. CICS Command Translator
4. Bind Package
5. Bind Plan
6. ASM, ASMH, C (C/370 Compiler), COBOL, COBOL2, FORTRAN, PLI
7. Prelink
8. Link
9. Run

Severity: *retcode*

DSNH741I   *process1***(NO) IMPLIES THAT** *process2* **CANNOT BE RUN**

Explanation: An inconsistent processing set was specified. An earlier phase and a later phase were requested, but not the intermediate step needed to process data between them. For example, requesting the precompile and the link edit without the compiler produces this error. Values for *process1* are PRECOMP, CICSXLAT, COMPILE, LINK, or BIND. Values for *process2* are COMPILE, LINK, or RUN.

This message is issued by the following CLIST: DSNH

System action: *process2* is not performed, but other processing is done.

User response: Specify a consistent combination of processing. Application program preparation needs to run all the steps to provide a correct application program. The process does allow beginning or ending steps to be run at a different time. See message DSNH740I for the required step order.

Severity: 4 (warning)

DSNH750I   INVALID DATA SET: *clist-keyword clist-keyword-parameter*

Explanation: The data set name specified on the *clist-keyword-parameter* contains unbalanced parenthesis.

This message is issued by the following CLIST: DSNH

System action: The CLIST is terminated.

User response: If a member name is to be associated with the specified data set, the member name must be enclosed in parentheses.

Severity: 8 (error)

DSNH751I   *clist* ERROR IN *parameter* VALUE *value*

Explanation: The specified CLIST keyword *parameter* does not have an acceptable value.

System action: The CLIST processing is terminated.

User response: Correct the invalid parameter, and resubmit the job.

Severity: 8 (error)

Related tasks:

➡ Compiling and link-editing an application (DB2 Application programming and SQL)

Related reference:

➡ DSNH (TSO CLIST) (DB2 Commands)

DSNH752I   INPUT DATA SET NAME IS MISSING OR NONE

Explanation: The DSNH CLIST requires a value for the input data set name (INPUT). You cannot use an input data set name of NONE.

This message is issued by the following CLIST: DSNH

System action: The CLIST prompts for a name, if prompting is allowed. If prompting is not allowed, for example in batch or when PROFILE NOPROMPT was entered, the CLIST terminates.

User response: Specify an INPUT name. See message DSNH753I.

Severity: 0 (successful)

DSNH753I   ENTER INPUT DATA SET NAME

Explanation: The CLIST is requesting the name of the input data set.

This message is issued by the following CLIST: DSNH

System action: The system waits for user to enter data set name or for the CLIST to be terminated.

User response: Enter the name of the data set. Do not specify the parameter or command, but just enter the name of the data set. The standard TSO prefixing will be done if the data set name is not entered within apostrophes.

Severity: 0 (successful)

DSNH754I   *clist* OUTNAME PARAMETER *name* MUST NOT CONTAIN SPECIAL CHARACTERS

Explanation: The DSNH and DSNHC CLISTs, *clist*, use the OUTNAME parameter to build names for a number of data sets that are passed from phase to phase. These names are prefixed by an authorization ID or TSO prefix value and have a number of suffixes. For this reason, OUTNAME parameter *name* must not contain special characters. It can contain a period or national characters. It cannot have apostrophes to specify a single data set name.

System action: The CLIST is terminated.

User response: Correct the input request. The default for this parameter is TEMP, which should be adequate for most usage.

Severity: 8 (error)

DSNH755I   **OUTNAME PARAMETER** *name* **MUST NOT BEGIN THE SAME AS INPUT** *input*

Explanation: The DSNH CLIST uses the OUTNAME parameter *name* to build a number of data sets that are passed from phase to phase. These data sets could conflict with the INPUT data set (*input*) if the names are the same, causing the INPUT data set to be overwritten. To avoid this problem, the names are not allowed to have the same initial characters.

This message is issued by the following CLIST: DSNH

System action: The CLIST is terminated.

User response: Correct the input request. Specify a different OUTNAME. The default for OUTNAME is TEMP.

Severity: 8 (error)

DSNH756I   *clist* **PRINT PARAMETER** *name* **MUST NOT CONTAIN SPECIAL CHARACTERS**

Explanation: The DSNH and DSNHC CLISTs (*clist*) use the PRINT parameter to build names for data sets used for the print listings from various phases. These names are prefixed by an authorization ID or a TSO prefix value and have a number of suffixes. The PRINT parameter *name* can contain a period or national characters, but it cannot contain special characters. It cannot have apostrophes to specify a single data set name.

The data set names built using the PRINT(print) qualifier are the following:

print.CXLIST
        CICS Command Translator

print.LINKLIST
        Linkage Editor

print.LIST
        Compiler or Assembler

print.PCLIST
        DB2 Precompiler

print.SYSCPRT.LIST
        C/370 Compiler

print.SYSOUT.PRELLIST
        C/370 Prelink Utility

System action: The CLIST terminates.

User response: Correct the *name* specified with the PRINT parameter.

Severity: 8 (error)

DSNH757I   **ERROR IN PLAN NAME LENGTH** *value*

Explanation: The *value* is too long for a plan name. A plan name has a maximum of 8 characters.

This message is issued by the following CLIST: DSNH

System action: The CLIST is terminated.

User response: Specify a plan name with 8 or fewer characters.

Severity: 8 (error)

DSNH758I   **HOST(**host-language**) IS NOT PERMITTED WITH RUN(IMS) OR RUN(CICS)**

Explanation: There is no facility for connecting a FORTRAN program to either IMS or CICS.

This message is issued by the following CLIST: DSNH

System action: The CLIST terminates.

User response: Use the specified language in the TSO or batch environment.

Severity: 8 (error)

DSNH759I   *clist* **TERMINATING,** *phase* **RC =** *retcode*

Explanation: The DSNH or DSNHC *clist* terminated at *phase* because the return code *retcode* was greater than or equal to the RCTERM parameter value.

System action: The CLIST terminates.

User response: Determine the cause for the high return code. If the parameters include TERM(NONE) and direct the PRINT to a data set, you might need to search the print data sets for the listing of the error. Refer to message DSNH756I for an explanation of the print data set names assigned during processing.

Problem determination: The location of the messages describing the prior errors depends upon the values of the TERM and PRINT parameters. The messages can be directed to the terminal, to preallocated data sets, or to data sets dynamically allocated during CLIST processing.

If TERM(NONE) PRINT(NONE) was specified, the messages are not available. Execute the CLIST again and specify a location for the error messages.

To find the documentation, note the immediately preceding DSNH740I message for the location of the listings. If the 'LISTING IN data-set' portion of the message is missing, specify a PRINT data set, SOURCE(YES) and XREF(YES), and execute the CLIST again. Use TERM(TERM) to direct error messages to your terminal. These listings and execution of the DSNH CLIST are required for problem diagnosis.

If the error is in the CLIST or the error location is CLIST CHECKING, use CONTROL(SYMLIST) to produce a detailed trace of the CLIST execution and provide the trace listing.

Check for DSNH751I messages that describe the error.

Severity: *retcode*

DSNH760I   **PRELINK OPTION INVALID WITH HOST LANGUAGE (** *host-language* **)**

Explanation: Invocation of the PRELINK step is not supported by *host-language*. This option is only valid for the following DSNH HOST language compilers: C, CPP, and IBMCOB.

This message is issued by the following CLIST: DSNH

System action: The DSNH PRELINK parameter is ignored and prelink utility processing is bypassed. Clist processing continues normally with a return code of 4.

User response: No action is required.

Severity: 4 (error)

Related tasks:

▶ Compiling and link-editing an application (DB2 Application programming and SQL)

Related reference:

▶ DSNH (TSO CLIST) (DB2 Commands)

DSNH794I   **W** *csectname* **LINE** *nnnn* **COL** *cc*. **THE OBJECT** *object-name* **CANNOT BE CREATED BECAUSE THE FIRST THREE CHARACTERS ARE RESERVED FOR SYSTEM OBJECTS**

Explanation: In general, SYS is a reserved prefix for names. The only exceptions to this rule include the following conditions:

- SYSADM and SYSTOOLS are valid schema names.
- SYSPROC is a valid schema name for stored procedures.
- SYSIBM is a valid schema name for a stored procedure when the current application process has the SYSADM or SYSCTRL privilege.
- SYSTOOLS is a valid schema name when the current application process has the SYSADM or SYSCTRL privilege.

This message is also issued if an attempt is made to grant the CREATEIN, ALTERIN or DROPIN privileges on a schema with the SYS prefix. The same exceptions apply to the grant.

This condition is similar to the condition reported in SQLCODE -20074.

User response: Select a name that does not start with a reserved prefix.

Severity: 4

DSNH867I   **E** *csectname* **LINE** *nnnn* **COL** *cc*. **INVALID SPECIFICATION OF A ROWID COLUMN**

Explanation: For a CREATE TABLE statement, the specification of a ROWID column might be invalid for one of the following reasons:

• A ROWID column cannot be a column of a temporary table.

• A ROWID column cannot be a column of a table with an EDITPROC.

• A ROWID column cannot be a column of a foreign key.

• A ROWID column cannot be a column of a primary key.

• A ROWID column cannot be referenced in the check-condition of a table check constraint.

System action: The statement was not executed.

User response: Correct the syntax.

Severity: 8

DSNH901I   **S** *csectname* **U LINE** *nnnn* **COL** *cc* **INTERNAL PRECOMPILER OR DB2 COPROCESSOR ERROR -**
                REASON CODE *ccc*

Explanation: An internal compiler consistency check failed. The following list contains the possible reason codes:

000   The message formatter was called with an unlisted internal message number.

001   The parser received a token code of 0 from the lexical scan. Lexical scan refers to a scan of words rather than syntax.

002   Error in storage suballocation control blocks.
          • The error might have occurred because a DB2 SQL statement limitation was exceeded. The SQL statement might be referencing, creating, or declaring more columns than the DB2 defined maximum number allowed for an SQL statement. Another possibility is that too many host variables are being referenced in the SQL statement.

003   Error in a storage suballocation request.
          • This error might have occurred because of storage constraints when building the DBRM entry for the SQL statement or because the SQL statement exceeds the DB2 maximum statement length.

004   Error in pointer structures.

005   On BIND or REBIND, a host variable was not found.

006   Unexpected value for parser option.

007   The size of a DBRM entry exceeds the maximum size.

008   Internal conversion error.
          • An error occurred when converting a timestamp version-id.

System action: The precompiler or coprocessor is terminated immediately.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

Problem determination: A copy of the application program source and the printout is useful (specify SOURCE and XREF).

Severity: 16 (unrecoverable error)

DSNH902I   U *csect-name* **SEVERE INTERNAL PRECOMPILER OR DB2 COPROCESSOR ERROR IN CONVERSION FROM CCSID** *ccsid-1* **TO CCSID** *ccsid-2*, **REASON** *reason-code*

Explanation: DB2 parses SQL statements in Unicode UTF-8 (CCSID 1208). The DB2 precompiler or coprocessor will convert the application program internally (not externally) from the application source EBCDIC CCSID to UTF-8 for processing. In some cases, DB2 also converts portions from UTF-8 back to the source EBCDIC CCSID. This message indicates that a severe error has occurred in the internal conversion.

csect-name
> The name of the control section that issued the message.

ccsid-1   The original CCSID.

ccsid-2   The converted CCSID.

reason-code
> A numeric value that indicates the reason for the failure. A numeric value that indicates the context of this error:

> **001**   The internal buffer is too small for the conversion.

> **002**   An invalid code point was found during conversion.

> **003**   The string being converted does not conform to the rules for valid data.

> **004**   A miscellaneous error occurred on conversion.

System action: The precompiler or coprocessor is terminated immediately.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

Problem determination: A copy of the application program source and the printout is useful. Specify SOURCE and XREF.

Severity: 16 (unrecoverable error)

DSNH904I   *csectname* **U STORAGE OVERFLOW**

Explanation: More storage is needed, but could not be obtained during precompiler/parser execution.

System action: The precompiler/parser is terminated immediately.

System programmer response: The storage required for the current statement could not be acquired or exceeds the maximum for a single request. If the region size is being used completely, check whether the region can be expanded. If it cannot be expanded, the SQL statement is too large to be processed.

Severity: 16 (unrecoverable error)

DSNH1760I   E *csectname* **LINE** *nnnn* **COL** *cc*. **CREATE STATEMENT FOR STORED PROCEDURE** *procedure-name* **MUST HAVE A VALID LANGUAGE CLAUSE AND EXTERNAL CLAUSE.**

Explanation: The CREATE statement for procedure *procedure-name* must have both a valid LANGUAGE clause and a valid EXTERNAL clause.

System action: The statement cannot be executed.

User response: Correct and reissue the failing statement.

Severity: 8 (error)

DSNH4001I   **HOST LANGUAGE STATEMENT EXCEEDS MAXIMUM OF 1000 LINES**

Explanation: A PL/I program contains a declaration that covers more than 1000 source lines and contains a LOB variable declaration.

System action: Processing continues.

User response: Split the DCL statement into two parts.

Severity: 8

DSNH4408I **E** *csectname* **LINE** *nnnn* **COL** *cc* **THE VALUE IS NOT COMPATIBLE WITH THE DATA TYPE OF ITS TARGET**

Explanation: The data type of the value to be assigned to the parameter or SQL variable is incompatible with the declared data type of the assignment target. Both must be:
* Numerals
* Characters
* Graphics
* Dates or character
* Times or character
* Timestamps or character
* Row ID's
* The same distinct types

System action: The statement cannot be processed.

User response: Examine the current definition for the parameter or SQL variable and ensure that it has the proper data type.

Severity: 8 (error)

DSNH4700I **E** *csect-name* **LINE** *line-number* **COL** *column-number* **ATTEMPT TO USE NEW FUNCTION BEFORE NEW FUNCTION MODE**

Explanation: An attempt was made to use syntax for functions that are newer than the version that is specified by the SQL processing option NEWFUN.

csect-name
> The name of the control section that issued the message.

line-number **and** *column-number*
> The location of the error in the source program.

When NEWFUN(NO) is specified, you cannot use syntax for functions that are introduced by the current version of DB2. When NEWFUN(V9) is specified, you cannot use syntax for any functions that were introduced after Version 9. When the value of NEWFUN(V8) is specified, you cannot use syntax for any functions that were introduced after Version 8.

System action: If the containing SQL statement is a declaration, it is discarded. If the containing SQL statement is executable, it is replaced by host language statements that will, if executed, set an error code in the SQLCA.

Programmer response: Check the value of the SQL processing option setting for NEWFUN. If the NEWFUN option is not specified, then the installation default value is used.

Do one of these things:
* Restrict your program to functions that earlier releases of DB2 introduced. These functions are allowed regardless of the value of NEWFUN.
* Process your program with a value of YES for the NEWFUN option. However, you cannot BIND the DBRM until New Function Mode has been enabled.

Severity: 8 (error)

Related concepts:

➡ Implications of falling back (DB2 Installation and Migration)

➡ Application defaults parameters (DB2 Installation and Migration)

Related tasks:

➡ Migrating from enabling-new-function mode* to new-function mode (DB2 Installation and Migration)

Related reference:

➡ Descriptions of SQL processing options (DB2 Application programming and SQL)

DSNH4701I   S *csectname* **OPTION:** *option* **OPTION NOT PERMITTED WITH NEWFUN NO**

Explanation: You cannot use the indicated SQL processing option when the SQL processing option NEWFUN has a value of NO.

System action: The precompiler is terminated.

Programmer response: To prevent this error in the future, do one of the following:

• Restrict your precompilation to options that were introduced in earlier releases of DB2. These options are valid regardless of the value of NEWFUN.

• Process your program with a value of YES for the NEWFUN option.

Severity: 12 (severe error)

DSNH4702I   *column* **IS NOT VALID IN THE CONTEXT WHERE IT IS USED**

Explanation: The specified column is not a column of the table or view that was specified as the object of the UPDATE or INSERT operation.

System action: The statement cannot be executed. No data was updated.

Programmer response: Verify that the column and table names are specified correctly in the SQL statement. If a correlation name is specified for a table or view, verify that subsequent table references of the table or view in qualified column names use the correlation name and not the table or view name.

SQLSTATE: 42703

DSNH4760I   THE DB2 SQL COPROCESSOR IS USING THE LEVEL *nnn* **INTERFACE UNDER DB2 VERSION** *DB2-version*

Explanation: This message specifies the level of the DB2 SQL coprocessor that is in use. This message is for IBM software support.

System action: No action required.

User response: No action required.

Severity: 0 (informational)

DSNH4761I   THE DB2 SQL COPROCESSOR HAS CHANGED THE NEWFUN OPTION VALUE FROM YES TO NO

Explanation: The DB2 SQL coprocessor option NEWFUN(YES) was specified, but the level of the DB2 SQL coprocessor interface is not compatible with NEWFUN(YES). Therefore, the DB2 SQL coprocessor is running with NEWFUN(NO). Functions that are specific to the new version of DB2 are not allowed.

System action: No action required.

User response: Upgrade to a later compiler version that supports the new DB2 functions.

Severity: 4 (warning)

DSNH4762I   **W** *csectname* **LINE** *nnnn* **COL** *cc* **SQL OPTIONS REQUIRE NEWER COMPILER**

Explanation: The DB2 SQL coprocessor was used during compilation of a DB2 application. One or more of the following SQL options were specified:

• STDSQL(YES)

• TWOPASS

The level of the compiler that was used does not support those SQL options.

System action: Processing continues.

User response: If the SQL options are required, upgrade to a newer level of the compiler.

Refer to message DSNH4760I in the output listing and the DB2 Program Directory for information on selecting the correct level of the compiler to use.

If the SQL options are not required, remove them.

Severity: 4 (warning)

DSNH4763I   **E** *csect-name* **LINE** *nnnn* **COL** *cc* **SQL TYPE IS CLAUSE WAS INCORRECTLY SPECIFIED**

Explanation: The SQL TYPE IS clause was incorrectly specified in a DB2 application that was prepared with the DB2 SQL coprocessor.

System action: The statement cannot be processed.

User response: Correct the statement syntax.

Severity: 8 (error)

Related concepts:

Differences between the DB2 precompiler and the DB2 coprocessor (DB2 Application programming and SQL)

Compatibility of SQL and language data types (DB2 Application programming and SQL)

Related tasks:

Processing SQL statements by using the DB2 coprocessor (DB2 Application programming and SQL)

Related reference:

Equivalent SQL and Fortran data types (DB2 Application programming and SQL)

DSNH4770I   **E** *csect-name* **SQLFLAG PRECOMPILER OPTION IS NO LONGER SUPPORTED**

Explanation: Federal Information Processing Standards (FIPS) and SAA flagging support is disabled. You can no longer specify the precompiler option SQLFLAG.

System action: Precompiler ends with a return code of 8.

User response: Remove the SQLFLAG precompiler option.

Severity: 8 (error)

DSNH4771I   **E** *csectname* **PRECOMPILER OPTION QUOTE CANNOT BE SPECIFIED IN THIS CONTEXT. USE THE APOST OPTION**

Explanation: When the value of the CCSID precompiler option is 1026 or 1155, and the value of the HOST precompiler option is IBMCOB, the APOST precompiler option must be used because the precompiler modified source defines the quotation mark with the hex point of 'FC'X, which is not taken by the COBOL compiler.

System action: Processing continues.

User response: Specify the APOST precompiler options, and precompile the application again. There is no need to modify the COBOL source.

Severity: 8 (error)

DSNH4772I   **I** *csect-name* **LINE** *line-number* **COL** *column-number* **UNQUALIFIED IDENTIFIER** *identifier-name* **IS USED IN AN AMBIGUOUS CONTEXT**

Explanation: The identifier that is indicated by *identifier-name* is not a unique reference in the SQL statement context.

csect-name
> The name of the control section that issued the message.

line-number **and** *column-number*
> The location of the error in the source program.

identifier-name
> The non-unique identifier. An *sql-parameter* or *sql-variable* with the same name is declared, but it does not have priority over a column with this name in this context.

System action: Processing continues.

User response: Inspect the ambiguous reference, and qualify the named identifier to clarify the intended usage in the SQL statement.

• For a reference to an SQL parameter, qualify the name with the SQL PL routine name.

• For a reference to an SQL variable, qualify the name with the label of the compound statement that declares the variable. If no label is present in the compound statement, add one.

• For a reference to a column, qualify the name with the appropriate table designator.

Severity: 0

DSNH4773I   **W** *csect-name* **LINE** *line-number* **COL** *column-number* **HANDLER BODY IS A CONTROL STATEMENT OTHER THAN A COMPOUND STATEMENT**

Explanation: The body of a handler declaration is an SQL control statement other than an SQL compound statement.

csect-name
> The name of the control section that issued the message.

line-number **and** *column-number*
> The location of the error in the source program.

This usage causes the original condition information available to the condition handler logic to be displaced in the following manner:

• The SQL diagnostic area accessed by a GET CURRENT DIAGNOSTICS statement will be modified and unrelated to the original condition.

• The values of variables SQLSTATE and SQLCODE will be modified and unrelated to the original condition.

System action: Processing continues.

User response: Replace the SQL control statement (IF, CASE, LOOP, REPEAT, WHILE) that defines the body of the handler declaration with an SQL compound statement (BEGIN).

Severity: 4 (warning)

DSNH4775I   **E** *csect-name* **LINE** *line-number* **COL** *column number* **STATEMENT SPECIFIED IN SQL ROUTINE IS NOT ALLOWED WITHIN A COMPOUND STATEMENT**

Explanation: A statement specified in the routine body is not allowed in a compound statement.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

An ATOMIC compound statement cannot be specified:

• In an SQL scalar function.

• In an SQL procedure.

- Within another compound statement that is defined as ATOMIC.

System action: The statement cannot be processed.

User response: Remove the statement from the SQL routine.

Severity: 8 (error)

DSNH4777I **E** *csectname* **LINE** *nnnn* **COL** *cc* **NESTED COMPOUND STATEMENTS NOT ALLOWED**

Explanation: An SQL procedure contains nested compound statements, which are not allowed.

System action: The statement cannot be executed.

User response: Rewrite the SQL procedure body so that it does not contain nested compound statements.

Severity: 8 (error)

DSNH4778I **E** *csect-name* **LINE** *line-number* **COL** *column number* **END LABEL** *label-name* **NOT SAME AS BEGIN LABEL**

Explanation: An SQL routine statement contains an ending label and a beginning label that do not match.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

label-name
> The name given for the ending label.

System action: The statement cannot be processed.

User response: Change the ending label in the statement to match the beginning label.

Severity: 8 (error)

DSNH4779I *csectname* **LINE** *nnnn* **COL** *cc* **LABEL** *label* **SPECIFIED** *statement* **IS NOT VALID**

Explanation: In an SQL procedure, *label* is specified on a GOTO or LEAVE statement. *label* is not defined or is not a valid label for the statement.

The label on a LEAVE statement must be the label for a LOOP, REPEAT, WHILE, or compound statement.

The label in a GOTO statement must be defined within a scope that the GOTO statement can reach.
- If the GOTO statement is defined in a compound statement, *label* must be defined inside the same compound statement.
- If the GOTO statement is defined in a handler, *label* must be defined in the same handler and must follow the other scope rules.
- If the GOTO statement is defined outside a handler, *label* must not be defined within a handler.

System action: The statement cannot be processed.

User response: Change the label in the LEAVE statement to match the label on the LOOP, REPEAT, WHILE, or COMPOUND statement that contains the LEAVE statement.

Ensure that the label in the GOTO statement is defined within a scope that the GOTO statement can reach.

Severity: 8 (error)

DSNH4780I **E** *csect-name* **LINE** *line-number* **COL** *column number* **UNDO SPECIFIED FOR A HANDLER**

Explanation: In an SQL routine, UNDO is specified for a handler, which is not supported.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

System action: The statement cannot be processed.

User response: Change the UNDO handler to a CONTINUE or EXIT handler.

Severity: 8 (error)

DSNH4781I   **E** *csect-name* **LINE** *line-number* **COL** *column number* **CONDITION** *condition-name* **SPECIFIED IN HANDLER NOT DEFINED**

Explanation: A handler is declared for a condition in an SQL routine, but the routine does not contain a condition declaration statement that defines the condition.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

condition-name
> The name of the condition that is not defined.

System action: The statement cannot be processed.

User response: Include a condition declaration statement in the SQL routine that relates the condition name to an SQLSTATE value.

Severity: 8 (error)

DSNH4782I   **E** *csect-name* **LINE** *line-number* **COL** *column number* **A CONDITION OR SQLSTATE** *condition-value* **SPECIFIED IS NOT VALID**

Explanation: An SQL routine specified a condition value that is not valid.

condition-value
> Either a specific condition value, which is provided as an SQLSTATE value, or a general condition value, such as SQLEXCEPTION, SQLWARNING, or NOT FOUND.

The condition or SQLSTATE value is not valid for one of the following reasons:

- The handler declaration or condition declaration specifies an SQLSTATE value that is not valid.
- The condition or SQLSTATE value has already been specified by another handler in the same scope.
- The condition or SQLSTATE was specified in the same handler as SQLEXCEPTION, SQLWARNING, or NOT FOUND.

System action: The statement cannot be processed.

User response: Correct the condition or SQLSTATE value.

Severity: 8 (error)

DSNH4783I   **E** *csectname* **LINE** *nnnn* **COL** *cc* **SELECT LIST FOR CURSOR** *cursor-name* **IN FOR STATEMENT NOT VALID**

Explanation: An SQL procedure contains a FOR statement in which the select list in the cursor declaration has a column that is not valid. That column is a duplicate of another column in the select list, or the column is not named.

System action: The statement cannot be executed.

User response: Ensure that all columns in the select list for a cursor declaration in a FOR statement are unique and are named. If a column of the select list is the result of a column function, specify an AS clause to name the column.

Severity: 8 (error)

DSNH4784I   E *csect-name* **DSNHDECP IS FOR DB2 VERSION** "*dsnhdecp-version*" **BUT** *product-name* **IS FOR DB2 VERSION** "*dsnhdecp-version*"

Explanation: The release information from the given DSNHDECP module or user-supplied application defaults module does not match the current level of the DB2 precompiler or SQL coprocessor.

System action: The precompiler or SQL coprocessor is terminated.

Programmer response: Ensure that a *dsnhdecp* data-only load module for the correct DB2 version is available to the precompiler or SQL coprocessor, and that *dsnhdecp* load module is the one that DB2 finds first. DB2 searches for the*dsnhdecp* module in the following places, and in the following order:

- The STEPLIB concatenation for the precompile or compiler step for SQL coprocessor
- The JOBLIB concatenation for the job that includes the precompile step or compiler step for SQL coprocessor
- The system link list

Severity: 12 (severe error)

DSNH4785I   E *csect-name* **LINE** *line-number* **COL** *column number* **THE DECLARATION OR USE OF THE SQLSTATE OR SQLCODE VARIABLE IS NOT VALID**

Explanation: The use of SQLCODE or SQLSTATE in an SQL routine body is invalid for one of the following reasons:

- SQLCODE or SQLSTATE was specified as a parameter name.
- SQLCODE is not declared as INTEGER in a return-codes-declaration clause.
- SQLSTATE is not declared as CHAR(5) in a return-codes-declaration clause.
- SQLCODE or SQLSTATE is assigned the value NULL.

csect-name
>   The name of the control section that issued the message.

line-number **and** *column number*
>   The location of the error in the source program.

System action: The statement cannot be processed.

User response: Declare SQLSTATE only as CHAR(5) and SQLCODE only as INTEGER. Do not assign the null value to SQLCODE or SQLSTATE. Do not use SQLCODE or SQLSTATE as parameter names.

Severity: 8 (error)

DSNH4787I   E *csectname* **LINE** *nnnn* **COL** *cc* **RESIGNAL STATEMENT NOT WITHIN A HANDLER**

Explanation: The RESIGNAL statement can only be used inside condition handlers.

System action: The statement cannot be processed.

User response: Remove the RESIGNAL statement or use a SIGNAL statement instead.

Severity: 8 (error)

DSNH4788I   *csectname* **CCSID** *ccsid* **FOR PROGRAM SOURCE CODE IS NOT SUPPORTED**

Explanation: The DB2 precompiler or DB2 coprocessor does not accept input source code with the specified CCSID. The rules for the CCSID are:

- When source code for an external SQL procedure is precompiled, the CCSID must be a Latin-1 EBCDIC CCSID.
- For other uses of the DB2 precompiler or DB2 coprocessor, the CCSID must be an EBCDIC CCSID.

System action: The precompiler or DB2 coprocessor terminates.

User response: Specify an EBCDIC CCSID.

Severity: 16 (unrecoverable error)

DSNH4789I *csectname* **SQL PROCESSING OPTION "***option***" IS IGNORED**

Explanation: When using the DB2 precompiler or the DB2 coprocessor, the SQL processing option *option* is ignored.

System action: Processing continues.

User response: Make sure the correct CCSID is provided to the precompiler or coprocessor.

If the CCSID provided is a single-byte CCSID, then x'0E' and x'0F' characters are not interpreted as shift-out and shift-in characters surrounding DBCS characters. If the CCSID provided is a mixed CCSID, then x'0E' and x'0F' characters are interpreted as shift-out and shift-in characters surrounding DBCS characters.

Related concepts:

➡ Options for SQL statement processing (DB2 Application programming and SQL)

DSNH4920I **E** *csectname* **TOKEN ID ARRAY TOO SMALL**

Explanation: The DB2 coprocessor needs more token identifier array entries than the coprocessor allocated.

System action: Control is returned to caller with no other action taken.

User response: Allocate a larger token identifier array and make the call to SQLGCMPL again.

Severity: 8 (error)

DSNH5011I **HOST VARIABLE ARRAY** *host-variable-array* **IS EITHER NOT DEFINED OR IS NOT USABLE**

Explanation: Host-variable-array *host-variable-array* was specified in a multiple-row FETCH or multiple-row INSERT statement. The host-variable-array is not defined or is not usable for one of the following reasons:
- The host variable is not a valid host-variable-array.
- The host variable is not a dimensioned array.
- The host-variable-array has more than 1 dimension.
- The host-variable-array defines a structure that does not conform to the rules for defining a host-variable-array.
- The dimension of the host-variable-array used for indicators is not equal to the dimension of the main host-variable-array.

System action: The statement cannot be processed.

Programmer response: Correct any of the following and precompile the program again. Ensure that:
- the dimension of the indicator variable array is equal to the dimension of the main host-variable-array.
- the host-variable-array is a single-dimensional array.
- the host-variable-array or host indicator variable array do not contain structures that are not permissible. The only structures that may be defined in a host-variable-array are those that are used to define variable length string host variables.

Related concepts:

➡ Host variable arrays (DB2 Application programming and SQL)

Related tasks:

➡ Declaring host variables and indicator variables (DB2 Application programming and SQL)

DSNH5033I **DECLARE VARIABLE NOT VALID FOR** *variable*

Explanation: Host variable *variable* specified in a DECLARE VARIABLE statement is not valid. The DECLARE VARIABLE statement occurs after any SQL statement (in a COBOL or FORTRAN application program) or after a DECLARE CURSOR statement (in an application program in any other host language), where the SQL statement and the DECLARE VARIABLE refer to the same variable.

System action: The statement cannot be processed.

User response: Ensure that the DECLARE VARIABLE occurs before the use of the host variable in an SQL statement in the application.

Severity: 8 (error)

DSNH5034I   DECLARE VARIABLE NOT VALID FOR *variable*

Explanation: Host variable *variable* specified in a DECLARE VARIABLE statement is not valid for one of the following reasons:

- A reference is made to a variable that does not exist.
- A reference is made to a variable that is a structure.
- A reference is made to a variable that is a component of two more structures.
- The host variable is not defined as character, varying-length character, graphic, varying-length graphic, CLOB, or DBCLOB. In this case, CLOB AS LOCATOR or DBCLOB AS LOCATOR is not considered a CLOB or DBCLOB.
- FOR BIT DATA, CCSID 65535, FOR MIXED DATA, FOR SBCS DATA is specified and the variable is defined as graphic, varying-length graphic, or DBCLOB.
- FOR BIT DATA, or CCSID 65535 is specified and the variable is defined as CLOB.

System action: The statement cannot be precompiled.

User response: Correct the definition of the host variable. Precompile the program again.

Severity: 8 (error)

DSNH5035I   HOST VARIABLE *variable* **IS SPECIFIED TWICE IN A DECLARE VARIABLE STATEMENT**

Explanation: The host variable *variable* was specified in two or more DECLARE VARIABLE statements, or was specified two or more times in one DECLARE VARIABLE statement. A host variable can be specified once at most in a DECLARE VARIABLE statement.

System action: The statement cannot be processed.

User response: Specify the host variable at most once in a DECLARE VARIABLE statement.

Severity: 8 (error)

DSNH5036I   CCSID VALUE OUT OF RANGE

Explanation: The CCSID for a DECLARE VARIABLE or CAST statement is outside the range of 1 to 65533, or the the value 65535. The value must be within the range of 1 to 65533, or the value must be 65535.

System action: The statement cannot be processed.

User response: If this message is issued for a "DECLARE VARIABLE CCSID integer"or "CAST CCSID integer" statement, change the value to be within the range of 1 to 65533 or the value 65535. If this message is issued for a DECLARE VARIABLE statement with the "FOR SBCS DATA" or "FOR MIXED DATA" clause, then the CCSIDs specified on panel DSNTIPF at installation time need to be updated with valid values.

Severity: 8 (error)

DSNH5037I   DECLARE VARIABLE STATEMENT FOUND WITH ONE-PASS PRECOMPILATION

Explanation: A DECLARE VARIABLE statement was found, but the source program is being precompiled with only one pass. If a DECLARE VARIABLE is used in an assembler source program, the ONEPASS precompiler option must not be used. If a DECLARE VARIABLE is used in a C, C++, or PL/I source program, the TWOPASS precompiler option must be used.

System action: The source program cannot be precompiled.

User response: Specify the TWOPASS precompiler option or, for assembler, do not specify the ONEPASS precompiler option.

Severity: 8 (error)

DSNH20008I UNSUPPORTED OPTION *keyword* **SPECIFIED**

Explanation: *keyword* is a deprecated feature that was supported in releases prior to DB2 Version 11, and is no longer supported.

For indexes, only one type is currently supported: type 2.

System action: The statement cannot be executed.

User response: Correct the syntax of the SQL statement to remove reference to the unsupported keyword.

SQLSTATE: 560A9

Severity: 8 (error)

Related reference:

➥ Statements (DB2 SQL)

DSNH20060I E *csect-name* **LINE** *line-number* **COL** *column-identifier* **UNSUPPORTED DATA TYPE** *data-type* **ENCOUNTERED IN SQL** *object-type object-name*

Explanation: An unsupported data type was specified in the definition of an object.

csect-name
　　　　The name of the control section that issued the message.

line-number **and** *column-identifier*
　　　　The location of the definition.

data-type
　　　　The data type that was specified. This data type is not supported for SQL procedure parameters or variables.

　　　　For native SQL procedures, you cannot use the ROWID data type.

　　　　For external SQL procedures, you cannot use the following data types:
　　　　- BINARY
　　　　- VARBINARY
　　　　- BIGINT
　　　　- DECIMAL FLOAT
　　　　- XML
　　　　- ROWID
　　　　- TIMESTAMP with precision
　　　　- TIMESTAMP WITH TIME ZONE
　　　　- Distinct type

object-type
　　　　The type of object.

object-type
　　　　The type of object.

object-name
　　　　The name of the object.

System action: The statement cannot be processed.

User response: Change the syntax to specify a supported data type.

Severity: 8 (error)

DSNH20061I E *csectname* **LINE** *nnnn* **COL** *cc* **UNEXPECTED ERROR RETURNED FROM LANGUAGE ENVIRONMENT: REASON CODE** *reason-code*, **RETURN CODE** *return-code module-name*

Explanation: An Language Environment® error occurred while the DB2 precompiler was processing an SQL procedure. The reason codes and associated return codes are:

Reason code
　　　　Meaning and associated return code

1　　　　No PIPI token. *return-code* is the return code from the CEEPIPI(init_sub) call.

2　　　　CEE could not be loaded. *return-code* is the return code from the CEEPIPI(add_entry) call.

3　　　　PIPI would not terminate. *return-code* is the return code from the CEEPIPI(term) call.

4   Call to *module-name* failed. *return-code* is the return code from the CEEPIPI(call_sub) call.

5   Bad response from PIPI. *return-code* is the return code from *module-name*.

See the explanation of return codes for the appropriate CEEPIPI call in *z/OS Language Environment Programming Guide* for explanations of the Language Environment return codes.

System action: The statement cannot be executed.

User response: Correct the condition that is described by *reason-code*.

Severity: 8 (error)

DSNH20111I E *csect-name* **LINE** *line-number* **COL** *column number* **CANNOT ISSUE SAVEPOINT, RELEASE SAVEPOINT, ROLLBACK TO SAVEPOINT FROM A TRIGGER, FROM A USER-DEFINED FUNCTION, OR FROM A GLOBAL TRANSACTION**

Explanation: An invalid statement was issued.

csect-name
        The name of the control section that issued the message.

line-number **and** *column number*
        The location of the error in the source program.

The following statements cannot be used in the body of a trigger, global transaction, or user-defined function:
- SAVEPOINT
- RELEASE SAVEPOINT
- ROLLBACK TO SAVEPOINT

System action: The statement is not processed.

Programmer response: Correct the logic of the application program so that this error does not occur.

Severity: 8 (error)

Related reference:

➦ SAVEPOINT (DB2 SQL)

➦ RELEASE SAVEPOINT (DB2 SQL)

➦ ROLLBACK (DB2 SQL)

DSNH20120I E *csect-name* **LINE** *line-number* **COL** *column number* **AN SQL TABLE FUNCTION MUST RETURN A TABLE RESULT**

Explanation: An SQL function that specifies TABLE in the RETURNS clause must return a result that is a table.

csect-name
        The name of the control section that issued the message.

line-number **and** *column number*
        The location of the error in the source program.

System action: The statement can not be processed.

User response: Either remove the TABLE keyword from the RETURNS clause to make it a scalar function or specify a fullselect in the RETURN statement of the TABLE function body.

Severity: 8 (error)

DSNH20129I E *csectname* **LINE** *nnnn* **COL** *cc* **SPECIAL REGISTER IS NOT VALID AS USED**

Explanation: An attempt was made to reference a special register, but the syntax was invalid for the special register.

The CURRENT PACKAGE PATH special register, or a local special register such as CURRENT SERVER, CURRENT PACKAGESET, or CURRENT APPLICATION ENCODING SCHEME cannot be referenced in a SET

assignment-statement statement or VALUES INTO statement where:

- The special register is part of an expression in the source value.
- Multiple source values are being assigned to multiple targets.
- The source value in an assignment is the CURRENT PACKAGE PATH special register or a local special register, and the target is an array variable.

If the target is not an array variable, the only valid uses of these special registers in a SET *assignment-statement* statement or VALUES INTO statement are:

- SET *variable*= CURRENT APPLICATION ENCODING SCHEME;
- SET *variable*= CURRENT SERVER;
- SET *variable*= CURRENT PACKAGESET;
- SET *variable*= CURRENT PACKAGE PATH;
- VALUES CURRENT APPLICATION ENCODING SCHEME INTO *variable*;
- VALUES CURRENT SERVER INTO *variable*;
- VALUES CURRENT PACKAGESET INTO *variable*;
- VALUES CURRENT PACKAGE PATH INTO *variable*;

System action: The statement cannot be processed.

Programmer response: Correct the statement in one of the following ways:

- Do not use the special register in an expression as the source value of a SET *assignment-statement* or VALUES INTO statement.
- Do not assign multiple values to multiple special registers in a SET *assignment-statement* or VALUES INTO statement.
- Do not assign a special register to a target that is an array variable.

Severity: 8 (error)

DSNH20148I E *csect-name* **LINE** *line-number* **COL** *column number* **A RETURN STATEMENT DOES NOT EXIST OR WAS NOT INVOKED DURING THE EXECUTION OF ROUTINE** *routine-name* **WITH SPECIFIC NAME** *specific-name*

Explanation: The SQL function must contain at least one RETURN statement and a RETURN statement must be executed when the function is invoked.

csect-name
> The name of the control section that issued the message.

line-number **and** *column number*
> The location of the error in the source program.

routine-name
> The name of the routine.

specific-name
> The specific function name, which is unique within the schema.

System action: The statement can not be processed.

User response: Ensure that a RETURN statement is executed when the function is invoked.

Severity: 8 (error)

DSNH20228I A STACKED DIAGNOSTICS AREA IS NOT AVAILABLE

Explanation: A stacked diagnostics area is only accessible from within a handler in a native SQL routine.

This error can occur when the STACKED keyword is specified in a GET DIAGNOSTICS statement outside of an SQL procedure.

System action: The statement was processed.

Programmer response: Modify the application logic so that the STACKED diagnostic is only accessed from within a

handler. If the routine is an external SQL procedure, or a host language program then the STACKED keyword cannot be used.

DSNH20289I INVALID STRING UNIT *unit* **SPECIFIED FOR FUNCTION** *function-name*

Explanation: The statement invoked a built-in function using a string length unit that is not valid with the type of data that is provided to the function. This can occur for the following reasons:

- Binary data was specified with *unit*. CODEUNITS16, or CODEUNITS32 are not valid with binary string data.
- Graphic data was specified with OCTETS as the string length unit. OCTETS is not valid with graphic data.

System action: The statement cannot be processed.

System programmer response: Change the invocation of the function to remove the invalid string length unit specification or change it to a valid string length unit for the data type being processed.

DSNH20367I W *csectname* **LINE** *nnnn* **COL** *cc* **OPTION** *clause* **IS NOT SUPPORTED IN THE CONTEXT IN WHICH IT WAS SPECIFIED**

Explanation: The *clause* in the statement is not supported in the context in which has been specified. The *clause* is ignored.

System action: Processing continues normally.

User response: No change is required for the current release; however, it is recommended that the statement be modified to conform to valid SQL syntax.

DSNH20428I E *csect-name* **LINE** *line-number* **COL** *column-number*, **URI SPECIFIED IN THE XMLSCHEMA CLAUSE IS AN EMPTY STRING**

Explanation: An empty string was found in the XMLSCHEMA clause.

csect-name
    The name of the control section that issued the message.

line-number **and** *column-number*
    The location of the error in the source program.

System action: The SQL statement that contains the empty string is discarded.

User response: Specify the URI in the correct format.

Severity: 8 (error)

DSNH20436I E *csect-name* **LINE** *line-number* **COL** *column number* **THE DATA TYPE SPECIFIED FOR AN ARRAY TYPE IS NOT VALID.**

Explanation: The data type that is specified for an array in one of the following contexts is not valid:
- CREATE TYPE statement
- Array constructor
- Argument to UNNEST
- Argument to ARRAY_AGG
- Target of ARRAY_AGG

The following restrictions apply to the data type for an array:

- These data types are not supported:
    - ROWID
    - XML
    - A user-defined data type other than an array data type
- The data type for an array index of an associative array must be INTEGER or VARCHAR, with a length attribute no greater than 32704.

System action: The statement cannot be processed.

Programmer response: Use a supported data type.

DSNH20441I

| **Related reference**:

| ➡ Array constructor (DB2 SQL)

| ➡ collection-derived-table (DB2 SQL)

| ➡ CREATE TYPE (DB2 SQL)

| **DSNH20441I E** *csect-name* **LINE** *line-number type-name* **TYPE IS NOT VALID WHERE SPECIFIED. REASON CODE**
| *reason-code*.

| **Explanation:** An array data type cannot be specified in the context in which it is used.

| *type-name*
| The name of the array type.

| *reason-code*
| A numeric value that indicates the reason that the statement cannot be executed. Possible values are:

| **1** An array data type was specified where an array type is not supported, or a value that is not an
| array data type was specified where an array value is required.

| Some of the contexts in which an array data type is supported are:
| • A parameter to an SQL function
| • A return data type from an SQL function
| • A Parameter to an SQL procedure
| • An SQL variable that is declared in an SQL function
| • An SQL variable that is declared in an SQL procedure

| Some of the contexts in which an array data type is not supported are:
| • A parameter or variable for an external routine
| • A return type of an inline SQL function
| • A column in a table

| The UNNEST operator must be an array value. That array value must be one of the following
| items:
| • An SQL variable
| • An SQL parameter
| • An array function invocation
| • A CAST specification of a parameter marker to an array type

| **2** A global variable was defined as a distinct type. A global variable cannot be defined as a distinct
| type.

| **System action:** The statement cannot be processed.

| **Programmer response:** See the documentation for the SQL statement on which the error occurred for a complete list
| of restrictions on the use of the array data type in that context. Change the SQL statement to use an array data type
| only in a context in which it is allowed.

| **Severity:** 8 (error)

| **Related reference**:

| Array constructor (DB2 SQL)

| ARRAY_DELETE (DB2 SQL)

| TRIM_ARRAY (DB2 SQL)

| EXISTS predicate (DB2 SQL)

| ALTER TABLE (DB2 SQL)

| CREATE FUNCTION (external scalar) (DB2 SQL)

| CREATE FUNCTION (external table) (DB2 SQL)

| CREATE PROCEDURE (external) (DB2 SQL)

| CREATE TABLE (DB2 SQL)

| DECLARE GLOBAL TEMPORARY TABLE (DB2 SQL)

DSNH20474I PERMISSION OR MASK CANNOT BE CREATED FOR *object-name* **OF TYPE** *object-type*. **REASON CODE** *reason-code*.

Explanation: The CREATE PERMISSION or CREATE MASK statement failed

object-name
> The name of the object.

object-type
> The type of object.

reason-code
> A numeric value that indicates the reason for the failure.
>
> **11** A SELECT clause contains either `*` or `name.*`.
>
> **14** An ordinary SQL identifier contains a dash (`-`).

System action: The statement cannot be processed.

Programmer response: Correct the syntax, and reissue the statement.

Related reference:

 CREATE MASK (DB2 SQL)

 CREATE PERMISSION (DB2 SQL)

| **DSNH20494I A PUBLIC ALIAS NAME,** *name*, **CAN ONLY BE QUALIFIED WITH SYSPUBLIC AND NOT THE SCHEMA NAME** *schema-name*.

| **Explanation:** A qualified name was specified for a public alias but the qualifier was not SYSPUBLIC.

| **System action:** The statement cannot be processed.

| **User response:** Change the two-part name to a one-part name or specify the qualifier SYSPUBLIC. If a public alias
| was not intended and PUBLIC was specified in the statement, remove the keyword PUBLIC.

| **DSNH20524I THE STATEMENT FAILED BECAUSE OF AN INVALID PERIOD SPECIFICATION OR PERIOD CLAUSE FOR PERIOD** *period-name*. **REASON CODE** *reason-code*.

| **Explanation:** A period specification or period clause is invalid.

| *period-name*
| > The name of the period that is invalid. Possible values are BUSINESS_TIME or SYSTEM_TIME.

| *reason-code*
| > A numeric value that indicates why the period is invalid. *reason-code* can have the following value:

DSNH20524I

| | **3** | *period-name* violated the following requirement: each expression must return a value of a built-in data type and can contain any of the following supported operands: |

- A constant
- A special register
- A variable, which can be either a host variable, an SQL variable, an SQL parameter, a transition variable, or a global variable
- A parameter marker
- A CAST specification where the cast operand is a supported operand
- An expression that uses arithmetic operators and operands
- A scalar function whose arguments are supported operands

These rules have the following exceptions:

- A period specification or period clause for a view must not contain an untyped parameter marker.
- The source expression of SET CURRENT TEMPORAL BUSINESS_TIME and SET CURRENT TEMPORAL SYSTEM_TIME statements must not contain a parameter marker or a transition variable.

System action: The statement cannot be processed.

User response: Correct the syntax and resubmit the statement.

Severity: 8 (error)

Related concepts:

➡ Data types (DB2 SQL)

➡ References to variables (DB2 SQL)

Related reference:

➡ Special registers (DB2 SQL)

➡ CAST specification (DB2 SQL)

➡ Scalar functions (DB2 SQL)

Chapter 7. DSNI messages

Message identifiers that begin with DSNI communicate information about the data manager.

DSNI001I   **RESTART HAS BEEN DEFERRED REASON reason TYPE type NAME name**

Explanation: Restart could not be performed for one or more pages of the resource identified by *type* and *name*. Those pages are placed in the logical page list (LPL) and restart for those pages is deferred. The reason code explains why restart was not performed.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

System action: DB2 restart continues. Pages for which restart activity was not performed are put into the LPL.

User response: Correct the problem indicated by the reason code. Then use the START DATABASE command or the RECOVER utility to remove the pages from the LPL.

DSNI002I   *csect-name* **UNABLE TO START OBJECT TYPE** *object-type* **OBJECT NAME** *object-name* **REASON** *reason* **TYPE** *type* **NAME** *name*

Explanation: The resource identified could not be started because it was unavailable. The resource could be a database, a table space, an index space, or a partition within a table space or an index space. The 'object-type' identifies the type. The 'object-name' identifies the name. If the DB2 catalog is unavailable, 'object-type' contains binary zeros and 'object-name' contains blanks.

The resource that was unavailable is identified by 'type' and 'name'. The 'type' code identifies the type of the resource as well as the format and content of the 'name'.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes. The 'reason' contains the reason code explaining why the resource was unavailable.

System action: The specified database, space, or partition is not started and remains in its current state. If there are other databases, spaces, or partitions to be started, they are processed normally.

User response: Correct the resource unavailable problem and reissue the -START command.

DSNI003I   *csect-name* **UNABLE TO STOP OBJECT TYPE** *object-type* **OBJECT NAME** *object-name* **REASON** *reason* **TYPE** *type* **NAME** *name*

Explanation: The identified object could not be stopped because a resource required to perform the stop was unavailable. The object could be a database, a table space, an index space, or a partition within a table space or an index space.

The 'object-type' identifies the type and 'object-name' identifies the name. If the DB2 catalog is unavailable, 'object-type' contains binary zeros and 'object-name' contains blanks.

The resource that was unavailable is identified by 'type' and 'name'. The 'type' code identifies the type of the resource as well as the format and content of the resource, 'name'.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes. The 'reason' contains the reason code explaining why the resource was unavailable.

System action: The specified database, space, or partition is not stopped and remains in the current state. If there are other databases, spaces, or partitions to be stopped, they are processed normally.

User response: Correct the resource unavailable problem and reissue the -STOP command.

DSNI004I RESOURCE HAS BEEN STOPPED TYPE type NAME name

Explanation: The resource identified by TYPE and NAME could not be opened/closed, initialized, or was found to be in an inconsistent state. The resource has been 'stopped' by DB2. If an open error occurred during DB2 startup, restart processing for the resource was deferred.

TYPE and NAME identify the resource that has been stopped. The TYPE code identifies the type of the resource as well as the format and content of the resource, NAME.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

System action: The database, table space or index space is stopped by DB2.

User response: Previous messages provide information to allow you to determine the cause of the problem and the method of correcting it. Refer to them, and correct the problem. Then restart the database, table space, or index space using the -START command; and invoke the application again.

DSNI005I *csect-name* **LPL OR GRECP RECOVERY CANNOT BE DONE FOR** *function-name* **- ID =** *function-id*
OBJECT TYPE *object-type* **OBJECT NAME** *object-name* **REASON** *reason-code* **TYPE** *resource-type* **NAME**
resource-name

Explanation: Resource recovery failed because of problems during group buffer pool recovery pending (GRECP) or logical page list (LPL) recovery.

csect-name
> The name of the control section that issued the message.

function-name
> The command or function that failed. Expected values are:
> * START DATABASE COMMAND
> * AUTOMATIC GRECP RECOVERY
> * AUTOMATIC LPL RECOVERY

function-id
> A decimal number that represents the recovery function.

object-name
> The name of the object.

object-type
> The type of object, which could be a table space, an index space, or a partition within a table space or index space.

reason-code
> A numeric value that indicates the reason for the failure.

resource-type **and** *resource-name*
> The resource that could not be recovered. The *resource-type* value identifies the type of the resource and the format of the resource name.

This message, in combination with message DSNI006I or DSNI022I, provides the LPL or GRECP recovery information for a recovery function that is identified by the *function-id*.

System action: The specified object is started but remains in a LPL or GRECP state. If there are other objects to be recovered, they are processed normally. Other status messages might be issued to the console, including:
* DSNI006I, which indicates the start of recovery
* DSNI022I, which provides the status of recovery operations
* DSNI051I, which indicates that second pass log apply has started

User response: Correct the problem that is specified by the reason code, and reissue the START DATABASE command.

Related information:

"DSNI006I"

"DSNI022I" on page 220

"DSNI051I" on page 230

Appendix C, "Resource types," on page 1153

DSNI006I   *csect-name* **ASYNCHRONOUS GRECP OR LPL RECOVERY IN PROGRESS FOR** *function-name* **- ID** *function-id* **TYPE** *object-type* **NAME** *object-name*

Explanation: Recovery has started for an object with group buffer pool recovery pending (GRECP) or logical page list (LPL) status.

csect-name
> The name of the control section that issued the message.

function-name
> The command or function that triggered the recovery. Expected values are:
> - START DATABASE COMMAND
> - AUTOMATIC GRECP RECOVERY
> - AUTOMATIC LPL RECOVERY

function-id
> A decimal number that represents the recovery function.

object-name
> The name of the object.

object-type
> The type of object, which could be a table space, an index space, or a partition within a table space or index space.

System action: The object is started and GRECP or LPL recovery is initiated. During LPL or GRECP recovery process, additional messages might be issued, including.
- DSNI005I, which indicates an error
- DSNI021I, which indicates successful completion
- DSNI051I, which indicates that second pass log apply has started

User response: No action is required.

You can issue a DISPLAY DATABASE command with the RESTRICT option for the specified table space or index space to determine whether the GRECP or LPL exception state has been cleared.

Related reference:

↪ -DISPLAY DATABASE (DB2) (DB2 Commands)

Related information:

"DSNI005I" on page 214

"DSNI021I" on page 220

"DSNI051I" on page 230

DSNI007I   *csect-name* **UNABLE TO START/RECOVER BECAUSE UNABLE TO DIRECT READ A LOG RECORD LOGRBA** *logrba*

Explanation: Unable to START (or RECOVER) because the necessary log records could not be accessed. Either an error occurred while attempting to allocate or open the log data set or an expected log record was not found on the log.

If the message is in response to a START command, issue the -DISPLAY command to determine which resource(s) was not started. If the message is in response to a RECOVER utility, the resource is the table space or index space that is the object of the RECOVER.

LOGRBA gives the log RBA of the log record that is unavailable.

System action: The resource is not started (or recovered) and remains in the current state. If this is a START command, the resources are processed normally if there are other databases or spaces to be started and they do not involve reading the log.

System programmer response: Run the print log map (DSNJU004) utility. This may be run with DB2 up or down. Find the data set that contains the requested RBA in the output of DSNJU004. If the required RBA is not in a listed data set, one of the following may have occurred:

- The data set was deleted from the bootstrap data set by a run of the change log inventory (DSNJU003) utility. In this case, rerun the utility to insert the missing data set name. DB2 must be down to run this utility.

- The bootstrap data set has been damaged. In this case, the BSDS must be restored from the dump that occurred with the most recent archive log data set. DSNJU003 must be run to supply changes that occurred during and after the last offload operation. Changes may be determined from the operator's console listing.

If a listed log data set contains the required RBA, this message will be preceded by one or more error messages from the log manager (DSNJ). See these messages for appropriate action.

User response: Resubmit the utility, or reissue the START command, when the problem with the log has been resolved.

Problem determination: Issue the -DISPLAY command to verify what got started.

DSNI008I   **DATABASE VALIDATION PROCEDURE name COULD NOT BE LOADED**

Explanation: The database validation procedure identified by 'name' could not be loaded.

User response: Correct the name, and rerun the application.

DSNI009I   *csect-name* **DATABASE EDIT PROCEDURE** *name* **COULD NOT BE LOADED**

Explanation: The database edit procedure identified by 'name' could not be loaded.

User response: Correct the name, and rerun the application.

DSNI010I   **BROKEN PAGE ACCESSED TYPE type NAME name MODNAME csect-name CONN-ID id CORR-ID id LUW-ID id**

Explanation: The page identified by TYPE and NAME could not be accessed, because it is marked as logically broken.

The TYPE and NAME identify the resource that is currently unavailable. The TYPE code identifies the type of the resource as well as the format and content of the resource NAME.

Type codes '00000301', '00000302', and '00000303' apply to this message. Refer to Appendix C, "Resource types," on page 1153 for a table showing all types and names.

'csect-name' has encountered the logically broken page. CONN-ID, CORR-ID, and LUW-ID identify the thread involved in the problem. If the LUW-ID is an '*', the thread originated at this site. Correlation IDs beginning with '0nn', where 'nn' is a number from 01 to 28, identify system agents.

User response: If the TYPE is '00000300', use NAME to determine if the page is in an index space or a table space. Use the REPAIR utility to fix the broken page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNI011I   **PAGE TEMPORARILY UNAVAILABLE TYPE type NAME name MODNAME modname ERQUAL erqual**

Explanation: The page identified by TYPE and NAME has been temporarily marked as logically broken to prevent access until it can be recovered.

The TYPE and NAME identify the resource that is currently unavailable. The TYPE code identifies the type of the resource as well as the format and content of the resource NAME. Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

The module named modname has called a page checking routine, which discovered an error in the page.

The ERQUAL gives the trace id for the error in the page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

System action: Automatic recovery for the page has been scheduled. The current agent will abend with reason code '00C90105', and a dump will be taken. The page will be unavailable to this and other agents until the online recovery has completed successfully. Message DSNI010I will be issued, and a 'resource unavailable' condition will be returned to any agents that attempt to access the page. If the online recovery fails, the page will be marked as broken until it is repaired or recovered (see message DSNI012I).

User response: Save the dump. Watch for message DSNB215I (indicating successful recovery of the page), or DSNI012I (indicating failure of recovery of the page).

DSNI012I   **PAGE LOGICALLY BROKEN TYPE type NAME name MODNAME modname ERQUAL erqual**

Explanation: The page identified by TYPE and NAME is logically incorrect. Additional attempts to access the page will be denied.

The TYPE and NAME identify the resource that is currently unavailable. The TYPE code identifies the type of the resource as well as the format and content of the resource NAME. Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

The module named 'modname' has called a page checking routine, which discovered an error in the page.

The ERQUAL gives the trace id for the error in the page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. An ERQUAL of X'0000' indicates that an abend occurred during online recovery or recover utility. The page has been marked as broken, and the recovery has failed.

System action: The page is marked as logically broken, an 04E abend with reason code '00C90102' is issued, and a dump is taken. The page will be unavailable to this and other agents until it is repaired.

Operator response: Notify the system programmer.

System programmer response: See the Explanation section of this message.

User response: Save the dump for analysis by IBM Support Center. Copy the table space or index space to another data set for later analysis by IBM Support Center. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNI013I   *csect-name* **POTENTIALLY INCONSISTENT DATA REASON** *reason* **ERQUAL** *erqual* **TYPE** *type* **NAME** *name* **CONN-ID** *id* **CORR-ID** *id* **LUW-ID** *id*

Explanation: This message identifies a page with potentially inconsistent data. This message is issued for each page involved in the current data manager operation. The 'csect-name' in the message identifies the CSECT that issued the message.

The 'erqual' in the message identifies the specific abend in the CSECT that caused this message to be issued. This information and the reason code are used by IBM Support Center to identify the cause of the problem.

'name' and 'type' identify the page and its type. The TYPE code identifies the type, as well as the format and content, of the NAME. Refer to Appendix C, "Resource types," on page 1153 for a table showing all types and names.

CONN-ID, CORR-ID, and LUW-ID together identify the thread involved in the problem. If the LUW-ID is an '*', the thread originated at this site. Correlation IDs beginning with '0nn' where 'nn' is a number from 01 to 28, identify system agents.

For further information, see the Explanation section for the abend reason code specified in this message.

System action: The job is abended shortly after this message appears.

Operator response: Collect the console output from the system on which the job was run. Save this output for use in problem determination.

System programmer response: See the Problem Determination section for the specific abend reason code.

User response: Notify the system programmer.

Problem determination: See the System Programmer Response section of the abend reason code specified in this message.

DSNI014I   *csect-name* **DATA IN USE DURING ABEND REASON 00C90101 ERQUAL** *error-qualifier* **TYPE** *page-type* **NAME** *page-name* **CONN-ID** *connection-id* **CORR-ID** *correlation-id* **LUW-ID** *luw-id*

Explanation: This message identifies a page that was in use at the time an abend condition occurred with reason code 00C90101. A separate instance of this message is issued for each page involved in the current operation.

csect-name
> The name of the control section that issued the message.

error-qualifier
> The abend in the control section that caused this message to be issued.

page-type
> A resource type code that identifies the type, format, and content of the page.

page-name
> The name of the page.

connection-id
> The connection identifier of the agent that is involved in the problem.

correlation-id
> The correlation identifier of the agent that is involved in the problem. Correlation IDs that begin with 001 through 028, inclusive, identify system agents.

luw-id   The logical unit of work identifier of the agent that is involved in the problem. If the thread originated at this site, the value of *luw-id* is a *.

System action: The job abends with reason code 00C90101 after this message appears.

Operator response: Collect the console output from the system on which the job was run for the period of time spanning the failure. Save this output for use in problem determination.

System programmer response: See abend reason code 00C90101 for more information.

User response: Notify the system programmer.

Problem determination: In general, the data in these pages is valid. However, if problems are reported that repeatedly involve the same pages or if the same pages are reported by other abends, analyze the set of pages involved.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Related information:

⮕ 00C90101 (DB2 Codes)

Appendix C, "Resource types," on page 1153

DSNI015I   **OBJECT name IS ALREADY STARTED FOR RW ACCESS. THE OBJECT WILL REMAIN STARTED FOR RW ACCESS**

Explanation: The -START DATABASE command with the ACCESS(FORCE) keyword was entered, but the object specified was already started for read/write access.

DSNI016I   **RESOURCE TYPE type NAME name MUST FIRST BE STOPPED BEFORE IT CAN BE STARTED IN A DIFFERENT ACCESS MODE**

Explanation: When issuing the -START DATABASE command on the ROSHARE OWNER system, the object to be started must first be stopped before changing access modes. For example, if a table space in a database defined as ROSHARE OWNER is started for RW access, before it can be started for RO access it must be stopped via the -STOP DATABASE command. It can then be started for RO access via the -START DATABASE command.

System action: The -START DATABASE command is not allowed to proceed.

User response: Issue a -STOP DATABASE command to stop the object before attempting to start it.

Severity: 8 (error)

DSNI017I **RESOURCE TYPE type NAME name IS INCONSISTENT DUE TO ONE OR MORE PENDING STATES**

Explanation: There are two possible reasons for this message:

1. The header page of a table space, index, or partition belonging to a database defined as ROSHARE OWNER could not be marked as consistent during the processing of the STOP DATABASE command because one or more of the following was true:

   - The table space, index, or partition is in recovery pending status.
   - The table space, index, or partition is in check pending status.
   - The table space, index, or partition has pages in the logical page list (LPL).
   - The table space, index, or partition status has a utility in progress (UTUT status on the output from a DISPLAY DATABASE command).
   - The table space, index, or partition has a utility in progress that allows read-only access (UTRO on the output from a a DISPLAY DATABASE command).
   - An I/O error range exists.

2. A retained lock is held on the table space, index, or partition. The retained lock could be signifying a pending status.

If the STOP DATABASE command was issued without specifying the SPACENAM parameter, then one or more of the conditions stated above is true for one or more table spaces, indexes, or partitions in the specified database.

System action: For case 1, if the STOP DATABASE command was specified with the SPACENAM parameter, the table space, index, or partition is stopped. DB2 does not allow access to an inconsistent table space, index, or partition by any ROSHARE READ system.

For case 2, no access to an inconsistent table space, index, or partition is allowed until the retained lock is released by the failed system at completion of its restart.

User response: Issue a DISPLAY DATABASE command to display the status of the object in question. Resolve the pending states and then reissue the STOP DATABASE command to ensure that the consistency indicator is reset.

Severity: 4 (warning)

DSNI018I **UNABLE TO GET LOCK FOR RESOURCE TYPE type NAME name. WILL RETRY TO GET LOCK.**

Explanation: The resource identified by 'type' and 'name' could not be locked because of lock contention. The lock request is made again for this resource.

The 'type' and 'name' identify the resource trying to get a lock. The 'type' code identifies the type of resource as well as the format and content of the resource 'name'.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

System action: The lock request is made again for the specified resource.

Severity: 0 (informational)

DSNI019I **THE STOP COMMAND FOR OBJECT name WAS OVERRIDDEN BY A START FORCE. THE STOP COMMAND IS IGNORED.**

Explanation: The object identified by 'name' was not stopped because a START FORCE command on this object was subsequently issued and processed.

This message is informational only.

System action: The object is not stopped.

Severity: 0 (informational)

DSNI020I   *csect-name* **UNABLE TO DRAIN RESOURCE TYPE** *type* **NAME** *name*. **WILL RETRY DRAINING THE RESOURCE.**

Explanation: The resource identified by 'type' and 'name' could not be drained. The 'type' code identifies the type of resource as well as the format and content of the resource 'name'. The drain request is made again for this resource.

Refer to Appendix C, "Resource types," on page 1153 for an explanation of resource type codes.

System action: The drain request is made again for the specified resource.

DSNI021I   *csect-name* **GRECP OR LPL RECOVERY FOR** *function-name* **- ID =** *function-id* **TYPE** *object-type* **NAME** *object-name* **HAS COMPLETED SUCCESSFULLY.**

Explanation: Recovery has completed successfully.

csect-name
>   The name of the control section that issued the message.

function-name
>   The function that triggered the recovery: Expected values are:
>   - START DATABASE COMMAND
>   - AUTOMATIC GRECP RECOVERY
>   - AUTOMATIC LPL RECOVERY

function-id
>   A decimal number that represents the recovery function.

object-type
>   The type of object, which could be a table space, an index space, or a partition within a table space or index space.

object-name
>   The name of the object.

This message is always preceded by message DSNI006I, which indicates the beginning of the LPL or GRECP recovery for the object. In addition, this message might be preceded by message DSNI051I, which indicates the beginning of second pass LPL or GRECP recovery for the object.

System action: The object is started normally. The group buffer pool recovery pending (GRECP) or logical page list (LPL) exception state is cleared.

User response: No action is required.

Related information:

"DSNI006I" on page 215

"DSNI051I" on page 230

DSNI022I   *csect-name* **LPL OR GRECP RECOVERY LOG APPLY FOR** *function-name* **- ID =** *function-id* **AT LOGPOINT** *current-position* **TO LOGPOINT** *end-position*

Explanation: This message is issued periodically to show the progress of group buffer pool recovery pending (GRECP) or logical page list (LPL) recovery.

csect-name
>   The name of the control section that issued the message.

function-name
>   The function that triggered the recovery: Expected values are:
>   - START DATABASE COMMAND
>   - AUTOMATIC GRECP RECOVERY
>   - AUTOMATIC LPL RECOVERY

function-id
>   A decimal number that represents the recovery function.

| *current-position*
| The log point of the log record that is being processed when message DSNI022I is issued. This value is an
| RBA in a non-data sharing environment or an LRSN in a data sharing environment.

| *end-position*
| The log point of the last log record that will be applied in this recovery. This value is an RBA in a non-data
| sharing environment or an LRSN in a data sharing environment.

| **System action:** Processing continues.

| During the LPL or GRECP recovery process, additional messages might be issued, including:
| • DSNI005I, which indicates an error
| • DSNI006I, which indicates the start of recovery
| • DSNI021I, which indicates successful completion
| • DSNI051I, which indicates that second pass log apply has started

| **User response:** No action is required.
| **Related information**:
| "DSNI005I" on page 214
| "DSNI006I" on page 215
| "DSNI021I" on page 220
| "DSNI051I" on page 230

DSNI023I   *csect-name* **PAGE SET** *dbnam.psnam* **PART** *part* **IS** *status* **ON BEHALF OF UR** *urid*. **BACKOUT TO**
             logpoint **IS REQUIRED.**

Explanation: The value of *status* is either RESTP (Restart Pending) or AREST (Advisory Restart Pending). This
message indicates that the named UR has backout work pending against the named page set or partition. The
backout work for this UR against this page set or partition will be complete after backward processing of the log
reaches the identified logpoint. The value for *logpoint* will be 'RBA *rba*' for non-datasharing and 'LRSN *lrsn*' for
datasharing environments.

For non-partitioned page sets, the partition given in the message is the string *n/a*.

DSNI024I   *csect-name* **BACKOUT PROCESSING IS COMPLETE FOR PAGE SET** *dbnam.psnam* **PART** *part*.

Explanation: This message indicates that backout processing is complete for the named table space, index space, or
partition. If the object was in restrictive restart pending state, that state is lifted. If the object was in an advisory
restart pending state, locks held on the object on behalf of postponed abort URs by the DB2 member issuing the
message are released. The advisory restart pending state is lifted if the DB2 member issuing the message is the only
member still contributing to the state.

For non-partitioned page sets, the partition given in the message is the string "n/a".

DSNI025I   *csect-name* **THE FAST LOG APPLY PROCESS REQUIRES** *aaaaaaaa* **BYTES OF STORAGE, BUT GOT**
             ONLY *bbbbbbbb* **BYTES**

Explanation: The Fast Log Apply process could not obtain all the required storage. However, the storage acquired is
enough for the Fast Log Apply process to continue.

System action: The Fast Log Apply process continues with the smaller amount of storage for the log buffers.

Severity: 0 (informational)

DSNI026I   *csect-name* **THE FAST LOG APPLY PROCESS IS DISABLED BECAUSE ALL OF THE ALLOWABLE**
             STORAGE IS EXHAUSTED

Explanation: There is no ssnmDBM1 storage available for the Fast Log Apply process to use. There are too many
jobs (RECOVER or -START DB command) running in parallel.

System action: The recovery of the requested objects continues without the Fast Log Apply process.

Severity: 0 (informational)

DSNI027I   *csect-name* **THE FAST LOG APPLY PROCESS IS DISABLED BECAUSE IT IS UNABLE TO ACQUIRE THE MINIMUM REQUIRED STORAGE**

Explanation: There is not enough ssnmDBM1 storage available or reserved for the Fast Log Apply process to use.

System action: The recovery of the requested objects continues without the Fast Log Apply process.

Severity: 0 (informational)

DSNI028I   *csect-name* **THE NUMBER OF QUALIFIED LOG RECORDS READ DURING THE FAST LOG APPLY PROCESS IS** *log-records* **AND THE NUMBER OF FAST LOG APPLY BUFFERS PROCESSED ARE** *fla-buffers*

Explanation: This informational message is issued during the Fast Log Apply process. *log-records* indicates the number of log records that have been applied during the Fast Log Apply process. *fla-buffers* indicates the number of buffers used by the Fast Log Apply process.

Severity: 0 (informational)

DSNI029I   *csect-name* **THE FAST LOG APPLY PROCESS IS ACTIVE AND THE AVAILABLE** *ssnm***DBM1 STORAGE IS** *bytes* **BYTES**

Explanation: This informational message is issued during the Fast Log Apply process for member *ssnm*. There is sufficient storage for the Fast Log Apply process to continue, and the remaining amount of available storage is displayed in hexadecimal as *bytes*.

Severity: 0 (informational)

DSNI030I   *csect-name* **BACKOUT PROCESSING IS CONTINUING BEYOND THE REQUESTED LIMIT DUE TO CATALOG OR DIRECTORY ACTIVITY OF UR** *urid***.**

Explanation: The number of log records requested to be processed during restart's backward phase have been processed. Under other circumstances further backout processing would be postponed. However, because the indicated UR has catalog or directory updates that are not backed out yet (and because backout against catalog or directory page sets cannot be postponed), backout processing is continuing. It will continue until this UR's backout is complete.

System action: Backout processing continues.

DSNI031I   *csect-name* **- LOCK ESCALATION HAS OCCURRED FOR RESOURCE NAME =** *resource-name*, **LOCK STATE =** *lock-state*, **PLAN NAME : PACKAGE NAME =** *plan-name***:***package-name*, **COLLECTION-ID =** *collection-id*, **STATEMENT NUMBER =** *statement-number*, **CORRELATION-ID =** *correlation-id*, **CONNECTION-ID =** *connection-id*, **LUW-ID =** *luw-id*, **THREAD-INFO =** *thread-information*

Explanation: Lock escalation has occurred for the specified object.

csect-name
> The name of the control section that issued the message.

resource-name
> The name of the escalated resource. The format of *resource-name* depends on the type of table space:

> **Simple or partitioned table spaces**
> > *database-name.tablespace-name*

> **Segmented table space**
> > Either *creator-name.table-name* or *DBID.OBID* of the table

lock-state
> The lock state of the escalated resource, after escalation.

plan-name
> The name of the executing plan that is associated with this thread.

package-name
> The name of the executing package that is associated with this thread, if one exists.

collection-id
> The collection identifier of the package resource, if one exists.

statement-number
> The statement number of the executing SQL statement that is associated with this plan.

correlation-id
> The correlation identifier of the thread that is associated with the escalated resource.

connection-id
> The connection identifier of the thread that is associated with the escalated resource.

luw-id　The logical unit of work identifier of the thread that is associated with the escalated resource.

thread-information
> Information about the thread. The information is presented in a colon-delimited list that contains the following segments:
>
> - The primary authorization ID that is associated with the thread.
>
>   In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.
>
> - The name of the user's workstation.
> - The ID of the user.
> - The name of the application.
> - The statement type for the currently executing statement: dynamic or static.
> - The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.
> - The name of the role that is associated with the thread.
>
> An asterisk (*) in any segment indicates that the information is not available.

System action: Processing continues.

User response: No action is required.

Related concepts:

⮕ Lock escalation (DB2 Performance)

DSNI032I　　*csect-name* **CANCEL THREAD NOBACKOUT COMMAND FAILED FOR THE THREAD =** *token*
REASON = *reason*

Explanation: DB2 displays this message when it cannot grant a request to cancel a thread without backing out data changes. The reason code explains why the request was rejected.

token　Identifies a thread whose processing you requested to cancel. The token is a 1- to 15-digit decimal number.

reason　Indicates the reason why the command failed.

System action: Processing terminates normally.

User response: The reason code specifies the action to be taken.

DSNI033I　　*csect-name* **PAGE SET** *dbname.psname* **PART** *part-number* **IS MARKED REFP AND** *status* **ON BEHALF OF UR** *ur-id*. **RECOVERY TO** *logout* **IS REQUIRED.**

Explanation: No further backout processing will be attempted on the specified page set or partition because backout for the specified unit of recovery was canceled. The specified page set or partition is placed in refresh-pending (REFP) status.

csect-name
> The name of the control section that issued the message.

dbname.psname
> The database and page set.

| *part-number*
| The value that identifies the partition or data set.

| *status*   Additional status set on the specified object.

| *ur-id*   The unit of recovery.

| *logout*   RBA or LRSN value.

| **User response:**   If the object in refresh-pending status is a table space, you can resolve the REFP,LPL state of the
| specified page set or partition in one of the following ways:

| • Run the RECOVER utility to perform a point-in-time recovery.

| • Run the LOAD utility with the REPLACE option to resolve the REFP,LPL state.

| If the page set is an index space, run the REBUILD INDEX utility to perform a point-in-time recovery and resolve the
| REFP,RBDP or REFP,LPL status. You can use the RBA or LRSN value in this message for point-in-time recovery.

| The DSNI033I message might be issued multiple times for a page set. (The message is issued once per unit of
| recovery.) If you run the RECOVER utility with point-in-time recovery to resolve the exception states, use the highest
| of the *logout* points.

| **Severity:**   0

| **Related reference**:

| ⏎ Advisory or restrictive states (DB2 Utilities)

| ⏎ RECOVER (DB2 Utilities)

| ⏎ REBUILD INDEX (DB2 Utilities)

DSNI034I   *csect-name* **TABLE SPACE** *table-space-name* **IS IN AUX WARNING STATE**

Explanation: DB2 attempted to recover table space *table-space-name*, but no log records exist because the table space
was created with the LOG NO option. The table space is placed in an auxiliary warning state. You can use SQL
statements to update the LOB values that are in error or to delete the row that contains the LOB values. In addition,
you can use SQL statements to access valid LOBs. However, you cannot use SQL statements to retrieve a LOB value
that is not valid.

System action: DB2 marks the LOB values that are in error as not valid. Processing continues.

Programmer response: Use SQL statements to update or delete the LOB values in error. Use the CHECK LOB utility
to verify that all errors were corrected and to reset the auxiliary warning state. The auxiliary warning state can also
be reset using the REPAIR utility with SET NOAUXWARN.

Severity: 0 (informational)

DSNI036I   *csect-name* **THE REAL-TIME STATISTICS COLLECTION PROCESS IS DISABLED**

Explanation: An error prevents DB2 from externalizing the in-memory statistics. The error is generally a persistent
error and must be remedied before the process can be enabled. Look for prior messages to identify the nature of the
problem.

System action: The statistics database is implicitly stopped and no further attempt will be made to externalize the
in-memory statistics.

Operator response: Notify the system programmer.

System programmer response: Inspect the console for prior messages and refer to those messages for problem
determination.

DSNI037I   *csect-name* **THE REAL-TIME STATISTICS COLLECTION PROCESS COULD NOT ACCESS AN
OBJECT IN THE REAL-TIME STATISTICS DATABASE**

Explanation: An error prevents DB2 from externalizing the in-memory statistics. An attempt failed to insert, update,
or delete a row in the statistics database. The error is generally transient (lock timeout or resource unavailable). If the
operation is retried, it is likely to succeed.

System action: The statistics are kept in-memory until the next opportunity to externalize them.

DSNI038I *csect-name* **THE REAL-TIME STATISTICS COLLECTION PROCESS IS ENABLED**

Explanation: The statistics database started successfully. The required object descriptors in the statistics database exist and have the correct attributes. DB2 proceeds to externalize in-memory statistics for all modified objects.

System action: The database is started R/w.

Severity: 0 (informational)

DSNI040I *csect-name* **RESTORE SYSTEM UTILITY PROCESSING LOG RANGE FROM** *rba/lrsn xxxxxx* **TO** *yyyyyy*

Explanation: This message periodically shows the progress of system level recovery made by the RESTORE SYSTEM utility. In non-data sharing, the log range is RBA. In data sharing, the log range represents LRSN value. The value for *xxxxxxxx* is the RBA or LRSN value of the current log record being processed and the *yyyyyyyy* value is the RBA or LRSN of the last log record to be processed by the RESTORE SYSTEM utility.

System action: System Level Point-In-Time recovery continues.

Severity: 0 (informational)

DSNI041I *csect-name* **START ACCESS(FORCE) CANNOT BE ISSUED AGAINST TARGET** *obj-type obj-qual.obj-name part* **DUE TO STOPPED UTILITIES ON VERSION** *version-number*. **REQUEST IS REJECTED.**

Explanation: The object cannot be started due to outstanding utilities on a different version. The status of the object is unchanged.

version-number is the version on which the utility was running.

obj-type is table space or index space.

obj-qual is the database name.

obj_name is the table space or index space name.

part represents the partition number for a partitioned table space.

System action: The request is rejected.

User response: Use the DISPLAY UTILITY command to see what utilities are outstanding. Issue the -START DATABASE ACCESS(FORCE) command on the same version as the version on which the stopped utility was running. If there is more than one utility outstanding on the object and the utilities were submitted on a different release of DB2, start the correct version of DB2 and use the -TERM UTILITY command to terminate each utility.

DSNI042I *csect-name* **- PAGE SET HEADER RBA** *function-name*- **ID** *function-id* **DB ID** *database-id*. **PS ID** *pageset-id* **PART #** *partition-number* **RBA** *header-page* **RBA**

Explanation: This informational message displays the header page RBA that is used to determine the LPL or GRECP recovery range for the specified page set. It is displayed one time for each LPL recovery, for each page set and each data sharing group.

csect-name
> The name of the control section that issued the message.

command-name
> The name of the MVS command that was issued: START, STOP, or MODIFY. The MODIFY command refreshes the DB2 health monitor.

function-name
> The name of the function, such as a START DATABASE command, that was issued.

function-id
> A decimal number that represents the function.

database-id
> The ID of the database where the page set exists.

pageset-id
> The ID of the page set where the pages exist.

partition-number
> The number of the partition where the pages exist.

header-page
> The RBA value that is on the header page of the page set.

System action: The recovery continues.

User response: Wait for message DSNI021I, which indicates the end of recovery, or for message DSNI005I, which indicates an error occurred during recovery.

Related information:

"DSNI005I" on page 214

"DSNI021I" on page 220

DSNI043I   *csect-name* **- MERGED LOG APPLY RANGE** *function-name***-ID** *function-id* **MEMBER ID** *member-id* **FROM** *type sequence-start* **TO** *type sequence-end*

Explanation: This informational message displays the broad LRSN or RBA range, merged from all members of the data sharing group. The range is used to determine the LPL or GRECP recovery range for the specified page set. It is displayed on time for each LPL recovery, for each member.

csect-name
> The name of the control section that issued the message.

command-name
> The name of the MVS command that was issued: START, STOP, or MODIFY. The MODIFY command refreshes the DB2 health monitor.

function-name
> The name of the function, such as a START DATABASE command, that was issued.

function-id
> A decimal number that represents the function.

member-id
> The numeric identifier for the data sharing member.

type   An indicator of the log record sequence number referenced in the message:

> **LRSN**   The log record sequence number is used in a data sharing environment.

> **RBA**   The relative byte address is used in a non-data sharing environment.

sequence-start
> Indicator of the beginning of the sequence identified by *type*.

sequence-end
> Indicator of the end of the sequence identified by *type*.

System action: The recovery continues.

User response: Wait for message DSNI021I, which indicates the end of recovery, or for message DSNI005I, which indicates an error occurred during recovery.

Related information:

"DSNI005I" on page 214

"DSNI021I" on page 220

DSNI044I   *csect-name*- **PAGE LOG APPLY RANGE** *function-name* **-ID** *function-id* **DB ID** *database-id* **PS ID** *pageset-id* **PART #** *partition-number* **MEMBER ID** *member-id* **FROM PAGE #** *from-page* **TO PAGE #** *to-page* **FROM** *type sequence-start* **TO** *type sequence-end*

Explanation: This informational message displays the LRSN or RBA range that is used to determine LPL or GRECP recovery range for the specified page set in the data sharing group member.

csect-name
>   The name of the control section that issued the message.

command-name
>   The name of the MVS command that was issued: START, STOP, or MODIFY. The MODIFY command refreshes the DB2 health monitor.

function-name
>   The name of the function, such as a START DATABASE command, that was issued.

function-id
>   A decimal number that represents the function.

database-id
>   The ID of the database where the page set exists.

pageset-id
>   The ID of the page set where the pages exist.

partition-number
>   The number of the partition where the pages exist.

member-id
>   The numeric identifier for the data sharing member.

from-page
>   The beginning page number for the range where the log is to be applied.

to-page   The ending page number for the range where the log is to be applied.

type   An indicator of the log record sequence number referenced in the message:

>   **LRSN**   The log record sequence number is used in a data sharing environment.

>   **RBA**   The relative byte address is used in a non-data sharing environment.

sequence-start
>   An indicator of the beginning of the sequence identified by *type*.

sequence-end
>   An indicator of the end of the sequence identified by *type*.

System action: The recovery continues.

User response: Wait for message DSNI021I, which indicates the end of recovery, or for message DSNI005I, which indicates an error occurred during recovery.

Related information:

"DSNI005I" on page 214

"DSNI021I" on page 220

DSNI045I   **OPEN FAILED FOR DB** *dbname* **SPACE NAME** *tsname*, **PART** *partno*

Explanation: The data set for the database name, space name, and partition number failed to be physically opened through an -ACCESS DATABASE command with keyword MODE(OPEN). PART will be unavailable for non-partitioned data sets.

Operator response: Examine the console for other related error messages. Ensure the bufferpool is activated, and then re-issue the command. If the failure persists, then contact DB2 Technical Support.

DSNI046I	*csect-name* **OBJECT IN ERROR REASON** *reason-code* **ERQUAL** *error-qualifier* **NAME** *object-name* **DBID** *database-id* **OBID** *object-id* **LRSN** *log-record*

Explanation: An error occurred during processing of a table space, table space partition, index space, index partition, or logical index partition. One of the following conditions occurred:

- An error occurred when DB2 processed a DBET exception state for a table space, index space, or partition of a partitioned object.
- An error occurred when DB2 was unable to access a page set in backward log recovery phase during restart.

csect-name
> The name of the control section that issued the message.

reason-code
> The DB2 reason code that is listed in the dump that accompanies this message.

error-qualifier
> The abend in the control section that caused this message to be issued.

object-name
> The name of the table space or index space that encountered the problem. A partition number is included where applicable.

database-id
> The database identifier (DBID).

object-id
> The object identifier (OBID).

log-record
> The LRSN or RBA of the log record that was being processed when the error occurred. If no log record was being processed, this value is blank.

System action: The system action depends on the situation in which the error occurred:

Accessing a page in backward log recovery phase
> DB2 places the table space or index space in RECP and DBETE status. It then abends and generates a dump. DB2 tolerates the page set access error so that it can restart.

Processing a DBET exception state
> DB2 places the table space or index space in RECP, RBDP, or PSRBD status and DBETE status. It then abends and generates a dump. DB2 tolerates the exception state error so that it can restart or the RESTORE SYSTEM utility can continue.

User response: Issue the appropriate DISPLAY command to see the status of the object. Use one of the following methods to reset the DBETE and RECP, RBDP, or PSRBD statuses:

- Use the RECOVER utility or the LOAD utility with the REPLACE option to recover a table space or index space.
- Use the REBUILD INDEX utility to rebuild an index space.
- Use the START DATABASE command with the following options:

```
START DATABASE(database-name)
  SP(space-name) ACCESS(FORCE)
```

Contact IBM Software Support to report the problem. DB2 log records need to be analyzed in order to diagnose the cause of the problem, and determine if the exception state error needs to be handled.

Related concepts:

⇨ DB2 reason codes (DB2 Codes)

Related reference:

⇨ -START DATABASE (DB2) (DB2 Commands)

⇨ LOAD (DB2 Utilities)

⇨ REBUILD INDEX (DB2 Utilities)

⇨ RECOVER (DB2 Utilities)

DSNI047I   *csect-name* **UNABLE TO ACCESS OBJECT TYPE** *object-type* **OBJECT NAME** *object-name* **REASON**
        reason **TYPE** *type* **NAME** *name*

Explanation: The resource identified could not be accessed because it was unavailable. The resource could be a database, a table space, an index space, or a partition within a table space or index space. The *object-type* identifies the type. The *object-name* identifies the name. If the DB2 catalog is unavailable, then *object-type* contains binary zeros and *object-name* contains blanks.

The resource that was unavailable is identified by *type* and *name*. The *type* code identifies the type of the resource, as well as the format and content of the *name*. The *reason* contains the reason code explaining why the resource was unavailable.

System action: The specified database, space, or partition is not accessed, and remains in tis current state. If there are other databases , spaces, or partitions to be accessed, they are processed normally.

User response: Correct the resource unavailable problem and reissue the -ACCESS DATABASE command.

Related information:

Appendix C, "Resource types," on page 1153

DSNI048I   *mod-name* **CLAIMERS EXIST FOR DATABASE** *dbname*, **SPACE NAME** *tsname*, **PART** *partno*. **GROUP**
        BUFFERPOOL DEPENDENCY CANNOT BE REMOVED.

Explanation: An -ACCESS DATABASE MODE(NGBPDEP) command was issued while one or more claimers still exist for the indicated object. DB2 cannot remove group buffer pool dependency while the claimers still exist. PART is not available for non-partitioned page sets.

Operator response: Use the -DISPLAY DATABASE CLAIMERS command to determine which threads hold claims that are blocking the ACCESS DATABASE MODE(NGBPDEP) command. Reissue the command after these threads have released their claims. Only write claimers will block the command on the member to which the command is issued. For all other members, either a read claimer or a write claimer will block the command.

DSNI049I   *csect-name* **GRECP OR LPL RECOVERY FOR** *function-name* **- ID =** *function-id* **HAS COMPLETED.**

Explanation: The group buffer pool recovery pending (GRECP) or logical page list (LPL) exception state of table spaces or index spaces in a recovery function has been completed.

csect-name
        The name of the control section that issued the message.

function-name
        The function that triggered the recovery: Expected values are:
        • START DATABASE COMMAND
        • AUTOMATIC GRECP RECOVERY
        • AUTOMATIC LPL RECOVERY

function-id
        A decimal number that represents a recovery function. This number corresponds to the function ID number
        in DSNI006I, DSNI021I, and DSNI051I messages.

DSNI051I

| Messages DSNI006I (beginning of recovery), DSNI051I (beginning of second pass recovery), and DSNI021I (end of
| recovery) report about recovering a single table space or index space within a recovery function. A recovery function
| can recover multiple objects.

| Message DSNI049I indicates the end of a recovery function.

| **System action:** The recovery function that is identified by *function-name* completes normally.

| **Operator response:** No action is required.

| **Related reference**:

| ➡ -START DATABASE (DB2) (DB2 Commands)

| **Related information**:

| "DSNI006I" on page 215

| "DSNI021I" on page 220

| "DSNI051I"

| **DSNI051I**   *csect-name* **ASYNCHRONOUS GRECP OR LPL RECOVERY IN PROGRESS FOR** *function-name* **- ID =**
| *function-id* **TYPE** *object-type* **NAME** *object-name*

| **Explanation:** The second pass log apply has started for an object with group buffer pool recovery pending (GRECP)
| or logical page list (LPL) status.

| *csect-name*
| The name of the control section that issued the message.

| *function-name*
| The function that triggered the recovery: Expected values are:
| • START DATABASE COMMAND
| • AUTOMATIC GRECP RECOVERY
| • AUTOMATIC LPL RECOVERY

| *function-id*
| A decimal number that represents the recovery function.

| *object-type*
| The type of object, which is an index space.

| *object-name*
| The name of the object.

| **System action:** The object is started and the second pass log apply for GRECP or LPL recovery is initiated. During
| the LPL or GRECP recovery process, additional messages might be issued, including:

| • DSNI005I, which indicates an error

| • DSNI006I, which indicates the start of recovery

| • DSNI021I, which indicates successful completion

| • DSNI022I, which provides the status of recovery operations

| **User response:** No action is required.

| You can issue a DISPLAY DATABASE command with the RESTRICT option for the specified index space to
| determine whether the GRECP or LPL exception state has been cleared.

| **Related reference**:

| ↪ -DISPLAY DATABASE (DB2) (DB2 Commands)

| **Related information**:

| "DSNI005I" on page 214

| "DSNI006I" on page 215

| "DSNI021I" on page 220

| "DSNI022I" on page 220

| **DSNI052I**   *csect-name* **AN AGENT HAS EXCEEDED THE WARNING THRESHOLD FOR STORAGE USE IN WORK FILE DATABASE** *database-name* **FOR** *object-type***. THRESHOLD=***percentage* **PERCENT. TOTAL STORAGE CONFIGURED=***total-storage* **KB. CONNECTION ID =** *connection-id***. CORRELATION ID =** *correlation-id***. LUWID=***luwid***. PACKAGE NAME=***package-name***. PLAN NAME=***plan-name***.**

| **Explanation:** The total amount of storage that an agent used in the work file database exceeded the threshold that is defined by subsystem parameter WFSTGUSE_AGENT_THRESHOLD.

| *csect-name*
| The name of the control section that issued the message.

| *database-name*
| The name of the work file database.

| *object-type*
| The type of object. Possible values are:

| **DECLARED GLOBAL TEMP TABLES**
| Specifies that the storage is being used for declared temporary tables.

| **WORK FILES**
| Specifies that the storage is being used for work files.

| **DECLARED GLOBAL TEMP TABLES AND WORK FILES**
| Specifies that the storage is being used for declared temporary tables and work files.

| *percentage*
| The maximum percentage of work file database storage that an agent can use before this message is issued. This value is the value of subsystem parameter WFSTGUSE_AGENT_THRESHOLD.

| *total-storage*
| The total amount of storage that is configured for the specified object type, in kilobytes.

| *connection-id*
| The connection identifier of the agent.

| *correlation-id*
| The correlation identifier of the agent.

| *luwid*   The logical unit of work ID of the agent. An asterisk (*) denotes that the thread originated at the local site.

| *package-name*
| The package name that is associated with the agent.

| *plan-name*
| The plan name that is associated with the agent.

| This message is issued once within each commit scope in the application.

| **System action:** Processing continues.

| **System programmer response:** Work with the application programmer to determine how the application program is using space in the work file database.

| Determine whether the threshold that is specified by subsystem parameter WFSTGUSE_AGENT_THRESHOLD is too low.

| If WFSTGUSE_AGENT_THRESHOLD is not too low, evaluate the type of work file activity that the agent is performing. Take one or both of the following actions:

DSNI053I

| • Create additional table spaces in the work file database of the object types that are specified in this message.

Use the DSNTWFG exec in job step DSNTIST of installation job DSNTIJTM to create additional table spaces.

| • Change the value of subsystem parameter WFDBSEP from NO to YES.

Work file table spaces can be used for declared temporary tables or for work file work, such as sorting. When subsystem parameter WFDBSEP is set to NO, DB2 selects work file table spaces that can grow into secondary extents for declared temporary tables, and chooses work file table spaces that cannot grow into secondary extents for work file work. If there is not enough space in one type of table space, DB2 switches to the other type. When DB2 uses work file table spaces that are designated for work file activity for declared temporary tables, space for work file activity might be quickly consumed. When DB2 uses work file table spaces that are designated for declared temporary tables for work file activity, space for declared temporary table tables might be rapidly consumed because of growth through secondary extents. To prevent DB2 from switching from one type of table space to the other, set WFDBSEP to YES. Doing so can cause high-consuming agents to experience resource unavailable conditions, but can help protect other agents on the system that are using work file resources.

Programmer response: Use the agent information in the message to determine which application exceeded space use threshold WFSTGUSE_AGENT_THRESHOLD. Examine the application to determine the ways in which the application is using space in the work file database.

Related tasks:

▶ Installation step 16: Create default storage group, define temporary work files, and bind DB2 REXX Language Support: DSNTIJTM (DB2 Installation and Migration)

Related reference:

▶ AGENT LEVEL THRESHOLD field (WFSTGUSE_AGENT_THRESHOLD subsystem parameter) (DB2 Installation and Migration)

▶ SEPARATE WORK FILES field (WFDBSEP subsystem parameter) (DB2 Installation and Migration)

| **DSNI053I** *csect-name* **THE DB2 SUBSYSTEM HAS EXCEEDED THE THRESHOLD FOR STORAGE USE IN WORK FILE DATABASE** *database-name* **FOR** *object-type*. **THRESHOLD=***percentage* **PERCENT. TOTAL STORAGE CONFIGURED=***total-storage* **KB.**

Explanation: This message warns of a possible storage shortage in the work file database. The amount of storage that is being used has exceeded the system-level warning threshold that is defined by subsystem parameter WFSTGUSE_SYSTEM_THRESHOLD.

csect-name
The name of the control section that issued the message.

database-name
The name of the work file database.

object-type
The type of object. Possible values are:

DECLARED GLOBAL TEMP TABLES
Specifies that the storage is being used for declared temporary tables.

WORK FILES
Specifies that the storage is being used for work files.

DECLARED GLOBAL TEMP TABLES AND WORK FILES
Specifies that the storage is being used for declared temporary tables and work files.

percentage
The maximum percentage of work file database storage that is used before this warning is issued. This value is the value of subsystem parameter WFSTGUSE_SYSTEM_THRESHOLD.

total-storage
The total amount of storage that is configured for the work file database, in kilobytes.

This message is issued at five-minute intervals, for as long as the criteria for issuing the message are met.

System action: Processing continues.

If the situation is not alleviated, applications that use the work file database might fail.

System programmer response: Determine whether the warning threshold that is specified by subsystem parameter WFSTGUSE_SYSTEM_THRESHOLD is too low.

If WFSTGUSE_AGENT_THRESHOLD is not too low, take one or both of the following actions:

- Create additional table spaces in the work file database of the object types that are specified in this message.

  Use the DSNTWFG exec in job step DSNTIST of installation job DSNTIJTM to create additional table spaces.

- Change the value of subsystem parameter WFDBSEP from NO to YES.

  Work file table spaces can be used for declared temporary tables or for work file work, such as sorting. When subsystem parameter WFDBSEP is set to NO, DB2 selects work file table spaces that can grow into secondary extents for declared temporary tables, and chooses work file table spaces that cannot grow into secondary extents for work file work. If there is not enough space in one type of table space, DB2 switches to the other type. When DB2 uses work file table spaces that are designated for work file activity for declared temporary tables, space for work file activity might be quickly consumed. When DB2 uses work file table spaces that are designated for declared temporary tables for work file activity, space for declared temporary table tables might be rapidly consumed because of growth through secondary extents. To prevent DB2 from switching from one type of table space to the other, set WFDBSEP to YES. Doing so can cause high-consuming agents to experience resource unavailable conditions, but can help protect other agents on the system that are using work file resources.

Related tasks:

Installation step 16: Create default storage group, define temporary work files, and bind DB2 REXX Language Support: DSNTIJTM (DB2 Installation and Migration)

Related reference:

SYSTEM LEVEL THRESHOLD field (WFSTGUSE_SYSTEM_THRESHOLD subsystem parameter) (DB2 Installation and Migration)

Chapter 8. DSNJ messages

Message identifiers that begin with DSNJ communicate information about the recovery log manager.

DSNJ001I   *csect-name* **CURRENT COPY** *n* **ACTIVE LOG DATA SET IS DSNAME=..., STARTRBA=..., ENDRBA=...**

Explanation: This message is generated for one of two reasons:

1. When the log manager is initialized, it sends this informational message to identify the current active log data sets (copy-1 and, if dual logging is used, copy-2).
2. When the current active log data set is full, DB2 will switch to the next available active log data set. This message identifies the next available active log data set which will be used for logging.

The value specified by the STARTRBA=value keyword is the RBA of the first byte of log data in the named data set. The value specified by the ENDRBA=value keyword is the RBA of the last possible byte in the data set. These values may later be needed for a conditional restart.

System programmer response: None required. However, if recovery is required, information from this message may be required as input to the change log inventory (DSNJU003) utility.

DSNJ002I   **FULL ACTIVE LOG DATA SET DSNAME=..., STARTRBA=..., ENDRBA=...**

Explanation: This message is sent at the time the log manager fills an active log data set and switches to a new empty data set. The message shows the name and log RBA range of the full data set.

System programmer response: None required. However, if recovery is required information from this message may be required as input to the change log inventory (DSNJU003) utility.

DSNJ003I   *csect-name* **FULL ARCHIVE LOG VOLUME DSNAME=..., STARTRBA=..., ENDRBA=..., STARTxxxx=..., ENDxxxx=..., UNIT=..., COPYnVOL=..., VOLSPAN=..., CATLG=...**

Explanation: Offloading for the specified archive log data set was successfully completed for the given volume. If the data set spans multiple tape volumes, this message is generated for each tape volume. If the data set spans multiple DASD volumes, only one message will be generated displaying the first VOLSER, appended with a plus sign ('+') to indicate the data set spans multiple volumes. 'xxxx' is replaced by either TIME or LRSN.

System action: An archive log data set has been created, and the archive log data set inventory in the bootstrap data set (BSDS) has been updated with the information in the message:

DSNAME
    The name of the archive log data set.

STARTRBA
    The starting RBA contained in the volume.

ENDRBA
    The ending RBA contained in the volume.

STARTLRSN
    The starting LRSN contained in the volume for data sharing systems.

ENDLRSN
    The ending LRSN contained in the volume for data sharing systems.

STARTTIME
    The starting store-clock value of the log records in the volume for non-data sharing systems.

ENDTIME
    The ending store-clock value of the log records in the volume for non-data sharing systems.

UNIT   The device unit to which the data set was allocated.

COPYnVOL

The name of the volume. This message keyword is displayed as COPY1VOL if this is the copy-1 archive log data set, and as COPY2VOL if this is the copy-2 archive log data set.

VOLSPAN

An indicator to denote one of four conditions:

- If '00', the data set is entirely contained on the volume specified by 'COPYnVOL'.
- If '01', this is the first entry of a multivolume data set.
- If '11', this is the middle entry of a multivolume data set.
- If '10', this is the last entry of a multivolume data set.

CATLG

An indicator to denote one of two conditions:

- If 'NO', the archive log data set is uncataloged.
- If 'YES', the archive log data set is cataloged.

The BSDS is automatically updated with the information contained in this message; however, if recovery is required, information from this message may be required as input to the change log inventory utility (DSNJU003).

DSNJ004I   **ACTIVE LOG COPY n INACTIVE, LOG IN SINGLE MODE, ENDRBA=...**

Explanation: This message is sent when the dual active logging option is selected and copy 'n' becomes inactive. A log copy becomes inactive when the next active log data set is not ready when required. ENDRBA is the last byte of log data written on copy 'n'. This is usually caused by a delay in off-load.

System action: The log is switched to single mode until the next data set for copy 'n' is ready for logging.

If DB2 is shut down or abends while in single mode with the ZPARM option still set for dual active data sets, the previous state of the active log data sets determines what happens when DB2 is started, as follows:

- If fewer than two data sets are available (not flagged as STOPPED) for each set of active logs, DB2 startup terminates and message DSNJ112E is issued.
- If an active log data set is in NOTREUSABLE state, DB2 can be started in single logging mode, but dual mode takes effect when the other active log data set becomes available after offloading.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Execute an MVS *Display System Requests* command (D R,L for example) to ensure that there are no outstanding requests that are related to the log off-load process. Take the necessary action to satisfy any requests, and permit off-load to continue.

System programmer response: If the switch to single mode was caused by the lack of a resource required for off-load, the necessary resource should be made available to allow off-load to complete and thus permit dual logging to proceed. If recovery is required, information from this message might be required as input to the change log inventory (DSNJU003) utility.

DSNJ005I   **ACTIVE LOG COPY n IS ACTIVE, LOG IN DUAL MODE, STARTRBA=...**

Explanation: This message is sent when copy 'n' of the log becomes active after previously being flagged as inactive. STARTRBA is the RBA of the first byte of log data written on copy 'n' after it was activated.

System programmer response: None required. However, if recovery is required, information from this message may be required as input to the change log inventory (DSNJU003) utility.

DSNJ006I   **ALLOCATION FOR NEW ARCHIVE LOG DATA SET HAS BEEN CANCELLED BY OPERATOR.**

Explanation: This message is sent if the operator answers 'N' to the WTOR message DSNJ008E.

System action: If the allocation is for copy 1 archive log data set, offload terminates processing until the next time it is activated. If copy 1 has already been allocated and this request is for copy 2, offload switches to single offload mode for this data set only.

DSNJ007I   *csect-name* **OPERATOR CANCELED MOUNT OF ARCHIVE VOL SER=** *volser*

Explanation: If the operator answers 'N' to the WTOR message DSNJ009I, this message is issued. The variable 'volser' is the volume serial of an archive log volume required for the read request.

System action: The read request for the archive log volume fails. If the request was issued with the COND=YES parameter, return code 12 and reason code 00D1032B are returned to the requester. Otherwise, DB2 abends with reason code 00D1032B.

DSNJ008E   *nn* **OF** *mm* **ACTIVE LOGS ARE FULL.** *ssname* **NEEDS ARCHIVE SCRATCH. REPLY Y WHEN DEVICE READY OR N TO CANCEL.**

Explanation: The installation has requested a WTOR before archive log data set allocation. DB2 needs a scratch volume for offloading an active log data set.

ssname   The name of the DB2 subsystem.

nn   The number of full active log data sets.

mm   The total number of active log data sets.

System action: The offload task waits for the operator's reply.

Operator response: The operator has three options:
- Get a scratch volume ready, make sure there is an available unit for the volume, and reply 'Y'. DB2 then continues with the offload.
- Determine from the number of active log data sets available whether the offload can be delayed until the next time an active log data set becomes full. If the process can be delayed, then reply 'N'.

  This response has two possible effects.
  - If dual archive logging is in effect and this allocation is for a copy 1 archive data set, the 'N' response delays the offload process until the next active log data set becomes full.
  - However, if the copy 1 archive data set has already been allocated and this request is for copy 2, the 'N' response causes the offload to switch to single archive mode (the switch is for this data set only).
- Defer giving a response. This causes offload to wait before processing. However, because offload is a separate service task, the wait does not affect DB2 performance.

If offloading to DASD, an error has occurred attempting to allocate an archive log data set. Reply 'Y' to receive the error messages.

DSNJ009E   *ssname* **NEEDS VOL SER=nnnnnn. REPLY Y TO CONTINUE OR N TO CANCEL**

Explanation: The installation requested WTOR before archive log data set allocation. DB2 needs the specified archive volume for a read operation. The subsystem name is 'ssname'.

System action: The archive log read service task waits for the operator's reply. This wait affects the agent for which the log read was issued and any other agents that might be waiting on the log read service task queue.

Operator response: Locate the requested volume, ensure that a device is available, and reply 'Y'. DB2 continues with dynamic allocation and then begins reading the log.

If dual archiving is in effect, a response of 'N' causes archive read to reissue the message for the copy 'Z' copy 2 archive VOLSER with the same RBA range. A response of 'N' to this second WTOR, or to the initial WTOR for single archiving, causes the archive read service task to fail with unpredictable results.

DSNJ010I   *csect-name* **INVALID RESPONSE. WTOR WILL BE REPEATED.**

Explanation: A WTOR was issued during archive data set allocation. The user did not respond correctly to the message. Either 'Y' or 'N' must be entered.

System action: The original WTOR is repeated.

Operator response: Reply as indicated in the repeated message.

DSNJ011I   **CONDITIONAL RESTART RECORD** *record-identifier* **CREATED AT** *timestamp* **WAS FOUND. REPLY Y TO USE, N TO CANCEL**

Explanation: When the log manager was initialized, a conditional restart control record was found in the bootstrap data set.

This message provides the following information to help identify the conditional restart record that will be used:

- A record identifier, which is a 4 byte hexadecimal number
- The creation timestamp, in the format *yy.ddd hh:mm*:

    yy   Two-digit indicator for the year.

    ddd   Three-digit indicator for the day of the year.

    hh:mm   Indicator for the time of the day.

System action: The system action depends on the reply.

'Y'   DB2 is started conditionally, using the record that was found.

'N'   DB2 startup terminates.

Operator response: If a DB2 conditional restart is to be done and if the date and time displayed are correct, then reply 'Y' to the outstanding message. If you are not sure, or if you do not want to use the indicated record, contact the system programmer.

System programmer response: If a normal restart of DB2 has failed and you have created a conditional restart control record with the change log inventory (DSNJU003) utility, check whether the time and date in the message agree with when you created that record. If they do agree, reply 'Y' to the outstanding WTOR. If they do not, reply 'N' and investigate the discrepancy.

Related concepts:

➡ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

Related reference:

➡ DSNJU003 (change log inventory) (DB2 Utilities)

➡ DSNJU004 (print log map) (DB2 Utilities)

DSNJ012I   **ERROR ccc READING RBA rrr IN DATA SET ddd CONNECTION-ID=xxxxxxxx, CORRELATION-ID=yyyyyyyyyyyy**

Explanation: While scanning log records read into a buffer, the log manager detected a logical error with reason code 'ccc'. 'rrr' is the log RBA of the segment in the buffer at which the error was detected. 'ddd' is the name of the active or archive log data set from which the record was read. If 'ddd' is blank, the data was read from an active log output buffer.

The connection and correlation IDs identify the user or application that encountered the problem. Messages that have the same connection ID and correlation ID relate to the same user. Correlation IDs beginning with '0nn' where 'nn' is a number from 01 to 28, identify system agents.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from log failures.

System action: The application program that invoked the log manager is terminated with reason code 'ccc'. However, information in this message may be useful in diagnosing the abend that will follow.

DSNJ013I   **TERMINAL ERROR ccc IN BUFFER rrr BEFORE ACTIVE LOG WRITE**

Explanation: A scan of the log output buffer just prior to writing the buffer detected an inconsistency in the log data. 'ccc' is the abend reason code associated with the SDUMP that is produced. 'rrr' is the log RBA at which the error was detected.

System action: DB2 will terminate with a dump, and will not write the damaged buffer to either COPY 1 or COPY 2 active log data set.

Operator response: Notify the system programmer. Start DB2 after it terminates.

System programmer response: Because the damaged buffer has not been written to a log data set, DB2 may be restarted. No corrective action is required.

Problem determination: In the DB2 formatted block section of the dump, locate the chain of LBCE blocks. Those are log buffer control elements, and each LBCE controls one 4KB log output buffer. LBCEBUFF, at offset X'20' in the LBCE, points to the buffer, which is always on a 4KB boundary. The log RBA of the first byte of the buffer is in a 6-byte field at offset X'FF2' in the buffer itself. The layout of the buffer is described by the DSNDLCID mapping macro. The header of each record within the buffer is described by the DSNDLRH mapping macro. Locate the buffer that contains the log RBA from the message and find the damage by referring to the description of the reason code later in this book.

DSNJ014I   **TERMINAL ERROR ccc IN BUFFER rrr AFTER ACTIVE LOG WRITE.**

Explanation: A scan of the log output buffer, after writing to the COPY 1 active log data set and before writing to COPY 2, detected an inconsistency in the log data. 'ccc' is the abend reason code associated with the SDUMP that is produced. 'rrr' is the log RBA at which the error was detected.

System action: DB2 will terminate with a dump, and will not write the damaged buffer to the COPY 2 data set.

Operator response: Notify the system programmer.

System programmer response: The block containing the indicated log RBA may be damaged. The buffer was found to be in error at the completion of the write to the COPY 1 data set of the active log.

If dual active logs are being used, use print log map (DSNJU004) to list the active log data sets for both copies of the active log. Find the COPY 2 data set with the corresponding RBA, and copy that data set (using access method services REPRO) to the COPY 1 data set. Start DB2.

If only a single active log is used, contact the IBM support center for assistance. An attempt to start DB2 may succeed if the damage to the buffer occurred after completion of the write to DASD.

Problem determination: In the DB2 formatted block section of the dump, locate the chain of LBCE blocks. Those are log buffer control elements, and each LBCE controls one 4KB log output buffer. LBCEBUFF, at offset X'20' in the LBCE points to the buffer, which is always on a 4KB boundary. The log RBA of the first byte of the buffer is in a 6-byte field at offset X'FF2' in the buffer itself. The layout of the buffer is described by the DSNDLCID mapping macro. The header of each record within the buffer is described by the DSNDLRH mapping macro. Locate the buffer that contains the log RBA from the message and find the damage by referring to the description of the reason code later in this book.

DSNJ015I   **CSRONLY PROHIBITS LOG ALTERATIONS. RECORD rrr CREATED .... REPLY Y TO USE, N TO CANCEL.**

Explanation: During DB2 startup, the log manager found an inactive conditional restart control record in the bootstrap data set. Because this control record indicates CSRONLY, no log data sets will be modified. If log truncation was specified, it will be simulated. Only the first phase of restart (current status rebuild) will be performed. The hexadecimal identifier (rrr) and creation timestamp (yy.ddd hh:mm) identify the record found.

System action: A 'Y' reply will continue the conditional restart. A 'N' reply will terminate DB2 startup.

Operator response: To continue this CSRONLY processing, reply 'Y' to this message. If you are not sure, contact the system programmer.

System programmer response: The conditional restart control record will not be deactivated at the end of the CSRONLY processing. To deactivate this record, either cancel this record or create a new record. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for more information.

DSNJ016E   *csect-name* **WARNING - SYSTEM CHECKPOINT PROCESSOR MAY HAVE STALLED. LAST CHECKPOINT WAS TAKEN** *date-time*.

Explanation: While switching active log data sets, DB2 found that there has not been a system checkpoint taken since the last time DB2 switched active log data sets. *date-time* is the date and time of the last system checkpoint. There should be at least one system checkpoint on each log data set.

DSNJ017E • DSNJ030I

System action: DB2 continues processing. If statistics class 3 is active, IFCID 0335 is written.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

This message is deleted when a system checkpoint is taken.

System programmer response: This message might be an indication that the system checkpoint processor may no longer be functioning. This situation can result in a very long restart if logging continues without a system checkpoint. You should monitor system checkpoint activity very closely. You can use the -DISPLAY LOG command or run the Print Log Map utility (DSNJU004) to display the most recent checkpoint. If DB2 continues logging beyond the defined checkpoint frequency, you should quiesce activity and terminate DB2 to minimize the restart time.

Prior to terminating DB2, obtain a dump of the DB2 address spaces for problem determination.

DSNJ017E   *csect-name* **WARNING - OFFLOAD TASK HAS BEEN ACTIVE SINCE** *date-time* **AND MAY HAVE STALLED.**

Explanation: While switching active log data sets, DB2 found that the offload task has been active since the last time DB2 switched active log data sets. The offload task began archiving an active log data set at *date-time*.

System action: DB2 continues processing. If statistics class 3 is active, IFCID 0335 is written. This message will be deleted when the offload task completes the archive process.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

System programmer response: This may be an indication that there may be an outstanding tape mount request or that the offload task may no longer be functioning. This situation can result in a log-full situation if DB2 can not archive log data sets when they fill. You should reply to any tape mount requests and monitor offload activity very closely. You can issue the -DISPLAY LOG command to display the status of the offload task, and the -ARCHIVE LOG CANCEL OFFLOAD command can be used to terminate and restart the offload process.

Prior to terminating the offload task, obtain a dump of the DB2 MSTR address space for problem determination.

DSNJ020I   *csect-name* **RECEIVED REPLY OF N TO** *msg-num*. **SUBSYSTEM STARTUP IS TERMINATED.**

Explanation: The operator chose to terminate DB2 startup by answering 'N' to 'msg-num'.

System action: DB2 will not restart.

Operator response: To restart DB2, follow the 'operator response' of 'msg-num'.

DSNJ030I   **RBA RANGE** *start-rba* **TO** *end-rba* **NOT AVAILABLE IN ACTIVE LOG DATA SETS**

Explanation: Previous errors have made the active log data sets that contain the relative byte address (RBA) range reported in the message unavailable to DB2. The status of these logs is STOPPED in the bootstrap data set (BSDS).

System action: DB2 terminates with a dump.

Operator response: Notify the system programmer.

System programmer response: The log RBA range must be available for DB2 to be recoverable. Correct previous errors and restore the active log data set(s) that contain the RBA range reported in the message.

- If the log data sets are recoverable, the active log data set inventory in the BSDS must be modified to reset the STOPPED status.
  1. Use the print log map (DSNJU004) utility to obtain a copy of the BSDS log inventory.
  2. Use the change log inventory (DSNJU003) utility to delete the active log data sets that are marked STOPPED, by using DELETE statements.
  3. Add the data sets again by using NEWLOG statements of the DSNJU003 utility. The starting and ending RBA for each active log data set must be specified on the NEWLOG statements.
- If the log data sets are not recoverable, conditionally restart DB2.

Problem determination: Examine previous messages to determine the reason the active log data sets are unavailable to DB2.

Related concepts:

➡ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

➡ Conditional restart (DB2 Administration Guide)

➡ Log RBA range (DB2 Administration Guide)

Related reference:

➡ DSNJU003 (change log inventory) (DB2 Utilities)

➡ DSNJU004 (print log map) (DB2 Utilities)

Related information:

➡ Recovering from active log failures (DB2 Administration Guide)

➡ Recovering from archive log failures (DB2 Administration Guide)

DSNJ031I   *csect-name* **WARNING - UNCOMMITTED UR HAS WRITTEN** *number* **LOG RECORDS - CORRELATION NAME =** *correlation-name* **CONNECTION ID =** *connection-id* **LUWID =** *logical-unit-of-work-id=token* **PLAN NAME =** *plan-name* **AUTHID =** *authorization-id* **END USER ID =** *user-id* **TRANSACTION NAME =** *transaction-name* **WORKSTATION NAME =** *workstation-name*

Explanation: A unit of recovery (UR) has reached the threshold number of log records that were written without a commit or rollback operation. This threshold was defined during installation in subsystem parameter URLGWTH.

csect-name
> The name of the control section that issued the message.

number   The cumulative number of log records that were written since the beginning of the UR.

correlation name
> The correlation name that is associated with the UR.

connection-id
> The connection ID that is associated with the UR.

logical-unit-of-work-id
> The logical unit of work ID. An asterisk (*) denotes that the thread originated at the local site.
>
> This value consists of a fully qualified LU network name followed by a period and an LUW instance number. The LU network name consists of a one- to eight-character network ID, a period, and a one- to eight-character network LU name. The LUW instance number consists of 12 hexadecimal characters that uniquely identify the unit of work.

token   A unique token number that is associated with the logical unit of work ID.

plan-name
> The plan that is associated with the UR.

authorization-id
> The authorization ID that is associated with the UR.

user-id   The user ID that is associated with the UR, if known. Otherwise, an asterisk (*).

transaction-name
> The transaction that is associated with the UR, if known. Otherwise, an asterisk (*).

workstation-name
> The workstation host name that is associated with the UR, if known. Otherwise, an asterisk (*).

System action: Processing continues. If statistics class 3 is active, IFCID 0313 is written.

This message is repeated each time the threshold is reached.

System programmer response: Consult with the application programmer to determine if this UR is problematic.

If an application program caused the UR to reach its threshold number of log records, you can use CANCEL THREAD (*token*) to cancel the UR. If you cancel an uncommitted UR, DB2 rolls back the changes. The amount of time that is required for this process depends on the amount of work that was done by the UR.

DSNJ032I

Programmer response: Ensure that the application commits frequently enough, or consult with your DB2 administrator about increasing the threshold number of log records. Use the URLGWTH subsystem parameter.

Related tasks:

➡ Displaying threads by LUWID (DB2 Administration Guide)

➡ Programming for concurrency (DB2 Performance)

Related reference:

➡ UR LOG WRITE CHECK field (URLGWTH subsystem parameter) (DB2 Installation and Migration)

DSNJ032I   *csect-name alert-level*- **APPROACHING END OF THE LOG RBA RANGE OF** *maximum-value*. **CURRENT LOG RBA IS** *current-level*

Explanation: The current log RBA is approaching the end of the log RBA range. The current log RBA should not be allowed to advance to the *maximum-value*.

csect-name
> The name of the control section that issued the message.

alert-level
> The type of alert, based on the current log RBA level:

> **WARNING**
>> The current log RBA reached x'F00000000000' for the 6-byte RBA format or x'FFFFFFFFF000000000000' for the 10-byte RBA format.

> **CRITICAL**
>> The current log RBA value reached x'FFFF00000000' for the 6-byte RBA format or x'FFFFFFFFFFF0000000000' for the 10-byte RBA format.

maximum-value
> The value is *FFFFFFFFFFFF* if the log is being written in the 6-byte RBA format. The value is *FFFFFFFFFFFFFFFFFFFF* if the log is being written in the 10-byte RBA format.

current-level
> The current log RBA value.

System action: DB2 continues processing if the *alert-level* value is WARNING or if DB2 is restarting. Otherwise, DB2 terminates with reason code 00D10251.

Any attempt to restart DB2 in general access mode, following the termination will cause DB2 to restart in light mode, and DB2 will shut down after restart completes the recovery of the incomplete units-of-recovery.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

If DB2 terminates and restarts in light mode, wait until it completes the recovery of incomplete units-of-recovery, and then restart it in ACCESS(MAINT) mode to recover any indoubt units-of-recovery, start any table spaces in a restricted state such as LPL or GRECP, or terminate any utilities that are in progress.

System programmer response: If the DB2 log is using the 6-byte RBA format, convert to the 10-byte RBA format. If the DB2 log is using the 10-byte RBA format, use the instructions for resetting the log RBA to determine when to reset the log and how to reset it, depending on your data sharing environment.

Related concepts:

➡ Restart light (DB2 Data Sharing Planning and Administration)

Related tasks:

➡ Starting a table space or index space that has restrictions (DB2 Administration Guide)

➡ Resetting the log RBA (DB2 Administration Guide)

Related reference:

➡ -TERM UTILITY (DB2) (DB2 Commands)

➡ -START DATABASE (DB2) (DB2 Commands)

➡ DSNJCNVT (DB2 Utilities)

Related information:

➡ 00D10251 (DB2 Codes)

DSNJ033E   *csect-name* **THE LOG RBA MUST BE RESET BEFORE THE END OF THE LOG RBA RANGE IS REACHED**

Explanation: This message is issued with the DSNJ032I message to indicate that DB2 is approaching the end of the log RBA range.

csect-name
> The name of the control section that issued the message.

System action: The *alert-level* value in the accompanying message DSNJ032I determines the system action.

User response: See the description of message DSNJ032I for information and recommended actions.

Related information:
"DSNJ032I" on page 242

DSNJ034I   *csect-name alert-level*- **APPROACHING END OF THE LRSN RANGE. CURRENT LRSN IS** *current-lrsn*

Explanation: The LRSN for the data-sharing group is approaching the maximum value. If the DB2 log is using the 6-byte RBA format, the maximum LRSN is x'FFFFFFFFFFFF'. If the DB2 log is using the 10-byte RBA format, the maximum LRSN is x'FFFFFFFFFFFFFFFFFFFF'.

csect-name
> The name of the control section that issued the message.

alert-level
> The type of alert, based on the current LRSN value:
>
> **WARNING**
>> The LRSN range will reach its limit in one year or less.
>
> **CRITICAL**
>> The LRSN range will reach its limit in one week or less.

current-lrsn
> The current LRSN value.

System action: DB2 continues processing if the *alert-level* value is WARNING or if DB2 is restarting. Otherwise, DB2 terminates with reason code 00D10251.

Any attempt to restart DB2 in general access mode, following the termination, causes DB2 to restart in light mode and to shut down after restart completes the recovery of the incomplete units-of-recovery.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

If DB2 terminates and restarts in light mode, wait until it completes the recovery of incomplete units-of-recovery, and then restart it in ACCESS(MAINT) mode to recover any indoubt units-of-recovery, start any table spaces in a

| restricted state such as LPL or GRECP, or terminate any utilities that are in progress.

| **System programmer response:** If the DB2 subsystem is using the 6-byte RBA log format, convert to the 10-byte RBA
| log format. This format allows the use of 10-byte LRSN values as well. If the database objects are in 6-byte basic
| format, they need to be converted to 10-byte extended format. If the DB2 subsystem is using the 10-byte RBA log
| format, contact IBM Software Support.

| **Related concepts**:

| ⟶ Restart light (DB2 Data Sharing Planning and Administration)

| **Related tasks**:

| ⟶ Starting a table space or index space that has restrictions (DB2 Administration Guide)

| **Related reference**:

| ⟶ -TERM UTILITY (DB2) (DB2 Commands)

| ⟶ -START DATABASE (DB2) (DB2 Commands)

| **Related information**:

| ⟶ 00D10251 (DB2 Codes)

| **DSNJ035E**   *csect-name* **ACTION MUST BE TAKEN BEFORE THE END OF THE LRSN RANGE IS REACHED**

| **Explanation:** This message is issued with the DSNJ034I message to indicate that DB2 is approaching the end of the
| LRSN range.

| *csect-name*
| The name of the control section that issued the message.

| **System action:** The *alert-level* value in the accompanying message DSNJ034I determines the system action.

| **User response:** See the description of message DSNJ034I for information and recommended actions.

| **Related information**:
| "DSNJ034I" on page 243

DSNJ050I   **LOAD MODULE DSNJL003 DOES NOT HAVE RMODE(24) ATTRIBUTE**

Explanation: The named load module was not link edited with the required RMODE(24) attribute.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer of the failure.

System programmer response: Verify that all installation and maintenance activities against this module were
executed using SMP. Verify that the JCLIN for SMP includes the correct RMODE control statement, and that the
Linkage Editor provided with the Data Facility Product (DFP) for the version of MVS being used was invoked by
SMP for the link edits. Correct the procedure or JCL which caused the error and rerun the necessary SMP jobs.

Problem determination: The MVS service aid AMBLIST provides Linkage Editor module attributes in the summary
section of the LISTLOAD function output.

DSNJ070I   *csect-name* **ARCHIVE LOG DSN PREFIX NOT IN PROPER FORMAT TO RECEIVE TIMESTAMP
               DATA. TIMESTAMPING OF** *dsnname* **BYPASSED**

Explanation: By way of a ZPARM parameter, the user has specified that the date and time of creation of an archive
log data set be included as part of the archive log data set name (DSN). To accomplish this, the DB2 subsystem
requires that the length of the archive log data set name prefix not exceed 19 characters if using the standard DATE
format (YYDDD), or 17 characters if using the extended DATE format (YYYYDDD). If the prefix requirement is not
met, this message is issued just prior to the allocation of the archive log data set specified in the message.

System action: The archive log data set will be allocated using the archive log prefix. However, the archive log DSN
will not contain the date and time as the user requested.

Operator response: Notify the system programmer.

System programmer response: The values of TSTAMP and ARCPFXn subsystem parameters are not consistent with one another and must be changed.

Related reference:

▶ TIMESTAMP ARCHIVES field (TSTAMP subsystem parameter) (DB2 Installation and Migration)

▶ COPY 1 PREFIX field (ARCPFX1 subsystem parameter) (DB2 Installation and Migration)

▶ COPY 2 PREFIX field (ARCPFX2 subsystem parameter) (DB2 Installation and Migration)

DSNJ071I   *csect-name* **TIMER FAILURE CAUSED TIMESTAMPING OF ARCHIVE** *dsname* **TO BE BYPASSED**

Explanation: By way of a ZPARM parameter, the user has specified that the date and time of creation of an archive log data set be included as part of the archive log data set name (DSN). However an attempt to get the current date and time from the system failed. This message is issued just prior to the allocation of the archive log data set specified in the message.

System action: The archive log data set will be allocated using the archive log prefix. However, the archive log DSN will not contain the date and time as the user requested.

DSNJ072E   **ARCHIVE LOG DATA SET** *dsname* **HAS BEEN ALLOCATED TO NON-TAPE DEVICE AND CATALOGED. ZPARM CATALOG OPTION OF 'NO' HAS BEEN OVERRIDDEN.**

Explanation: The following archive log subsystem parameters specified that all archive log data sets are to be allocated to a non-tape device and uncataloged:

- The UNIT subsystem parameter specified a non-tape device.
- The CATALOG subsystem parameter value was specified as NO.

However, DB2 requires that all archive log data sets that are allocated to non-tape devices must be cataloged.

dsname   The archive log data set.

System action: The specified archive log data set was allocated to a non-tape device, as specified by the UNIT subsystem parameter, and was cataloged. The specified CATALOG subsystem parameter value of NO was overridden. The BSDS reflects that the data set was cataloged.

Operator response: Notify the system programmer.

System programmer response: Change the values of the UNIT and CATALOG subsystem parameters so that they are consistent with one another

Related reference:

▶ DEVICE TYPE 1 field (UNIT subsystem parameter) (DB2 Installation and Migration)

▶ CATALOG DATA field (CATALOG subsystem parameter) (DB2 Installation and Migration)

DSNJ073I   **LOG ARCHIVE UNIT ALLOCATION FAILURE DETECTED, RETURN CODE=nnnn. ALLOCATION OR OFFLOAD OF ARCHIVE LOG DATA SET MAY FAIL.**

Explanation: While building the SVC99 text entries to dynamically allocate a new archive log data set, a unit allocation failure was detected. This message is only generated when a trace for IFCID 320 is active. The reason code, indicated by 'nnnn' in the message, further clarifies the problem as follows:

4 (X'04')
   The device type (unit name) on which the archive log data set is to be allocated is specified by the user in the DB2 system parameters. No UCBs could be found for the device type (unit name).

8 (X'08')
   The device type (unit name) on which the archive log data set is to be allocated is specified by the user in the DB2 system parameters. This device type (unit name) could not be found in the MVS eligible device table (EDT).

32 (X'20')
   DB2 was able to obtain a list of devices corresponding to the device type (unit name) specified in ZPARMs. However, it was determined by DB2 that this list contained a mixture of tape and nontape devices.

DSNJ096E

40 (X'28')

The device type (unit name) specified by the user in ZPARMs is valid. However, no devices are currently associated with the given device type (unit name).

44 (X'2C')

The device type (unit name) specified by the user in ZPARMs is valid. However, no DASD volumes are available with a volume use attribute of 'storage'.

48 (X'30')

The CVT indicates that UCB services are not available.

System action: This message is issued by DB2 after the SVC99 text entries are built, but prior to the allocation of the new archive log data set. As a result of the error, the dynamic allocation of the archive log data set will be attempted using standard default values. The standard default values are generally acceptable; however, the allocation may fail or the subsequent offload may produce undesirable processing results. For example:

- A return code of 4 or 44 (X'2C') will foretell an allocation error (DSNJ103I) when the SVC99 is issued for the archive data set.

- Offload processing to tape may fail. DB2 uses a volume count of 20 when allocating to tape, and uses the standard MVS volume count default of 5 volumes when writing to nontape devices. In the case of most of the above errors, it would be impossible for the DB2 subsystem to determine the device type on which the data set is to be allocated. Therefore, the standard MVS default is assumed for the volume count. If the data set is successfully allocated to a tape device, and the volume of data is such that more than five volumes will be used for the archive data set, the offload processing will receive an MVS ABEND 837-08 with message IEC028I when attempting to write to the sixth tape volume.

- Offload processing to a direct access device may fail. When allocating a new archive log data set on a direct access device, the DB2 subsystem will use a unit count to facilitate multivolume archive data sets. With most of the above errors, it may be impossible for the DB2 subsystem to correctly determine the type of device on which the data set is to be allocated. Therefore, the standard default (1) is assumed for the unit count. If the data set is successfully allocated to a direct access device, and during the offload processing it becomes necessary to extend the data set to another device, the offload processing will receive an MVS B37 (out of space) ABEND, and the archive log data set will be deallocated.

Operator response: Notify the system programmer.

System programmer response: The required action is based on the return code indicated in the message.

4 (X'04') or 8 (X'08')

The most likely cause is an incorrect specification in the archive log ZPARMs (DSNTIPA panel). Correct the DEVICE TYPE field. If the DEVICE TYPE field from the archive log ZPARMs appears to be correct, check the EDT to ensure that the esoteric or generic unit name specified in the ZPARMs is actually in the EDT. Subsequent offload processing will archive the log data which could not be previously archived due to the allocation failure (DSNJ103I).

32 (X'20') or 40 (X'28')

To correct this situation, change the archive log DEVICE TYPE ZPARM to use a device type (unit name) which contains homogenous devices, or modify the device list associated with the device type (unit name) via a system generation to supply a list of homogenous devices.

44 (X'2C')

To correct this situation, issue an MVS mount command to change the volume use attribute of a mounted 'private' volume to 'storage'. If this is a recurring problem, a system generation may be required to add permanently resident volumes with a volume use attribute of 'storage' to the esoteric or generic unit, or ZPARMs may have to be changed to use a different esoteric or generic unit name for the DEVICE TYPE (on the DSNTIPA installation panel).

48 (X'30')

UCB services became available in MVS/ESA Version 4.1 (HBB4410).

DSNJ096E   *csect-name* **CRCR ENDLRSN RBA VALUE OF** *rrr* **IS INVALID, NO TRUNCATION WILL OCCUR**

Explanation: Truncation cannot take place because the RBA value specified on the active CRCR record is not equal to the RBA of a log record. *rrr* is the ENDLRSN specified in the active record. The ENDLRSN RBA value must match the start of a log record in a non-data sharing environment.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Confirm that the ENDLRSN value specified in the Conditional Restart control record is correct. Run the Change Log Inventory (DSNJU003) utility with the CRESTART CANCEL parameters to cancel the conditional restart. If log truncation is needed, run the DSNJU003 utility with the correct ENDLRSN value. Restart DB2.

DSNJ098E   *csect-name* **CRCR ENDLRSN OF** *rrr* **IS** *xxx* **THAN ANY KNOWN** *yyyy*, **NO TRUNCATION WILL OCCUR**

Explanation: Truncation cannot take place because the CRCR ENDLRSN is not in the range of LRSN or RBA values known to either the active or archive logs. *rrr* is the end LRSN specified in the active record. *xxx* is replaced by either HIGHER or LOWER. In a data sharing environment, if *xxx* is HIGHER, the CRCR ENDLRSN is higher than the LRSN of the last record written to the log data sets. If *xxx* is LOWER, the CRCR ENDLRSN is lower than the starting LRSN of the oldest archive data set. *yyyy* is replaced by LRSN. In a non-data sharing environment, if *xxx* is HIGHER, the CRCR ENDLRSN is higher than the RBA of the last record written to the log data sets. If *xxx* is LOWER, the CRCR ENDLRSN value is an RBA value that is lower than the starting LRSN of the oldest archive data set. *yyyy* is replaced by RBA.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Confirm that the ENDLRSN value specified in the Conditional Restart control record is correct. If it is not correct, run the Change Log Inventory (DSNJU003) utility with the CRESTART CANCEL parameters to cancel the conditional restart. If log truncation is needed, run the DSNJU003 utility with the correct ENDLRSN. Restart DB2.

DSNJ099I   **LOG RECORDING TO COMMENCE WITH STARTRBA=...**

Explanation: This message is generated when the log manager is initialized during DB2 startup. The value specified by the STARTRBA=value keyword is the RBA of the next byte of log data to be recorded in the DB2 active log data set(s). This information may later be needed for a conditional restart.

This message is preceded by one DSNJ001I message (if single logging) or two DSNJ001I messages (if dual logging).

System programmer response: None required. However, if recovery is required, information from this message may be required as input to the change log inventory (DSNJU003) utility.

DSNJ100I   *csect-name* **ERROR OPENING BSDSn DSNAME=..., ERROR STATUS=***eeii*

Explanation: Log manager initialization or the -RECOVER BSDS command could not open the specified bootstrap data set (BSDS). BSDSn matches the ddname in the log startup JCL of the data set that cannot be opened. The value of *n* is 1 or 2. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from BSDS or log failures.

System action: When this error occurs at initialization time, DB2 startup must be terminated, because the log data sets cannot be determined and allocated without the BSDS. When this error occurs during -RECOVER BSDS processing, the command is terminated, and the system continues in single BSDS mode.

Operator response: Notify the system programmer.

System programmer response: Recover the BSDS that cannot be opened. Normally, if dual BSDSs are being used, recovery consists of making a copy of the remaining good BSDS.

Problem determination: The 'error status' contains the VSAM open return code in 'ee', and the VSAM open reason code in 'ii'. Refer to *z/OS DFSMS: Macro Instructions for Data Sets* for a list of the VSAM open return codes and reason codes, and the steps required to take corrective action.

DSNJ101E   *csect-name* **CRCRERBA OF** *rrr* **IS** *xxx* **THAN ANY KNOWN RBA, BUT COLD START WAS NOT SPECIFIED**

Explanation: Truncation cannot take place because the CRCR end RBA is not in the range of RBA values known to either the active or archive logs. 'rrr' is the end RBA specified in the active record. 'xxx' is replaced by either HIGHER or LOWER. If HIGHER is the case, the CRCRERBA is higher than the end RBA of the most recent active log data set. LOWER means the CRCRERBA is lower than the starting RBA of the oldest archive data set.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Most likely, the archive log data set that contained the requested RBA has been deleted from the bootstrap data set (BSDS) by the change log inventory (DSNJU003) utility. Locate the output from an old Print Map run, and identify the data set that contains the missing RBA. If the data set has not been reused, run the DSNJU003 utility to add this data set back into the inventory of log data sets. Restart DB2.

DSNJ102I   **LOG RBA CONTENT OF LOG DATA SET DSNAME=..., STARTRBA=..., ENDRBA=..., DOES NOT AGREE WITH BSDS INFORMATION**

Explanation: The log RBA range shown in the BSDS for the specified data set does not agree with the content of the data set. This message indicates that the change log inventory (DSNJU003) utility was used incorrectly or that a down-level volume is being used.

System action: DB2 startup processing is terminated.

Operator response: Notify the system programmer.

System programmer response: Use the print log map (DSNJU004) and change log inventory (DSNJU003) utilities to make the BSDS consistent with the log data sets.

DSNJ103I   *csect-name* **LOG ALLOCATION ERROR DSNAME=***dataset-name*, **ERROR STATUS=***status-indicator*, **SMS REASON CODE=***ssssssss*

Explanation: The log manager encountered an error while attempting to allocate the active or archive log data set indicated by the *dataset-name* value. The S*status-indicator* value indicates the error reason code returned by MVS dynamic allocation (SVC99).

This message might be preceded by message DSNJ073I.

System action: Subsequent log manager actions are based upon the type of data set involved.

For active log data sets, if the error is encountered during log manager initialization, DB2 startup is terminated. If two copies of the active log data sets are defined, this message appears only once.

For archive log data sets, if two copies of the archive log data sets are defined, processing continues on the remaining archive log data set.

Operator response: Notify the system programmer.

System programmer response: The error status portion of this message contains a 2-byte error code (S99ERROR) followed by the 2-byte information code (S99INFO) from the SVC99 request block. If the S99ERROR code indicates an SMS allocation error (97XX), then 'ssssssss' contains additional SMS reason code information obtained from S99ERSN.

For active log data sets, if the problem occurred during DB2 initialization, you can resolve the problem by either:
- Resolving the error associated with the active log data set as indicated by the *status-indicator* value.
- Using AMS, providing another copy of the active log data set
- Updating the BSDS with the change log inventory (DSNJU003) utility
- Restarting DB2.

For archive log data sets:
- If the problem occurred during allocation with the intent to write the data set, no immediate action is required. However, failure to resolve the SVC99 error (indicated by the STATUS value in the message) might eventually cause the available space in the active log to be exhausted (DSNJ111E), since all future offloads might fail due to the same error.

- If the problem occurred during allocation with the intent to read the data set, determine the problem, and use the DSNJU003 utility DELETE function to delete the archive log data set from the BSDS archive log inventory. Then use the NEWLOG function to add the data set back into the archive log inventory, pointing to the correct volume and device.

This message can also be issued as the result of a possible user error. If the *status-indicator* value is'17080000', then the site might have one or more active log data sets defined in the BSDS, but not allocated on DASD. To correct the situation, print the contents of the current active log data set inventory using the print log map (DSNJU004) utility, and then take one of the following actions:

- For each active log data set listed in the BSDS, but not actually allocated on DASD, allocate the active log data set using access method services. The access method services command syntax for active log data sets can be found in the DSNTIJIN installation job stream.
- Use the DSNJU003 utility DELETE statement to delete the errant active log data set name, and the NEWLOG statement to add the correct name to the active log data set inventory. The name specified on the NEWLOG statement must be the same as the name of the actual DB2 active log data set allocated on DASD.

Related concepts:

➦ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

Related reference:

➦ DSNJU003 (change log inventory) (DB2 Utilities)

Related information:

➦ Recovering from active log failures (DB2 Administration Guide)

➦ Recovering from archive log failures (DB2 Administration Guide)

"DSNJ073I" on page 245

"DSNJ111E" on page 252

DSNJ104I   *csect-name* **RECEIVED ERROR STATUS** *nnn* **FROM** *macro-name* **FOR DSNAME** *dsname*

Explanation: DB2 encountered an error while issuing macro 'macro-name'. Error status is the return code from the specified macro.

- For an OPEN of a VSAM data set, the return code in the error field of the access method services control block is included in this message as the error status value. Refer to the appropriate VSAM publication for a definition of these values.
- If the OPEN was for a non-VSAM data set, the error status is zero.
- For MMSRV errors, error status contains the error information returned by media manager services. If an MMSRV CATUPDT failure occurs attempting to truncate an active log data set, the log data set will be unavailable to DB2 and the status of the log data set will be flagged as 'STOPPED' in the BSDS.
- For VSAM OPEN and MMSRV errors, this message is preceded by an IEC161I message that defines the error that occurred.
- For a PROTECT of an archive log data set, the return code is from DADSM PROTECT. Refer to the appropriate MVS publication for a description of the MMSRV return codes.
- For a CLOSE of an archive log data set, the return code is the TCB completion code for an abend received during CLOSE processing. This abend is normally preceded by an IEC210I or IEC212I message that defines the error.
- For EDTINFO and UCBLOOK errors, the first 2 bytes contain the return code and the second 2 bytes contain the reason code.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from log failures.

System action: If this condition is encountered during log manager initialization, DB2 startup is terminated. If the error occurs later and the data set is needed either for offload or for input operations, DB2 processing continues. If a second copy of the data is available, DB2 attempts to allocate and open the second data set.

If the data set is needed as an active log data set, the log manager attempts to retry the failing request. If the retry fails, the DB2 subsystem is terminated.

DSNJ105I • DSNJ106I

Operator response: Notify the system programmer.

System programmer response: If the error occurred during initialization, the installation must either correct the problem so that the data set is available or provide another copy of the data set and change the BSDSs to point to the new data set.

If the error occurred during DB2 processing, the return code should be reviewed and the appropriate action taken to correct the problem so that the data set can be used at a later time or the data set entry can be removed from the BSDS via the change log inventory (DSNJU003) utility.

If the error was received from PROTECT, there may be a problem with the PASSWORD data set. You should refer to the appropriate DADSM publication to determine the cause of the problem. When the problem has been corrected, you should ensure the archive log data sets receiving the error are added to the PASSWORD data set. If these archive log data sets are not added to the PASSWORD data set, archive read will fail when attempting to OPEN these data sets.

If the error was received from EDTINFO or UCBLOOK and the allocation for the archive log data set fails, verify that the archive UNIT names are correct in ZPARMs.

DSNJ105I   *csect-name* **LOG WRITE ERROR DSNAME=..., LOGRBA=..., ERROR STATUS=***ccccffss*

Explanation: The log manager encountered a write error in the specified active log data set. If 'csect-name' is DSNJW107 or DSNJW127, the error occurred writing the log buffers to an active log data set. If 'csect-name' is DSNWJ207, the error occurred while preformatting the next control area before writing log data into it.

Error status contains the error information returned by media manager in the form 'ccccffss', where 'cccc' is a 2-byte return code that describes the error, 'ff' is a 1-byte code that defines the functional routine that detected the error, and 'ss' is the 1-byte status code that defines a general category of error. Refer to the appropriate MVS publication for a description of the MMCALL return codes.

If the dual active logging option is selected, the log manager switches to the next data set for this copy. If the next data set is not ready, the log manager temporarily enters single logging mode and allocates a replacement data set for the one that encountered the error. Dual logging is resumed as soon as possible.

If single active logging option is selected and the next data set is not ready, the log manager waits for that data set to be available. In this case, log writing is inhibited until the replacement is ready for output.

System action: See the Explanation section of this message.

Operator response: Notify the system programmer.

System programmer response: See the Explanation section of this message.

DSNJ106I   **LOG READ ERROR DSNAME=***dsname-value***, LOGRBA=***log-rba-value***, ERROR STATUS=***ccccffss*

Explanation: The log manager encountered an error while reading an active log data set. The ERROR STATUS value contains the error information returned by the media manager in the form *ccccffss*, where *cccc* is a 2-byte return code that describes the error, *ff* is a 1-byte code that defines the functional routine that detected the error, and *ss* is the 1-byte status code that defines a general category of error.

System action: If another log data set contains the data, the log manager attempts to read the data from the alternate source. If an alternate source is not available, the log manager returns a read error return code to the program requesting the log data.

Operator response: Notify the system programmer.

System programmer response: If the site is using dual logging, the requested RBA was probably retrieved from the corresponding dual active log data set, and no immediate response is necessary. However, if this error occurs frequently, or if the site is using single logging, immediate attention may be required.

Take corrective action based on the media manager return codes.

It might be necessary to replace the bad data set with a new data set containing the log data, and to update the BSDSs to reflect the new data set using the change log inventory (DSNJU003) NEWLOG operation.

This message can also be issued as the result of a possible user error. If the data set name associated with the DSNAME value is missing, and the ERROR STATUS value contains '00180408' or '00100408', then the site is using

dual logging, but only one set of active log data sets are defined in the BSDS. To resolve this condition, take one of the following actions:

- Define a second set of active log data sets using access method services (if they are not defined already), and update the BSDS log inventory using the DSNJU003 NEWLOG operation.
- To specify single logging, set the value of the TWOACTV subsystem parameter to NO.

Related concepts:

➡ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

Related reference:

➡ z/OS DFSMSdfp Diagnosis

➡ NUMBER OF COPIES field (TWOACTV subsystem parameter) (DB2 Installation and Migration)

➡ DSNJU003 (change log inventory) (DB2 Utilities)

Related information:

➡ Recovering from active log failures (DB2 Administration Guide)

➡ Recovering from archive log failures (DB2 Administration Guide)

DSNJ107I   READ ERROR ON BSDS DSNAME=_name_ **ERROR STATUS=**_status_

Explanation: The log manager encountered an error while reading the specified bootstrap data set. Error Status contains the VSAM return and feedback codes. It is a 2-byte field with the first byte containing the hexadecimal return code and the second containing the hexadecimal feedback code. Refer to the appropriate MVS publication for a description of VSAM return and reason codes.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from BSDS or log failures.

System action: If dual BSDSs are available, the log manager attempts to read from the other BSDS. If read from the second BSDS fails or if there is only one BSDS, an error code is returned to the log request that caused access to the BSDS.

If read error is detected during start-up, DB2 initialization process is terminated to prevent users from operating in a cripple mode. User should take action to correct the problem.

Operator response: Notify the system programmer.

System programmer response: It may be necessary to replace or repair the BSDS, depending on what conditions resulted from the read error. To replace a BSDS, first delete the bad BSDS, and then define the new BSDS with the same name and attributes. If a new name is used for the new BSDS, change the log startup JCL to specify the new BSDS name.

DSNJ108I   WRITE ERROR ON BSDS DSNAME=... ERROR STATUS=...

Explanation: The log manager encountered an error while writing to the specified bootstrap data set. Error Status contains the VSAM return and feedback codes. It is a 2-byte field with the first containing the hexadecimal return code and the second containing the hexadecimal feedback code. Refer to the appropriate VSAM publication for a description of VSAM return and reason codes.

System action: If dual BSDSs are available, the log manager enters single BSDS mode using the remaining good BSDS. Otherwise, an error code is returned to the log request that caused access to the BSDS.

Operator response: Notify the system programmer.

System programmer response: If dual BSDS mode is being used, execute an offline access method services job to rename the error BSDS and define a new BSDS with the same name. Then enter the log manager -RECOVER BSDS command to reestablish dual BSDS mode.

If dual BSDS mode is not being used, the DB2 subsystem must be shut down, and the BSDS must be recovered from a backup copy. To recover the BSDS, use the change log inventory (DSNJU003) utility.

DSNJ109I   OUT OF SPACE IN BSDS DSNAME=...

Explanation: The specified bootstrap data set does not have any more space. The operation that encountered the out-of-space condition did not complete properly.

System action: If dual BSDSs are available, the log manager enters single BSDS mode using the remaining good BSDS. Otherwise, an error code is returned to the log request that caused access to the BSDS.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

System programmer response: If dual BSDS mode is being used, execute an offline access method services job to rename the full BSDS and define a new, larger BSDS with the same name. Enter the DB2 -RECOVER BSDS command to reestablish dual BSDS mode.

If dual BSDS mode is not being used, the DB2 subsystem must be shut down and the BSDS recovered offline. In this case, execute the same access method services job mentioned above to rename the full data set and define a larger data set. Next, execute an access method services REPRO job to copy the full BSDS into the new BSDS.

DSNJ110E   LAST COPYn ACTIVE LOG DATA SET IS nnn PERCENT FULL

Explanation: This message is issued when the last available active log data set is 5% full, and is reissued after each additional 5% of the data set space is filled.

System action: Each time the message is issued, the offload processing is retried. If the situation is not corrected, the active log data space fills to capacity, message DSNJ111E is issued, and the DB2 online processing halts.

Operator response: To clear this condition, steps must be taken to complete other waiting offload tasks. Once an active log data set is made available (reusable) by completing the offload process for it, the DB2 logging activity can continue.

Execute an MVS *Display System Requests* command (D R,L for example) to determine the outstanding requests related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue. If offload fails to complete normally or cannot be initiated, notify the system programmer.

System programmer response: Either correct the problem that is causing the offload process to fail, or add more active log data sets. The latter action requires DB2 to be down and the change log inventory (DSNJU003) utility to be run.

Possible causes for the shortage of active log data space are:

- Excessive logging. For example, too many users have requested logs at the same time or too many utilities have specified LOG(YES).
- Delayed or slow offloading. For example, failure to mount archive volumes, incorrect replies to offload messages, or slow device speeds.
- Excessive use of the -ARCHIVE LOG command. Each invocation of the -ARCHIVE LOG command causes the DB2 subsystem to switch to a new active log data set and to initiate an offload of the active log. Although the -ARCHIVE LOG command will not be processed when only one active log data set remains in a copy of the active log (see DSNJ319I), excessive use of the command could have consumed all space in the active log except the current active log data sets.
- Offloads failed. See Operator Response.
- Insufficient active log space. See Operator Response.
- Offload task no longer functioning. Issue the -ARCHIVE LOG CANCEL OFFLOAD command to abnormally terminate and restart the offload process.

DSNJ111E   OUT OF SPACE IN ACTIVE LOG DATA SETS

Explanation: Due to delays in offload processing, all available space in all active log data sets has been exhausted. DB2 recovery logging cannot continue.

System action: The DB2 log manager waits for an available data set.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Execute an MVS Display System Requests command (D R,L for example) to ensure that there

are no outstanding requests that are related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue.

System programmer response: Examine the JES job log for the *ssnm*MSTR address space to determine the cause of the delay. If the delay was caused by the lack of a resource required for offload, the necessary resource must be made available to allow offload to complete and thus permit logging to proceed.

Related concepts:

Effects of interruptions and errors on the offload process (DB2 Administration Guide)

Related information:

Recovering from temporary resource failure (DB2 Administration Guide)

Recovering from active log failures (DB2 Administration Guide)

Recovering from being out of space in active logs (DB2 Administration Guide)

DSNJ112E   *csect-name* **INSUFFICIENT ACTIVE LOG DATA SETS DEFINED IN BSDS**

Explanation: There are not enough active log data sets defined in the BSDS to start DB2. This condition usually exists for one of the following reasons.

- Fewer than two data sets are defined for one of the active log copy sets.
- DSN6LOGP specified TWOACTV=YES in DB2 initialization parameters, but data sets for two copies of active log are not defined in BSDS.
- Fewer than two data sets are available (not flagged as STOPPED) for one of the active log copy sets.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Use the change log inventory (DSNJU003) utility to make the number of active log data sets defined in the BSDS consistent with the parameters specified on DSN6LOGP. Log data sets that are flagged as 'STOPPED' will not be reused by DB2. These data sets must be recovered (either redefined or recovered from the other copy of the log) and the log data sets must be deleted and re-added to the BSDS (using DSNJU003) with the appropriate RBA's, to clear the STOPPED status. Restart DB2.

DSNJ113E   *csect-name log-type log-value* **NOT IN ANY ACTIVE OR ARCHIVE LOG DATA SET.**
CONNECTION-ID=*aaaaaaaa*, **CORRELATION-ID=***aaaaaaaa*, **MEMBER-ID=***member-id*

Explanation: A request was made to read the log starting at this *log-value* and *log-type* (RBA or LRSN). However, the log value cannot be found in any active or archive log data set for the indicated member. The *member-id* contains the ID of the data sharing member whose logs were being requested at the time of the failure or zero if data sharing is not active. The connection and correlation IDs identify the user or application that encountered the problem. Correlation IDs beginning with '0nn' where 'nn' is a number from 01 to 28, identify system agents. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from log failures.

System action: Depending upon what log record is being read and why, this error could result in an abend with a reason code of '00D1032A'.

Operator response: Notify the system programmer.

System programmer response: This error could be a user error. In most cases, the archive log data set that contained the requested log value has been deleted from the bootstrap data set (BSDS) by the change log inventory (DSNJU003) utility. Complete the following steps to correct the error:

- Locate the output from an old print log map run of the BSDS for the DB2 member identified by *member-id*.
- Identify the data set that contains the missing log value. If the data set has not been reused, run the DSNJU003 utility to add this data set back into the inventory of log data sets.
- Restart DB2.

DSNJ114I   ERROR ON ARCHIVE DATA SET, OFFLOAD CONTINUING WITH ONLY ONE ARCHIVE DATA SET BEING GENERATED

Explanation: An error occurred while accessing one of the archive data sets being created by offload. Because the dual archive option is specified, offload is continuing with the other archive data set. For the RBA range being offloaded, there is only one copy of the archive data set instead of the usual two copies.

System action: Offload produces a single archive data set.

System programmer response: A second copy of this archive log data set can be made, and the BSDS can be updated with the change log inventory utility (DSNJU003).

DSNJ115I   OFFLOAD FAILED, COULD NOT ALLOCATE AN ARCHIVE DATA SET

Explanation: Offload could not allocate or open an archive log data set. The offload was not performed. If this message is preceded by message DSNJ103I, the offload data set could not be allocated. If this message is not preceded by DSNJ103I, the offload data set was successfully allocated, but could not be opened.

System action: Offload will be tried at a later time.

Operator response: Notify the system programmer.

System programmer response: If the allocation failed with DSNJ103I, review the error status information of message DSNJ103I. Correct the condition that caused the data set allocation to fail so that, on retry, the offload can take place.

DSNJ116I   ERROR ATTEMPTING TO ADD ARCHIVE ENTRY TO BSDS

Explanation: Offload could not add an archive entry to the BSDS. The offload is considered incomplete. The active log data set is not marked as reusable for new log data. This message is preceded by message DSNJ107I, DSNJ108I, or DSNJ109I.

System action: Offload will be retried at a later time.

Operator response: Notify the system programmer.

System programmer response: Refer to the specific preceding message for action.

DSNJ117I   INITIALIZATION ERROR READING BSDS DSNAME=......, ERROR STATUS=...

Explanation: The log manager encountered an error during initialization reading from the specified bootstrap data set. Error status contains the VSAM return and feedback codes. It is a two byte field with the first containing the hexadecimal return code and the second byte containing the hexadecimal feedback code. Refer to the appropriate VSAM publication for a description of VSAM return and reason codes.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the read error using the VSAM error status information provided. Restart DB2.

DSNJ118I   LOG INITIALIZATION RECEIVED RETURN CODE nnn FROM xxxxxxxx MACRO

Explanation: Log manager initialization received a bad return code from the named macro.

System action: DB2 initialization is terminated.

Operator response: Notify the system programmer.

System programmer response: Determine the problem from the documentation on the named macro and return code. Then take appropriate steps, and restart the DB2 subsystem.

DSNJ119I   **BOOTSTRAP ACCESS INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the bootstrap data set access function was unable to complete its initialization process. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for information about recovery from BSDS or log failures.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error have preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ120I   **DUAL BSDS DATA SETS HAVE UNEQUAL TIMESTAMPS, BSDS1 SYSTEM=**_timestamp_**, UTILITY=**_timestamp_**, BSDS2 SYSTEM=**_timestamp_**, UTILITY=**_timestamp_

Explanation: When the log manager was initialized, the timestamps of the dual bootstrap data sets (BSDS) did not agree. The timestamps from the system and from the change log inventory (DSNJU003) utility are shown for each BSDS. The timestamps have the format _yy.ddd hh:mm:ss.th_:

_yy_   Two-digit indicator for the year.

ddd   Three-digit indicator for the day of the year.

_hh:mm:ss.th_
             Indicator for the time of the day.

System action: DB2 attempts to re-synchronize the BSDS data sets to restore dual BSDS mode. If re-synchronization is successful, DB2 restart continues.

Operator response: If restart fails, notify the system programmer.

System programmer response: If restart fails, run the print log map (DSNJU004) utility against each BSDS. From the output, determine which data set is obsolete, delete it, define a replacement for it, and copy the remaining BSDS to the replacement.

If output from the print log map (DSNJU004) utility for both data sets is similar, delete the data set with the oldest timestamp, and copy the data set with the most recent timestamp.

Related reference:

� DSNJU003 (change log inventory) (DB2 Utilities)

� DSNJU004 (print log map) (DB2 Utilities)

DSNJ121I   **LOG MANAGER ENCOUNTERED AN ERROR READING THE JFCB. DDNAME=**_nnnnnnnn_

Explanation: During log manager initialization (if dual BSDS data sets are specified), the job file control block (JFCB) in MVS is read to obtain the data set names associated with DDNAME BSDS1 and BSDS2. This error is caused by a missing DD statement.

System action: DB2 is terminated.

Operator response: Notify the system programmer.

System programmer response: Ensure that a DD statement exists in the DB2 startup procedure for DDNAME BSDS1. If dual BSDS data sets are used, ensure that a DD statement also exists in the DB2 startup procedure for DDNAME BSDS2.

DSNJ122I   **DUAL BSDS DATA SETS ARE OUT OF SYNCHRONIZATION**

Explanation: During log manager initialization, the dual BSDS data sets were found to differ in content.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Run the print log map (DSNJU004) utility against each BSDS to determine which

data set was last used as copy 1. Delete the copy 2 data set, define a replacement for the deleted data set, and copy the remaining BSDS to the replacement.

DSNJ123I   **CHANGE LOG INVENTORY FAILURE DETECTED**

Explanation: During log manager initialization, the BSDS data set was found to have been incompletely processed by the change log inventory (DSNJU003) utility.

System action: DB2 startup is terminated.

System programmer response: Run the print log map (DSNJU004) utility to determine what operation against the BSDS did not complete. Run the change log inventory (DSNJU003) utility against the BSDS(s) to allow any unfinished processing to be completed.

DSNJ124I   **OFFLOAD OF ACTIVE LOG SUSPENDED FROM RBA *xxxxxx* TO RBA *xxxxxx* DUE TO I/O ERROR**

Explanation: During offload, an unrecoverable I/O error was encountered on an active log data set. The data set experiencing the error is marked unusable, and no further logging is done to that data set.

System action: Active log data sets continue to be offloaded as they become full.

Operator response: Notify the system programmer to initiate recovery actions.

System programmer response: Complete the following steps:
1. Recover the data manually from the data set.
2. Copy it to an archive data set.
3. Run the change log inventory (DSNJU003) utility to make the new archive data set available to the DB2 subsystem.
4. Remove the error-prone active log data set from DB2 by using the change log inventory utility.

DSNJ125I   **ERROR DUMPING BSDS, OFFLOAD CONTINUING WITHOUT PRODUCING THE BSDS DUMP**

Explanation: An error occurred while dumping the bootstrap data set (BSDS) during the offload process. The data set is not produced, but the offload process continues. The volume containing the offloaded data set does not contain a BSDS for recovery use.

System action: The system continues the offload process without producing a dump of the bootstrap data set.

DSNJ126I   **BSDS ERROR FORCED SINGLE BSDS MODE**

Explanation: The log manager encountered an I/O error or a VSAM logical error on a bootstrap data set. This message is preceded by message DSNJ107I or DSNJ108I.

System action: The log manager enters single BSDS mode using the remaining BSDS.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

System programmer response: Execute an offline access method services job to rename the error BSDS and define a new BSDS with the same name. Then enter the log manager -RECOVER BSDS command to reestablish dual BSDS mode.

DSNJ127I   **SYSTEM TIMESTAMP FOR BSDS=*timestamp***

Explanation: When the log manager is initialized, the system timestamp for the bootstrap data sets (BSDS) is displayed. The timestamp has the format *yy.ddd hh:mm:ss.th*:

yy   Two-digit indicator for the year.

ddd   Three-digit indicator for the day of the year.

hh:mm:ss.th
            Indicator for the time of the day.

This time stamp should be close to the last time at which this subsystem was stopped. If it not, then it might indicate that a restart is being attempted with the wrong BSDS.

System action: DB2 startup continues.

Operator response: If the time displayed is not close to the time when this subsystem was last stopped, contact the system programmer.

System programmer response: If you cannot explain the apparent time discrepancy, cancel the subsystem. From the startup procedure for the subsystem services address space, determine the data set names of the bootstrap data sets and run the print log map (DSNJU004) utility. Check whether the active and archive log data sets all belong to this subsystem. If not, then change the startup procedure for the subsystem services address space to use the correct bootstrap data sets.

Related reference:

➡ DSNJU004 (print log map) (DB2 Utilities)

DSNJ128I   **LOG OFFLOAD TASK FAILED FOR ACTIVE LOG** *nnnnn*

Explanation: The offload task abended while attempting to offload the relative byte address (RBA) range in active log data set 'nnnnn'.

System action: The offload task terminates and the archive data sets allocated to the offload task are deallocated and deleted. The status of the active log data sets involved in the failed offload processing remains set to 'not reusable'.

The log offload task will be re-initiated by one of several events. The most common are:
- All the available space in the current active log data set has been used (normal case).
- A DSNJ110E message is issued.
- The master address space is started, but data in the active log has not been archived.
- An I/O error occurs on the active log, which will force the system to truncate and offload the active log data set, and switch to a new active log data set.

Operator response: This message is the result of an offload failure, and will be preceded by one or more DB2 messages (for example, DSNJ073I) and/or MVS ABEND messages (for example, IEC030I, IEC031I, IEC032I). If DB2 is operating with restricted active log resources (see message DSNJ110E), quiesce the system to restrict logging activity until the ABEND or the DSNJ110E condition can be resolved. Notify the system programmer.

System programmer response: This message is the result of an offload failure, and will be preceded by one or more DB2 messages (for example, DSNJ073I) and/or MVS ABEND messages (for example, IEC030I, IEC031I IEC032I). Investigate and correct the cause of the ABEND before the offload is attempted again by DB2.

Problem determination: This message is the result of an offload ABEND and will be preceded by one or more DB2 messages and/or MVS ABEND messages. See the documentation for the associated DB2 and/or MVS messages to formulate a course of corrective action. Use the print log map (DSNJU004) utility to print the BSDS (both copies if running in dual mode), and then use the DSNJU004 output to determine the current status of the active and archive log data sets.

This message can be generated for a variety of reasons. However, the most likely are:
- Archive log data set allocation errors. Refer to the text for message DSNJ103I for corrective action.
- The size of the archive log data set is too small to contain the active log data set(s) during offload processing. All secondary space allocations have been used. This condition is normally accompanied by MVS ABEND message IEC030I.
- All available space on the DASD volume(s) to which the archive data set is being written has been exhausted. This condition is normally accompanied by MVS ABEND message IEC032I.
- The primary space allocation for the archive log data set (as specified in ZPARMs) is too large to allocate to any available online DASD device. This condition is normally accompanied by MVS ABEND message IEC032I.

DSNJ129I END OF *log-rba eol-rba* X COULD NOT BE FOUND IN ANY ACTIVE LOG DATA SET HIGHEST RBA FOUND WAS *hi-rba* X

Explanation: The log manager has been requested to find 'eol-rba', the log record which has been recorded in the bootstrap data set (BSDS) as the highest RBA written. This RBA cannot be found in any active log data set. The highest RBA which could be found in any active data set was 'hi-rba'.

System action: DB2 startup processing is terminated.

Operator response: Notify the system programmer.

System programmer response: Most likely, the active log data set containing the requested RBA has been deleted from the BSDS by the change log inventory (DSNJU003) utility. If the data set has not been reused, run DSNJU003 to add this data set back into the BSDS. Restart DB2.

If the data set is not available, a conditional restart must be performed to either truncate the log prior to the RBA value of the missing data set, or to cold start at a higher RBA value. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for more information.

DSNJ130I ICF CATALOG NAME IN BSDS DOES NOT AGREE WITH DSNZPARM. BSDS CATALOG NAME=*name*, DSNZPARM CATALOG NAME=*name*

Explanation: The ICF catalog name defined in DSNZPARM does not agree with the name contained in the bootstrap data set (BSDS).

System action: DB2 startup is abended.

Operator response: If you attempt to start the DB2 subsystem, you will receive this same message and abnormal termination will result until the problem is resolved.

System programmer response: Change the ICF catalog name in the BSDS to agree with the name defined in DSNZPARM during DB2 initialization. To change the ICF catalog name in the BSDS, use the BSDS change log inventory (DSNJU003) utility with the NEWCAT control statement.

User response: Notify the system programmer.

Problem determination: During log manager initialization, the ICF catalog name defined in DSNZPARM is compared to the catalog name in the BSDS to ensure that the catalog name used by the log manager during DB2 processing is correct. These names were found not to be the same.

DSNJ131I *csect-name* DUAL BSDS MODE RESTORED FROM BSDS*n*

Explanation: Dual BSDS mode has been restored using BSDS copy *n*. This copy is the BSDS data set with the most recent system timestamp.

System action: DB2 startup continues.

DSNJ132I *csect-name* ARCHIVE LOG READ ERROR DSNAME=*data-set-name*, *diag-information*

Explanation: An I/O error occurred attempting to read from archive log data set *data-set-name*. The value for *diag-information* contains a portion of the message text generated by the SYNADAF macro to describe the error. The generated message text contains information such as jobname, stepname, unit address, device type, ddname, operation attempted, and error description.

System action: DB2 will attempt the read request on a secondary copy of the archive log, if it is available. If a secondary copy of the archive does not exist, or this error occurs on the secondary copy of the archive log, the log read request fails with reason code 00D10329.

Operator response: Notify the system programmer.

System programmer response: If the archive log read request is satisfied by a secondary archive log data set, no immediate response is required. Otherwise, attempt to determine the cause of the I/O error and recover the archive data set as required. For example, if the archive log data set has been reblocked by moving or copying the data set, or by an HSM recall, it could result in a 'wrong length record' error. Use IEBGENER, or an equivalent program, to restore the data set with the original DCB information.

DSNJ138I   *csect-name* **OFFLOAD COMPLETE FOR LOG TRUNCATION FROM AN ARCHIVE LOG COMMAND, ENDRBA=***xxxxxxxxxxxx*

Explanation: This message indicates that the offload of the truncated active logs, resulting from an -ARCHIVE LOG command, has completed successfully. *xxxxxxxxxxxx* is the ENDRBA of the truncated active logs and the newly created archive log data sets.

DSNJ139I   LOG OFFLOAD TASK ENDED

Explanation: Processing of the active log offload ended.

System action: This message is written to the MVS console.

Operator response: This message does not guarantee that the offload completed without errors. Check the console log and task messages to review the execution of the offload task to determine whether any abnormal events occurred during the offload.

DSNJ140I   *csect-name* **INVALID BSDS RELEASE LEVEL ACTIVE=***xxxxxx* **ARCHIVE=***yyyyyy* **DSN=***zzzzzz*

Explanation: At initialization time DB2 detected that the BSDS log records are not at the proper release level for this release of DB2. 'xxxxxx' contains the release level for the ACTIVE log records and 'yyyyyy' contains the release level for the ARCHIVE log records. 'zzzzzz' contains the data set name for the BSDS encountering the error.

System action: DB2 startup will be terminated.

Operator response: Notify the system programmer so operating procedures can be changed.

System programmer response: Run the appropriate conversion program (DSNJMIG0 or DSNJFAL0) to convert the BSDS data set to a compatible release level.

DSNJ150E   LOG CAPTURE EXIT ABEND, EXIT DEACTIVATED

Explanation: An abnormal program interrupt was detected while executing in the installation-supplied log capture exit code (i.e. entry point DSNJW117 in load module DSNJL004). As a result of the abend, the log capture exit will no longer be active; log data will no longer be available for Exit capture/processing.

This message can only occur when an installation-supplied log capture exit (entry DSNJW117) is active for this instance of DB2.

System action: The log capture exit (entry point DSNJW117) is terminated. No further calls will be attempted for this DB2 instance. A full dump is provided for diagnostic purposes.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the DSNJL004 load module (DSNJW117 entry point) abend and take corrective action.

Attention: A correctly-functioning copy of load module DSNJL004/entry DSNJW117 <u>must</u> be available in order to start DB2. If the problem which caused the above abend cannot be corrected, ensure that the default DSNJW117 entry (load module DSNJL004 - supplied with the DB2 release) is available during the next DB2 subsystem start.

User response: See "System Programmer Response" below.

DSNJ151I   *csect-name* **IN READING** *log-type log-value*, **A NONZERO REASON CODE WAS RETURNED. CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **REASON-CODE=***reason-code*, **MEMBER-ID=***member-id*

Explanation: DB2 could not successfully complete the read of this *log-value* and *log-type* (RBA or LRSN) due to reason code 'ccc'. The *member-id* contains the ID of the data sharing member whose logs were being requested at the time of the failure or zero if data sharing is not active. The user or application that encountered the error is identified by the connection and correlation ID. Messages that have the same connection ID and correlation ID relate to the same application. Correlation IDs beginning with '0*nn*', where *nn* is a number from 01 to 28, identify system agents.

System action: DB2 attempts to recover from the error.

System programmer response: If DB2 was able to recover from the error and successfully complete the application, no further action is required. If the application abnormally terminated or DB2 could not recover successfully, this message is followed by one or more messages. Refer to the information in this message and the subsequent messages to determine the appropriate corrective action.

Related concepts:

▶ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

Related reference:

▶ DSN1LOGP (DB2 Utilities)

Related information:

▶ Recovering from active log failures (DB2 Administration Guide)

▶ Recovering from archive log failures (DB2 Administration Guide)

DSNJ152I   *csect-name* **ERROR ENCOUNTERED WHEN ATTEMPTING TO BUILD ARCHIVE LOG VOLUME LIST. CONNECTION-ID=***collection-id***, CORRELATION-ID=***correlation-id***, REASON-CODE=***reason-code*

Explanation: An error occurred while attempting to create the Archive Log Volumes report. An RBA range could not be successfully mapped into one or more archive data sets due to reason code 'ccc'. The user or application that encountered the error is identified by the connection and correlation ID. This message might be preceded by one or more related error messages. Messages that have the same connection ID and correlation ID relate to the same user. Correlation IDs beginning with '0*nn*', where '*nn*' is a number from 01 to 28, identify system agents.

This failure could be caused by one or more missing archive log data sets, or a system error (for example, an I/O error reading BSDS).

The Archive Log Volumes report (see message DSNJ330I) is not produced. In addition, no premounting of tapes is possible.

System action: The user or application continues execution. The physical read process for the user or application continues until the job completes normally or terminates abnormally. The job can terminate abnormally if the error is encountered again when the data set is physically required for the read process.

System programmer response: If the user or application completes successfully, no further action is necessary. If the user or application does not complete successfully, refer to the messages related to the actual failure to determine the appropriate corrective actions.

Related concepts:

▶ Recovering from BSDS or log failures during restart (DB2 Administration Guide)

Related reference:

▶ DSN1LOGP (DB2 Utilities)

Related information:

▶ Recovering from active log failures (DB2 Administration Guide)

▶ Recovering from archive log failures (DB2 Administration Guide)

DSNJ153E   *csect-name* **CRITICAL LOG READ ERROR CONNECTION-ID = ***connection-id*** CORRELATION-ID = ***correlation-id*** LUWID = ***logical-unit-of-work-id=token*** REASON-CODE = ***reason-code*

Explanation: While processing a DB2 must-complete function, DB2 encountered an error during an attempt to access the required log data sets.

CONNECTION-ID, CORRELATION-ID, and LUWID together identify the thread associated with the request. *token* is a unique token number associated with the LUWID. *reason-code* identifies the most recent log-read failure.

System action: The DSNJ154I message requests to retry the log-read request, and immediately follows this error message. DB2 will wait for a reply to message DSNJ154I before retrying the failed log-read request, or before abending.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

System programmer response: Review the description of the reason-code associated with the log-read failure. Examine the system console log for additional messages regarding the log- read error. Correct the cause of the error, if possible, and reply 'Y' to the DSNJ154I message in order to retry the log-read request. If it is not possible to correct the problem causing the log-read errors, and the failing log read request is the result of a thread rollback or -RECOVER POSTPONED command, consider canceling the log read request with the -CANCEL THREAD NOBACKOUT or -RECOVER POSTPONED CANCEL command. See the DB2 Command Reference for details about these commands. If canceling the log read request is not an option, quiesce the work on the DB2 system before replying 'N' to the DSNJ154I message in preparation for DB2 termination.

DSNJ154I   *csect-name* **REPLY Y TO RETRY LOG READ REQUEST, N TO ABEND**

Explanation: While processing a DB2 must-complete function, DB2 encountered an error during an attempt to access the required log data sets.

System action: The DSNJ153E message identifies the critical log-read error, and immediately precedes this message. DB2 will wait for the operator's reply to proceed with the log-read request.

Operator response: Notify the system programmer. When actions have been taken to correct the cause of the log-read errors, reply 'Y' to request a retry of the failing log-read request. A reply of 'N' will fail the log-read request and the task will abend. DB2 then terminates.

System programmer response: Review the description of the reason-code associated with the log-read failure, and examine the system console log for additional messages regarding the log-read error. Correct the cause of the error, if possible, and reply 'Y' to the DSNJ154I message to retry the log-read request. If it is not possible to correct the problem causing the log-read errors, quiesce the work on the DB2 system before replying 'N' to the DSNJ154I message in preparation for DB2 termination.

DSNJ156I   **INVALID BSDS CONVERSION, DSN=**_dataset-name_

Explanation: During initialization time, DB2 detected an invalid BSDS. An invalid BSDS can result from a failure during a prior attempt to run a conversion program.

dataset-name
          The data set name for the BSDS encountering the error.

System action: DB2 startup terminates.

System programmer response: The procedure for running the conversion program includes making a copy of the BSDS before conversion. Restore the BSDS to the original pre-conversion copy, and try the conversion again. After the conversion is successful, restart DB2.

Related reference:

➦ DSNJCNVB (DB2 Utilities)

➦ DSNJCNVT (DB2 Utilities)

DSNJ157I   **BSDS HAS NOT BEEN CONVERTED, DSN=**_xxxxxx_

Explanation: During restart, DB2 detected a BSDS that was neither created in nor converted to the format that allows 10,000 archive logs. When migrating from DB2 Version 8 new-function mode to DB2 Version 9, running the conversion program, DSNJCNVB, is required if not previously done when running DB2 new-function mode.

xxxxxx is the data set name of the BSDS encountering the error.

System action: DB2 startup is terminated.

System programmer response: Run the DSNJCNVB program and restart DB2.

DSNJ200I   *csect-name utility-name* **UTILITY PROCESSING COMPLETED SUCCESSFULLY** *member-name*

Explanation: The control section *csect-name* successfully completed the *utility-name* utility.

After the utility processing completes, the DSNJCNVT BSDS conversion utility displays the DB2 *member-name*.

System action: DB2 writes this message to the SYSPRINT data set, not to a console.

DSNJ201I *csect-name xxxxxxxx* **UTILITY PROCESSING WAS UNSUCCESSFUL**

Explanation: The control section 'csect-name' was unable to successfully complete processing of 'xxxxxxxx' utility.

System action: The current utility is terminated. DB2 writes this message to the SYSPRINT data set, not to a console.

User response: Review other messages produced by the utility to determine the appropriate action to be taken.

DSNJ202I *csect-name* **INSUFFICIENT VIRTUAL STORAGE AVAILABLE TO CONTINUE WITH UTILITY**

Explanation: A GETMAIN macro instruction has failed, because virtual storage is unavailable for the job-step.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Rerun the utility after providing a larger virtual storage allocation for the job-step.

DSNJ203I *xxxxxxxx* **OPERATION IS INVALID**

Explanation: The user entered a utility control statement operation of *xxxxxxxx* that is invalid.

The DSNJU003 utility fails if the DSNTIJGF job is used as part of disabling data sharing on a Version 11 new-function mode BSDS that was not converted by the DSNJCNVT conversion utility.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

If disabling data sharing is required, convert the BSDS with the DSNJCNVT conversion utility before rerunning the utility.

DSNJ204I *xxxxxxxx* **PARAMETER IS INVALID**

Explanation: The user specified a utility control statement parameter of 'xxxxxxxx' that is invalid.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ205I *xxxxxxxx* **PARAMETER HAS NO ARGUMENT**

Explanation: 'xxxxxxxx' contains the name of a parameter that requires an argument.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Specify an argument for the identified parameter and then rerun the utility.

DSNJ206I *xxxxxxxx* **PARAMETER REQUIRES NO ARGUMENT**

Explanation: 'xxxxxxxx' contains the name of the parameter that has been incorrectly followed by an = symbol.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ207I *csect-name* **PARAMETERS INCONSISTENT WITH SPECIFIED OPERATION**

Explanation: The user has specified utility control statement parameters that are inconsistent with the specified utility operation.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ208E *csect name* **MEMBER** *member name* **WAS NOT FOUND IN THE BSDS SPECIFIED IN THE GROUP DD STATEMENT.**

Explanation: You are running the print log map (DSNJU004) utility. No member record exists for this member in the bootstrap data set (BSDS) specified in the GROUP DD statement.

System action: The log map for this member is not printed. If the records for other members are requested and are found in the BSDS specified in the GROUP DD statement, they are printed.

User response: Use the DISPLAY GROUP command to verify that the member name is a valid name and that the member has successfully joined the group. Also verify that the BSDS specified in the GROUP DD statement is correct. If the member name and the BSDS name are correct, then the most likely causes of this message are:

*   The 'missing member' never joined the group because it was never started.
*   The member whose BSDS you specified in the GROUP DD statement was down when the missing member joined the group and has not been restarted.

To gather the information from the missing member, you can rerun the utility, using the BSDS DDNAME option to point to the missing member's BSDS. To avoid this problem in the future, make sure the BSDS named in the GROUP statement is from a recently started member.

DSNJ209I **ENDRBA VALUE MUST BE GREATER THAN STARTRBA VALUE**

Explanation: The 'STARTRBA' and 'ENDRBA' parameters specify an RBA range. Hence, the ENDRBA value must be greater than the STARTRBA value.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ210I **OLD ICF CATALOG NAME=oldname, NEW CATALOG NAME=newname**

Explanation: Change log inventory (DSNJU003) utility processing for the NEWCAT control statement has completed successfully.

oldname
> Either:
> *   The name of the prior ICF catalog or
> *   'NULL' if no prior name existed.

newname
> The name of the ICF catalog defined by the VSAMCAT keyword.

DSNJ211I **UNEXPECTED END OF DATA ON SYSIN DATA SET**

Explanation: Additional control statements were expected but could not be found.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement(s), and rerun the utility.

DSNJ212I *csect-name* **ERROR RETURNED FROM BSDS READ, RPLERRCD=**yy**, DDNAME=**$xxxxxxxx$

Explanation: A VSAM GET was issued that resulted in a nonzero return code. 'yy' contains the error code returned by VSAM. 'xxxxxxxx' contains the ddname of the BSDS encountering the error.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: The action taken is dictated by the return code. The BSDS may have to be recovered by use of a backup copy.

DSNJ213I *csect-name* **ERROR RETURNED FROM BSDS WRITE, RPLERRCD=**vsam-code**, DDNAME=**dd-name

Explanation: A VSAM PUT or ERASE macro was issued, and it returned a nonzero return code.

csect-name
>   The name of the control section that issued the message.

vsam-code
>   The return code that was returned by VSAM.

dd-name
>   The data definition name of the BSDS that encountered the error.

System action: The current utility is terminated. Message DSNJ213I is written to the SYSPRINT data set. The message is not written to console.

User response: Us the information provided for VSAM return codes to resolve the problem. The BSDS might have to be recovered from a backup copy.

Related reference:

➡ VSAM Macro Return and Reason Codes (DFSMS Macro Instructions for Data Sets)

DSNJ214I **SPECIFIED DSNAME ALREADY EXISTS IN BSDS, DDNAME=xxxxxxxx**

Explanation: The user attempted a NEWLOG operation with a data set name that already exists in the BSDS. An entry is never made in a BSDS if the specified DSNAME currently exists in either the active or archive records of that BSDS. 'xxxxxxxx' contains the ddname of the subject BSDS.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Either correct the control statement and rerun the utility, or delete the existing DSNAME from the BSDS and rerun the utility.

DSNJ215I *csect-name* **SVC 99 INFORMATION REQUEST FOR DD NAME '**ddname**' FAILED. RC=** rc**, RSN=** rsncd**, AND INFO CODE** infocode**.**

Explanation: The Print Log Map utility requested information from the MVS Dynamic Allocation facility about the DD name shown in the message, but the system was unable to provide the information. The return code, reason code, and information code were returned by the system's SVC 99 facility. See *MVS/ESA Programming: Authorized Assembler Services Guide* for an explanation of these codes.

System action: If the information request was for the SYSIN DD statement, the Print Log Map utility assumes the control statement was not provided and attempts to continue the request using the SYSUT1 DD statement to locate the BSDS to be mapped.

If the information request was for the DD name of a BSDS, attempts to get information about BSDS data sets is terminated. If information about any BSDS data sets was previously acquired, the maps of those BSDS data sets are printed. Otherwise, the utility terminates.

System programmer response: Use the MVS manual (or the equivalent manual for the level of MVS on your system) to help you in problem determination.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

DSNJ216I   **NO SPACE IN BSDS FOR ADDITIONAL ACTIVE LOG ENTRIES, DDNAME=***xxxxxxxx*

Explanation: The maximum number of active log data sets is fixed. No further entries can be inserted after the maximum has been reached. *xxxxxxxx* contains the ddname of the subject BSDS.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Run the print log map utility to determine the current status of the BSDS. Subsequent actions can then be formulated, depending upon the status of the BSDS.

DSNJ217I   **SPECIFIED DSNAME DOES NOT EXIST IN BSDS, DDNAME=xxxxxxxx**

Explanation: The DELETE operation specifies a DSNAME that cannot be found in the BSDS. 'xxxxxxxx' contains the ddname of the subject BSDS.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ218I   **SPECIFIED VOLUME DOES NOT EXIST IN BSDS, DDNAME=xxxxxxxx**

Explanation: The DELETE operation specifies a COPY1VOL or COPY2VOL argument that cannot be found in the BSDS. 'xxxxxxxx' contains the ddname of the subject BSDS.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement, and rerun the utility.

DSNJ219I   *csect-name* **OPEN ERROR, DDNAME=***xxxxxxxx*

Explanation: An error occurred when the CSECT 'csect-name' tried to open a data set named 'xxxxxxxx'.

This error can be caused by a number of different conditions. The most probable conditions are:

1. The ddname of the SYSPRINT, SYSIN, or SYSUT1 data set was not specified in the user's job control language (JCL).
2. The DB2 subsystem is currently executing.
3. The bootstrap data set (BSDS) was allocated by another job with a disposition (DISP) that conflicts with the DISP specified in the user's JCL.
4. The data set associated with 'xxxxxxxx' is already open, possibly due to an earlier system abend.
5. The user is not authorized to access the data set associated with 'xxxxxxxx'
6. Insufficient virtual storage is available to perform the OPEN command.
7. The catalog indicates that the data set associated with 'xxxxxxxx' has an invalid physical record size.

System action: The current utility is terminated.

DB2 writes this message to the SYSPRINT data set, not to a console.

User response: The user's action depends on the condition that caused the OPEN error. The following is a list of appropriate actions corresponding to the conditions listed in the explanation:

1. Provide the missing DD statements, and then rerun the utility.
2. Wait until the DB2 subsystem is inactive before running the utility again because the log utility cannot execute while the DB2 subsystem is active.
3. Correct the disposition conflict and then rerun the utility.
4. Submit an access method services (IDCAMS) VERIFY job against the data set associated with 'xxxxxxxx'. Rerun the log utility job.
5. In the case of an authorization failure, a separate message is usually generated from the authorization facility (RACF, for example). Investigate the authorization messages and obtain the proper authorization before running the utility again.
6. Insufficient virtual storage is usually accompanied by a separate error from the MVS operating system. Increase the REGION size on the JCL job step and rerun the utility.
7. Contact the system programmer.

Related concepts:

Invoking stand-alone utilities (DB2 Utilities)

DSNJ220I   **BSDS IN CREATE MODE. NO DATA TO MAP, DDNAME=***nnnnnnnn*

Explanation: A utility found the BSDS to be in create mode, meaning it contains no data. *nnnnnnnn* contains the ddname of the BSDS data set.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Ensure that the BSDS data set name specified as input to the utility is correct.

DSNJ221I   **PREVIOUS ERROR CAUSED xxxxxxxx OPERATION TO BE BYPASSED**

Explanation: DB2 encountered errors while processing a utility. These errors subsequently caused 'xxxxxxxx' to be bypassed.

This message is a warning only and is displayed after messages that specify the error or errors that caused the utility operation to abend. Note that the error or errors may not be associated with the current 'xxxxxxxx' operation; rather, under log utility processing, a significant error in any operation causes the control statements for this and any subsequent operations to be checked for syntax only. BSDS updates do not occur for any operation specified in this message.

System action: The log utility continues to process. However, for this and all subsequent operations, the bootstrap data set (BSDS) is not updated and the utility only checks the syntax of the control statements.

DB2 writes this message to the SYSPRINT data set, not to a console.

User response: Consult the previous messages and correct any errors that caused this message to be generated. Resubmit the log utility job for all operations that have been bypassed.

DSNJ222I   **INVALID SPECIFICATION OF xxxx PARAMETER ARGUMENT**

Explanation: You specified the parameter 'xxxx'. This parameter is not valid for the argument.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the parameter argument on the control statement, and rerun the utility.

DSNJ223I   **xxxx PARAMETER ARGUMENT EXCEEDS MAXIMUM ALLOWABLE LENGTH**

Explanation: The argument value of the parameter named 'xxxx' exceeds the maximum length allowed.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the parameter argument on the control statement, and rerun the utility.

DSNJ224I   **xxxx PARAMETER APPEARS TOO OFTEN**

Explanation: The parameter named 'xxxx' was specified more than once on the same control statement.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Remove the redundant parameter, and rerun the utility.

DSNJ225I   **xxxx OPERATION COMPLETED SUCCESSFULLY**

Explanation: The change log inventory (DSNJU003) utility operation named 'xxxx' completed successfully.

System action: This message is written to the SYSPRINT data set, not to a console.

DSNJ226I   **SPECIFIED VOLUME ALREADY EXISTS IN BSDS, DDNAME=xxxxxxx**

Explanation: The specified volume currently exists in the archive log records of the BSDS. 'xxxxxxxx' specifies the ddname of the subject BSDS.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Either correct the parameter argument on the control statement, or delete the specified volume and rerun the utility.

DSNJ227I   **NO SPACE AVAILABLE IN BSDS FOR ADDITIONAL ARCHIVE VOLUME ENTRIES,**
                  DDNAME=xxxxxxxx

Explanation: The maximum number of archive volumes has been exceeded, and no more space is available for volume entries in the copy specified.

System action: The current utility is terminated.

User response: Delete some of the archive entries in the specified copy number, and rerun the utility.

DSNJ228I   *csect-name* **LOG DEALLOCATION ERROR DSNAME=**_dsname_, **ERROR STATUS=**_eeeeiiii_, **SMS**
                  REASON CODE=_ssssssss_

Explanation: The log manager encountered an error while trying to dynamically deallocate the data set. Error status is the error reason code returned by MVS dynamic allocation.

System action: Log manager processing continues.

Operator response: Notify the system programmer.

System programmer response: The error status portion of this message contains the 2-byte error code (S99ERROR) followed by the 2-byte information code (S99INFO) from the SVC request block. If the S99ERROR code indicates an SMS deallocation error (97XX), then 'ssssssss' contains additional SMS reason code information obtained from S99ERSN. Refer to the *MVS/ESA SPL: Application Development Guide* for a description of these codes.

DSNJ229I   *csect-name* **OPEN FAILED FOR SYSIN**

Explanation: An error occurred when the DB2 Print Log Map utility tried to open the SYSIN data set.

Possible causes are:

1. The DD name of the SYSIN data set was not specified in the user's job control language (JCL).
2. Insufficient virtual storage is available to perform the OPEN command.
3. The characteristics of the SYSIN data set are different from those expected by the print log map (DSNJU004) utility.

System action: DSNJU004 attempts to continue processing without an input control statement. DB2 writes this message to the SYSPRINT data set, not to the system console.

User response: Take one of the following actions based on the cause of the error.

1. Provide the missing DD statement and rerun the utility.
2. Insufficient virtual storage is usually accompanied by a separate error message from the MVS operating system. Increase the REGION size on the JCL job step and rerun the utility.
3. Check that the SYSIN data set is a fixed record format (blocked or unblocked) with an LRECL of 80. You might need to contact the system programmer for help if this is a JES data set.

Related concepts:

➦ Invoking stand-alone utilities (DB2 Utilities)

Related reference:

➦ Syntax and options of the DSNJU004 control statement (DB2 Utilities)

DSNJ230I   **LOG OFFLOAD SUBCOMPONENT INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the off-load function was unable to complete its initialization process.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate actions to take.

DSNJ231I   **LOG COMMAND INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the log command function was unable to complete its initialization process.

System action: DB2 startup is terminated.

Operator response: The rest of subsystem termination will direct the actions you should take.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

This message will be of interest to the IBM support center when the problem is described.

DSNJ232I   **OUTPUT DATA SET CONTROL INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the output data set control function was unable to complete its initialization process.

System action: DB2 startup is terminated.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message for error analysis and the appropriate action to take.

This message will be of interest to the IBM support center when the problem is described.

DSNJ233I   **ARCHIVE LOG READ INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the archive log read function was unable to complete its initialization process.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ234I   **ARCHIVE LOG COMMAND QUIESCE INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the quiesce function which supports the -ARCHIVE LOG MODE(QUIESCE) command processing was unable to complete its initialization process.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ235I   **OUTPUT BUFFER WRITER INITIALIZATION PROCESSING FAILED**

Explanation: During log manager initialization, the output buffer writer function was unable to complete its initialization process.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ236I **BOOTSTRAP ACCESS TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the bootstrap data set access function was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ237I *csect-name* **INVALID CONTROL STATEMENT. THE FIRST WORD MUST BE 'MEMBER'.**

Explanation: The print log map (DSNJU004) utility read an invalid control statement from the SYSIN data set.

System action: DSNJU004 ignores the control statement and attempts to continue processing by looking for a SYSUT1 DD statement to describe the BSDS that is to be printed.

User response: Correct or remove the control statement and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related reference:

➡ DSNJU004 (print log map) (DB2 Utilities)

➡ Sample DSNJU004 control statement (DB2 Utilities)

DSNJ238I **LOG OFFLOAD TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the offload function was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific messages for error analysis and the appropriate action to take.

DSNJ239I **LOG COMMAND TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the log command function was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ240I   **OUTPUT DATA SET CONTROL TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the output data set control function was unable to complete its termination process.

System action: The log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ241I   **ARCHIVE LOG READ TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the archive log read function was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ242I   **ARCHIVE LOG COMMAND QUIESCE TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the quiesce function which supports the -ARCHIVE LOG MODE(QUIESCE) command processing was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ243I   **OUTPUT BUFFER WRITER TERMINATION PROCESSING FAILED**

Explanation: During log manager termination, the output buffer writer function was unable to complete its termination process.

System action: Log manager shutdown continues to the next step.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ244I   **LOG TERMINATION RECEIVED RETURN CODE nnn FROM xxxxxxxx MACRO**

Explanation: Log manager termination was passed a bad return code from the named macro.

System action: Log manager termination processing continues.

Operator response: Record the return code and macro-name, and notify the system programmer.

System programmer response: This message is associated with a failure in the named macro. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNJ245I   **CONDITIONAL RESTART - LOG TRUNCATION AT *xxx rrr*. REPLY *Y* TO CONTINUE, *N* TO CANCEL**

Explanation: The conditional restart control record in use indicates that the DB2 log should be truncated at the specified RBA or LRSN.*xxx* is replaced by either RBA or LRSN, and *rrr* is the value specified in the current record.

System action: If *Y*, DB2 startup continues. If *N*, DB2 startup is terminated.

Operator response: Reply *N* if the truncation is going to occur at an undesirable point. Notify system programmer. Reply *Y* to continue the restart.

System programmer response: Rerun the change log inventory (DSNJU003) utility to modify the conditional restart record.

| **DSNJ246I** **CONDITIONAL RESTART - COLD START AT RBA** *rrr*. **REPLY** *Y* **TO CONTINUE,** *N* **TO CANCEL**

Explanation: The conditional restart control record in use indicates that DB2 is to be restart and that logging is to begin at the specified RBA.

System action: If *Y*, DB2 startup continues. If *N*, DB2 startup is terminated.

Operator response: Reply *N* if the truncation is going to occur at an undesirable point. Notify the system programmer. Reply *Y* to continue the cold start.

System programmer response: Rerun the change log inventory (DSNJU003) utility to modify the conditional restart record.

DSNJ247E *csect-name* **I/O ERROR DURING PROCESSING OF** *rrr* **RC =** *rc*, **REASON =** *reason*

Explanation: The I/O error occurred at record '*rrr*'. '*rc*' indicates the return code received from the I/O operation. '*reason*' indicates the reason code received from the I/O operation.

If a return code '4' was received, this indicates that the log manager detected a problem. If a return code '8' was received, this indicates a VSAM error.

System action: DB2 startup terminates.

Operator response: Notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

For a return code of '4', if you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. For a return code of '8', execute an offline access method services job to determine the cause of the VSAM error.

DSNJ248I **ARCHIVE LOG READ DEALLOCATION INITIALIZATION PROCESSING FAILED**

Explanation: During log initialization, the archive log read deallocation process was unable to complete its initialization.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error precede this message. Refer to these messages for error analysis and the appropriate action to take.

DSNJ249I **ARCHIVE LOG READ SCHEDULER INITIALIZATION PROCESSING FAILED**

Explanation: During log initialization, the archive log read scheduling process was unable to complete its initialization.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error precede this message. Refer to these messages for error analysis and the appropriate action to take.

DSNJ250E   *csect-name* **DATA SHARING REQUIRES ALL ACTIVE LOG DATA SETS TO BE DEFINED WITH SHAREOPTIONS(2 3) OR HIGHER.**

Explanation: DB2 detected an active log data set defined with share options that do not permit recovery operations in a data sharing environment. This message is followed by message DSNJ251E, which identifies the specific active log data set.

All active log data sets for a DB2 data sharing member must be defined with a minimum of SHAREOPTIONS(2 3) for successful DB2 initialization. See the description of message DSNJ251E for more details.

System action: The DB2 subsystem detecting this condition is terminated.

DSNJ251E   *csect-name* **DATA SET** *dsname* **FAILS TO MEET THE MINIMUM REQUIRED SHAREOPTIONS FOR DATA SHARING.**

Explanation: The share options defined for the data set named in the message are too low to permit the data set to participate in a DB2 data sharing group. All active log data sets must be defined with SHAREOPTIONS(2 3) or higher in order to participate in a data sharing group.

System action: The DB2 subsystem detecting this error is terminated.

System programmer response: Using IDCAMS LISTCAT or TSO LISTCAT, display the share options of all of the active log data sets of all of the members in the data sharing group. For any active log data set that is defined with less than SHAREOPTIONS(2 3), use IDCAMS ALTER or TSO ALTER to alter the data set definitions to SHAREOPTIONS(2 3) or higher.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
* Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
* IDCAMS LISTCAT output that shows the attributes of all active log data sets for the DB2 member that failed.

.

DSNJ253I   *csect-name alias-port* **USE IS INVALID**

Explanation: The *alias-port* value has been used in an invalid manner. The error could be due to one of the following conditions:

* *alias-port* was specified after more than one alias-name parameter
* *alias-port* duplicates the port value for the PORT or RESPORT parameters which have been specified with this update, or are already stored in the BSDS

System action: The update of the BSDS will not be performed.

User response: Retry the BSDS update with unique *alias-port* values, which also do not match the values for the PORT or RESPORT parameters.

DSNJ254I   *xxxx* **NUMBER OF PARAMETER ARGUMENTS EXCEED MAXIMUM VALUE**

Explanation: The number of arguments for the parameter named *xxxx* exceed the maximum value allowed.

System action: The current operation is terminated. This message is written to the SYSPRINT data set, but not written to a console.

User response: Change the number of parameter arguments on the control statement, and rerun the utility.

DSNJ255I   *csect-name* **CONDITIONAL RESTART ENDTIME OF** *end-time* **WILL RESULT IN LOG TRUNCATION AT** *xxxx yyyyyyyyyyyy*

Explanation: The conditional restart control record in use indicates that the DB2 log should be truncated with ENDTIME *end-time*. The ENDTIME value translates to truncation type *xxxx* with value *yyyyyyyyyyyy*. This value is either an LRSN value (for data sharing) or a STCK (store clock) value (for non-data sharing).

System action: Startup continues.

User response: Reply *Y* to the DSNJ245I WTOR if this value is the correct log truncation value.

DSNJ256I   *csect-name* **CONDITIONAL RESTART RECORD INDICATES SYSPITR RESTART WITH NO LOG TRUNCATION**

Explanation: The conditional restart control record in use indicates that DB2 will be restarted in system recover pending mode without truncating the logs. No database update activity will be allowed, and DB2 will be in restricted access mode.

System action: Startup continues.

Operator response: Reply 'Y' to the DSNJ011I WTOR if this is correct.

DSNJ260I   csect_name **BSDS CONVERSION FOR DDNAME=**nnnnnnnn **COMPLETED SUCCESSFULLY**

Explanation: The BSDS conversion utility issues this message to indicate a successful conversion for the BSDS data set specified by the *nnnnnnnn* DDNAME.

System action: Processing continues.

DSNJ261I   csect_name **BSDS CONVERSION FOR DDNAME=**nnnnnnnn **WAS NOT SUCCESSFUL**

Explanation: The BSDS conversion utility issues this message to indicate conversion was not successful for the BSDS data set specified by the *nnnnnnnn* DDNAME.

System action: The action taken depends on whether dual BSDS data sets are specified, and if so, which conversion failed. The DDNAME in the message indicates the failing BSDS data set.

If dual BSDS data sets are specified, the SYSUT1 data set is converted first. If that conversion is successful, the SYSUT2 data set is then converted. A failure at this point does not affect the successful SYSUT1 conversion.

If conversion fails for the SYSUT1 data set, processing is terminated at that point. In this case, any SYSUT2 data set that was specified will not be converted.

User response: Review other messages produced by the utility to determine the cause of the failure and the resulting action to take.

DSNJ266I   **WARNING - BSDS UTILITY TIMESTAMP MISMATCH DETECTED. PROCESSING TERMINATED.**

Explanation: DB2 detected a mismatch in the timestamps for the SYSUT1 and SYSUT2 BSDS copies during execution of a BSDS conversion utility. This mismatch indicates the possibility that the dual BSDSs are out of sync.

System action: The program terminates with no action taken.

System programmer response: Run the print log map utility (DSNJU004) against each BSDS. From the output, determine which data set is obsolete, delete it, and define a replacement for it. Then copy the remaining data set into the replacement and try the utility again.

If output from the print log map utility for both data sets is similar, delete the data set with the oldest timestamp, and copy the data set with the most recent timestamp into the replacement.

Related reference:

DSNJU004 (print log map) (DB2 Utilities)

DSNJCNVB (DB2 Utilities)

DSNJCNVT (DB2 Utilities)

DSNJ301I   *csect-name* **ERROR** *action* **ONLINE BOOTSTRAP DATA SET**

Explanation: During command processing for the -RECOVER BSDS command or the -ARCHIVE LOG command, an error occurred while performing an operation on the bootstrap data set (BSDS). The type of operation is specified in the 'action' keyword in the message text:

- **ACCESSING:** Unable to OPEN the BSDS.
- **READING:** Unable to read a required record from the BSDS.
- **UPDATING:** Unable to write a required record to the BSDS.
- **RESTORING DUAL:** The contents of the stable BSDS was successfully copied to the replacement BSDS; however, the DB2 subsystem was unable to successfully restore dual BSDS operation.

System action: If this message was received during processing of the -RECOVER BSDS command, the DB2 subsystem will continue in single BSDS mode. If this message was received during processing of the -ARCHIVE LOG command, the archive log history record in the BSDS will not be updated to reflect the occurrence of an -ARCHIVE LOG command; logging and the offload processing will continue.

Operator response: Inform the system programmer of the type of error that occurred, and whether the error occurred while processing a -RECOVER BSDS command or an -ARCHIVE LOG command.

System programmer response: If this message was received during processing of the -RECOVER BSDS command, recovery action must be performed on the bootstrap data set before reentering the command. If this message was received during processing of the -ARCHIVE LOG command, no action is necessary.

DSNJ302I   **ALLOCATION ERROR ON REPLACEMENT BSDS DSNAME=... ERROR STATUS=...**

Explanation: The RECOVER BSDS command encountered an error while trying to dynamically allocate the specified data set. DSNAME is the data set name. Error Status is the error code and information code returned by MVS dynamic allocation.

System action: Processing of the command is terminated. The system continues in single BSDS mode.

Operator response: Determine the cause of the error from the error status contained in the message, and correct the condition. Then reenter the -RECOVER BSDS command.

System programmer response: The error status portion of this message contains the 2-byte error code (S99ERROR) followed by the 2-byte information code (S99INFO) from the SVC request block. Refer to the appropriate MVS publication for a description of these codes.

DSNJ303I   **WRITE ERROR ON REPLACEMENT BSDS DSNAME=... ERROR STATUS=...**

Explanation: The -RECOVER BSDS command encountered an error while attempting to write to the specified bootstrap data set. Error Status contains the VSAM return and feedback codes. It is a 2-byte field with the first containing the hexadecimal return code and the second containing the hexadecimal feedback code.

System action: Processing of the command is terminated. The system continues in single BSDS mode.

Operator response: Notify the system programmer.

System programmer response: Execute an offline access method services job to delete or rename the replacement BSDS and define a new BSDS with the same name. Reenter the log manager -RECOVER BSDS command to reestablish dual BSDS mode.

DSNJ304I   ERROR CLOSING REPLACEMENT BSDS DSNAME=... ERROR STATUS=...

Explanation: The -RECOVER BSDS command encountered an error while attempting to close the specified bootstrap data set. Error Status contains the VSAM return and feedback codes. It is a 2-byte field with the first containing the hexadecimal return code and the second containing the hexadecimal feedback code.

System action: Processing of the command is terminated. The system continues in single BSDS mode.

Operator response: Notify the system programmer.

System programmer response: Execute an offline access method services job to delete or rename the replacement BSDS and define a new BSDS with the same name. Reenter the log manager -RECOVER BSDS command to reestablish dual BSDS mode.

DSNJ305I   REPLACEMENT BSDS NOT EMPTY DSNAME=...

Explanation: The -RECOVER BSDS command was issued, but the replacement bootstrap data set was not empty; that is, it contained data.

System action: Processing of the command is terminated. DB2 continues in single BSDS mode.

Operator response: Notify the system programmer.

System programmer response: Execute an offline access method services job to delete or rename the error BSDS and define a new BSDS with the same name. Reenter the log manager -RECOVER BSDS command to reestablish dual BSDS mode.

DSNJ306I   DUAL BSDS MODE ALREADY ESTABLISHED

Explanation: The -RECOVER BSDS command was issued, but DB2 was already in dual BSDS mode.

System action: The command is ignored.

DSNJ307I   LOG INITIALIZED IN SINGLE BSDS MODE

Explanation: The -RECOVER BSDS command was issued, but the system was initialized in single BSDS mode.

System action: Processing of the command is terminated. The system continues in single BSDS mode.

DSNJ310I   (A)SYNCHRONOUS ARCHIVE LOG COMMAND QUIESCE PROCESSING STARTED FOR MAXIMUM OF xxx SECONDS

Explanation: An -ARCHIVE LOG command with the MODE(QUIESCE) option has been accepted by the DB2 subsystem. The quiesce processing has commenced.

System action: The DB2 subsystem will attempt to stop all updates to DB2 resources within the time period specified in the message. Users/jobs operating on the DB2 subsystem will be allowed to reach a point of consistency (commit point) before being blocked from further update activity. Jobs and users will be suspended until they are released by DB2 following the initiation of the offload processing. If the DB2 subsystem can effectively block all users from performing updates before the maximum specified time, the offload will be initiated immediately, and normal processing will be resumed.

The message will display either the word SYNCHRONOUS or ASYNCHRONOUS, to inform the user of the type of quiesce processing which has been used.

* If the word SYNCHRONOUS is displayed, then the invoker used the WAIT(YES) option. This means that quiesce processing will be synchronous to the user; that is, the user can enter additional DB2 commands, but the commands will not be processed by the DB2 command processor until the quiesce processing has ended.
* If the word ASYNCHRONOUS is displayed, then the invoker used the WAIT(NO) option, or did not specify the WAIT option. This means that quiesce processing will be asynchronous to the user; that is, control will be returned to the invoker as soon as the quiesce task has been started. Thus, the DB2 command processor will accept - and process - any new DB2 commands while the quiesce task is running.

This message will be followed by message DSNJ311I or DSNJ317I.

Operator response: No operator response is necessary. However, it can be expected that users and jobs using DB2

resources will be suspended (queued) by DB2 through the duration of the specified time interval, or until the DB2 subsystem can be certain that all update activity has been effectively blocked. At some point, this message will be followed by the DSNJ311I message or DSNJ317I message.

DSNJ311I   *csect-name* **ASYNCHRONOUS LOG ARCHIVE (OFFLOAD) TASK INITIATED**

Explanation: A user-initiated -ARCHIVE LOG command has been accepted by the DB2 subsystem. An asynchronous task to archive (offload) the active log data set has been started.

System action: The current active log data set(s) will be truncated and switched to the next available active log data set(s), and an asynchronous offload task will be initiated to archive the Active log data set(s).

The term ASYNCHRONOUS is used to indicate that control will be returned to the invoker as soon as the offload task has been started. Thus, the DB2 command processor will accept and process any DB2 commands while the offload task is running.

This message is followed by message DSNJ312I if the MODE(QUIESCE) option was used with the -ARCHIVE LOG command.

Operator response: The operator should respond per normal operational procedures when the offload task begins.

DSNJ312I   **ARCHIVE LOG QUIESCE ENDED. UPDATE ACTIVITY AGAINST DB2 RESOURCES IS NOW RESUMED.**

Explanation: An -ARCHIVE LOG command with the MODE(QUIESCE) option was processed by the DB2 subsystem. As part of the MODE(QUIESCE) processing, an attempt was made to stop all new update activity against DB2 resources. This message signals the end of the quiesce processing, and the resumption of normal DB2 activity for all users and jobs which were blocked during the quiesce period.

This message follows message DSNJ311I or DSNJ317I.

System action: The DB2 subsystem has now resumed all normal DB2 activity for all users and jobs that were blocked during the quiesce period.

DSNJ313I   **PARAMETER SPECIFIED ON THE** *keyword* **KEYWORD (***value***) IS NOT IN THE RANGE OF ALLOWABLE VALUES**

Explanation: This message was generated by an error occurring in one of the following commands:
- ARCHIVE LOG command with MODE(QUIESCE)
- SET ARCHIVE
- SET LOG

You specified a value for one of the command options that was outside of the range of allowed values.

System action: Processing for the ARCHIVE LOG, SET ARCHIVE, or SET LOG command is terminated. This message is followed by message DSN9023I.

Operator response: Correct the syntax and reissue the command.

Related reference:

↪ -ARCHIVE LOG (DB2) (DB2 Commands)

↪ -SET ARCHIVE (DB2) (DB2 Commands)

↪ -SET LOG (DB2) (DB2 Commands)

Related information:
"DSN9023I" on page 1145

DSNJ314I   *keyword-1* **KEYWORD REQUIRES** *keyword-2* **KEYWORD TO ALSO BE SPECIFIED ON THE COMMAND**

Explanation: A command was entered which specified the *keyword-1* keyword. However, use of the *keyword-1* keyword requires that the *keyword-2* keyword also be used.

System action: Command processing terminates for the -ARCHIVE LOG or -RECOVER BSDS command. This message is followed by message DSN9023I.

Operator response: Correct the syntax and reissue the command.

Related reference:

➥ -ARCHIVE LOG (DB2) (DB2 Commands)

➥ -RECOVER BSDS (DB2) (DB2 Commands)

Related information:
"DSN9023I" on page 1145

DSNJ315I   -STOP DB2 MODE(FORCE) IN PROGRESS

Explanation: An attempt was made to issue an -ARCHIVE LOG command when a -STOP DB2 MODE(FORCE) command was already in progress.

System action: Command processing terminates for the -ARCHIVE LOG command. The -STOP DB2 MODE(FORCE) processing continues.

DSNJ316I   SYSTEM QUIESCE ALREADY IN PROGRESS

Explanation: An attempt was made to issue an -ARCHIVE LOG command with the MODE(QUIESCE) option when a system quiesce was already in progress. The system quiesce could be the result of processing by another -ARCHIVE LOG MODE(QUIESCE) command, or by a -STOP DB2 MODE(QUIESCE) command.

System action: Command processing will terminate. The system quiesce currently in progress will continue.

DSNJ317I   ARCHIVE LOG QUIESCE PERIOD EXPIRED. NUMBER OF OUTSTANDING UR'S = xxxx. ARCHIVE LOG PROCESSING WILL BE TERMINATED, AND UPDATE ACTIVITY AGAINST DB2 RESOURCES WILL BE RESUMED. HH:MM:SS

Explanation: An -ARCHIVE LOG MODE(QUIESCE) command was processed by the DB2 subsystem; however, the DB2 subsystem was not able to quiesce all update activity in the user-specified quiesce time interval.

System action: This message is informational only. The DB2 subsystem determined that 'xxxx' units of recovery (URs) did not reach a point of consistency during the quiesce period, and therefore could not be stopped from continuing their associated update processing.

Consequently, the -ARCHIVE LOG processing will be terminated. The current active log data set(s) will not be truncated, and will not be switched to the next available active log data set(s). The log archive (offload) task will not be created. All jobs and users suspended during the quiesce will be resumed, and normal update activity against DB2 resources will be commenced.

This message will be followed by the DSNJ312I message.

Operator response: Contact the system programmer.

System programmer response: A determination must be made whether the outstanding (non-quiesced) URs represent significant work.

Each user on the system has a unit of recovery if they are modifying DB2 resources. URs are also created by the DB2 subsystem itself for internal processing. Since the purpose of the MODE(QUIESCE) option is to have all URs reach a point of consistency (commit point) before the active log data set is truncated and offloaded, determine all outstanding non-queued jobs and users by using -DISPLAY THREAD and the MVS 'D A,L' operator command.

It should be noted that URs may be outstanding due to lock contention between a user or job which holds a resource (and has reached a point of consistency), and a user or job which wants a lock (and therefore cannot reach a point of consistency).

Before resubmitting the -ARCHIVE LOG command with the MODE(QUIESCE) option, either:

- Wait until the threads have been deallocated.
- Wait until the DB2 subsystem is less busy.
- Force the offending thread(s) to terminate.

- Use the TIME option to dynamically override and extend the maximum quiesce time period specified in subsystem parameters.
- Or, if having all URs reach a point of consistency in the active log is no longer critical, issue the -ARCHIVE LOG command without the MODE(QUIESCE) option.

Important: If the decision is made to use the -ARCHIVE LOG command without the MODE(QUIESCE) option, the active log data set(s) will be truncated without regard to quiescing activity on the DB2 subsystem. If the resulting archive log data set is used for recovery, it is possible that some URs may be found to be inflight, in-abort, in-commit, or indoubt during DB2 initialization.

If expiration of the quiesce period before all URs reach a consistent point are a chronic problem, you might need to adjust the value of the QUIESCE subsystem parameter.

Related reference:

⤵ QUIESCE PERIOD field (QUIESCE subsystem parameter) (DB2 Installation and Migration)

⤵ -ARCHIVE LOG (DB2) (DB2 Commands)

Related information:

"DSNJ312I" on page 276

DSNJ318I   **ARCHIVE LOG COMMAND ALREADY IN PROGRESS**

Explanation: An attempt was made to issue an ARCHIVE LOG command when an ARCHIVE LOG command was already in progress. In data sharing, the ARCHIVE LOG command currently in progress could be initiated by another member.

Operator response: If the conflicting command satisfies your archival request, do not reissue the command. Otherwise, wait for the conflicting command to complete and then reissue the -ARCHIVE LOG command.

DSNJ319I   *csect name* **CURRENT ACTIVE LOG DATA SET IS THE LAST AVAILABLE ACTIVE LOG DATA SET. ARCHIVE LOG PROCESSING WILL BE TERMINATED.**

Explanation: The -ARCHIVE LOG command was rejected because the current active log is the last available active log data set. To process the command when these conditions exist would cause the DB2 subsystem to exhaust its available active log resources and immediately halt processing.

System action: Processing for the command is terminated. If the situation is not corrected, the DB2 subsystem will issue the DSNJ110E message (if it has not already done so) as the available active log data space reaches critically low levels. Ultimately, message DSNJ111E will be issued when the available active log data space is exhausted, and the DB2 online processing will halt until active log space is made available.

Operator response: To clear this condition, steps must be taken to complete other waiting offload tasks. Once another active log is made available (re-usable) by completing the offload process for it, the command processing for the current active log can proceed.

Execute an MVS *Display System Requests* command (D R,L for example) to determine the outstanding requests related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue. If offload fails to complete normally, or cannot be initiated, notify the systems programmer.

System programmer response: Either correct the problem that is causing the offload process to fail, or add more active log data sets. Note that the latter action requires DB2 to be down and the change log inventory (DSNJU003) utility to be run.

Possible causes for the shortage of active log data space are:

- Excessive logging. For example, too many users requesting logs at the same time, or too many utilities specifying LOG (YES).
- Delayed or slow offloading. For example, failure to mount archive volumes, incorrect replies to offload messages, or slow device speeds.
- Excessive use of the -ARCHIVE LOG command. Each invocation of the command causes the DB2 subsystem to switch to a new active log data set. Excessive use could consume the available active log data space if the resulting offloads were not processed in a timely manner.
- Off-loads(s) failed (see Operator Response).

- Insufficient active log space (see Operator Response).

User response: Notify the operator or system programmer.

DSNJ320I *csect-name* **UNABLE TO PROCESS LOG TRUNCATION REQUEST DUE TO SUBSYSTEM COMPONENT** *action* **FAILURE**

Explanation: While processing an -ARCHIVE LOG command, an internal request was made of the log buffer output routine to force-write the log buffers and to truncate and switch the active log to the next available active log data set(s). The request failed during the processing of the 'action' specified in the message.

System action: Processing for the command is terminated.

Operator response: Notify the system programmer.

System programmer response: This error is an internal error detected by the DB2 subsystem. The error may be caused by an unrelated ABEND in the Log Buffer Writer component (DSNJWxxx), by a -STOP DB2 MODE(FORCE) command, or by abnormal subsystem termination. Refer to messages and ABENDS which precede this message.

User response: Notify the system programmer.

DSNJ321I **UNABLE TO CONTINUE ARCHIVE LOG QUIESCE DUE TO INTERNAL SUBSYSTEM ERROR. ARCHIVE LOG PROCESSING WILL BE TERMINATED, AND UPDATE ACTIVITY AGAINST DB2 RESOURCES WILL BE RESUMED.**

Explanation: An -ARCHIVE LOG command with the MODE(QUIESCE) option was processed by the DB2 subsystem. As part of the MODE(QUIESCE) processing, an attempt was made to stop all new update activity against DB2 resources. During the internal subsystem interaction between the log manager and the recovery manager, an internal interface error was detected by the recovery manager.

System action: The -ARCHIVE LOG MODE(QUIESCE) processing is terminated. This message will be followed by message DSNJ312I after all users and jobs quiesced by the MODE(QUIESCE) processing are resumed.

Operator response: Notify the system programmer.

System programmer response: This error is an internal error detected by the DB2 subsystem. Retry the -ARCHIVE LOG MODE(QUIESCE) command. If the error persists, the active log data sets may be switched using the -ARCHIVE LOG command without the MODE(QUIESCE) option.

User response: Notify the system programmer.

DSNJ322I **DISPLAY ARCHIVE REPORT FOLLOWS—**

Explanation: This message displays output from the DISPLAY ARCHIVE command. This output contains information about archive log data sets that are in use for the subsystem or data-sharing member. For data-sharing environments, it does not include information about the entire group.

The output begins with the following lines that describe the basic characteristics of the archive units for reading the archive logs:

```
DSNJ322I  -DISPLAY ARCHIVE REPORT FOLLOWS-
              COUNT                 TIME
            (TAPE UNITS)          (MIN,SEC)
name           count                time
name           count                time
```

name   An identifier for the *count* and *time* values that follow on the same line.

   name can have one of the following values:

   DSNZPARM
      The *count* and *time* values that follow are the values that were specified during DB2 installation. These values correspond to the following subsystem parameters:

      - The *count* value is from the MAXRTU subsystem parameter.
      - The *time* value is from the DEALLCT subsystem parameter.

   CURRENT
      The *count* and *time* values that follow are the current values in the DB2 subsystem. These values

might be the same as the subsystem parameter values that are listed in the DSNZPARM line. However, if those values were overridden by the SET ARCHIVE command, this line lists those changed values.

count   The maximum number of tape units that can be allocated to concurrently read archive log tape volumes.

time   The length of time that an allocated archive read tape unit can remain unused before it is deallocated. The format is *minutes,seconds*.

After the information about archive units, the output contains the following lines:

```
===============================================
ADDR     STATUS  CORR-ID  VOLSER  DATASET_NAME
addr status corr-id vname    dsname
...
END OF DISPLAY ARCHIVE REPORT.
```

These lines describe the archive log data sets that are being read. After the heading information, one line is displayed for each archive log data set that is being read. Each of those lines includes the following information:

addr   The physical address of the device that is allocated to read the archive log.

status   The status of the device, which can be one of the following values:

   BUSY   The device is actively processing an archive log data set.

   PREM   The device is active and allocated for premount.

   AVAIL   The device is inactive and waiting for work.

   RECALL
       The archive data set is being recalled by HSM.

corr-id   The correlation id that is associated with the user of the device that is being processed. If the status is PREM or AVAIL, this column might contain *****, which indicates that a current user is not associated with this device.

vname   The name of the archive volume that is mounted at the device address.

dsname   If *status* is BUSY, PREM or RECALL, the name of the data set on the archive volume that is being processed.

   If *status* is AVAIL, the name of the data set on the archive volume that was last processed.

If no archive logs are currently being read, the following message is displayed instead of the lines that describe the archive log data sets:

```
NO ARCHIVE READ ACTIVITY
```

System action: Processing continues.

User response: No action is required.

Related reference:

➡ -DISPLAY ARCHIVE (DB2) (DB2 Commands)

➡ -SET ARCHIVE (DB2) (DB2 Commands)

➡ READ TAPE UNITS field (MAXRTU subsystem parameter) (DB2 Installation and Migration)

➡ DEALLOC PERIOD field (DEALLCT subsystem parameter) (DB2 Installation and Migration)

DSNJ330I   **ARCHIVE LOG VOLUMES REQUIRED FOR CONNECTION ID conn-id CORRELATION ID corr-id**

Explanation: This message is the title for the ARCHIVE LOG VOLUMES report output. The ARCHIVE LOG VOLUMES report lists the archive log volumes needed by the correlation ID for that connection ID. This message is generated automatically by the archive read process at the first archive log tape mount for corr-id. This message precedes the other messages generated by this action.

This message, in combination with messages DSNJ331I and DSNJ332I, provides the formatted report for the ARCHIVE LOG VOLUMES requested.

conn-id
An identifier representing the 'connection name' used to establish the thread.

corr-id An identifier associated with a specified thread. In TSO, it is either the authorization ID or the job name.

DSNJ331I   avolume, avolume, avolume, avolume ...

Explanation: This message provides the tape volume information for the Archive Log Volumes report output.

avolume
The name of the archive log volume required for this correlation ID. An 'avolume' prefixed with an '*' signifies that the data on the archive log volume is also mapped by an active log data set. As such, the 'avolume' **might not** be required for the read process, since the data is read from the active log, if possible.

DSNJ332I   END OF ARCHIVE LOG VOLUMES REPORT

Explanation: This message follows messages DSNJ330I and DSNJ331I and indicates that the ARCHIVE LOG VOLUMES report was generated successfully. The message number is not included in this message.

DSNJ333I   *csect-name* SYSTEM CHECKPOINT INITIATED

Explanation: This message indicates that the SET LOG command processed successfully and that a system checkpoint has been initiated.

DSNJ334I   PARAMETER SPECIFIED ON THE *keyword* KEYWORD (VALUE) CONTAINS INVALID CHARACTERS

Explanation: A -SET ARCHIVE command was processed by the DB2 subsystem. One of the command keywords specified a value that contains an invalid character.

System action: Processing for the -SET ARCHIVE command is terminated. This message is followed by the DSN9023I message.

Operator response: Correct the syntax and reissue the command.

Related reference:

➡ -SET ARCHIVE (DB2) (DB2 Commands)

Related information:

"DSN9023I" on page 1145

DSNJ335I   INVALID KEYWORDS COMBINATION IN SET ARCHIVE COMMAND

Explanation: This command requires that keyword COUNT or keyword TIME be specified. If both are omitted, the keyword DEFAULT must be specified. It is invalid to specify keyword DEFAULT when either the COUNT keyword or TIME keyword is specified.

System action: Processing for the -SET ARCHIVE command is terminated. This message is followed by the DSN9023I message.

Operator response: Correct the syntax and reissue the command.

Related reference:

➡ -SET ARCHIVE (DB2) (DB2 Commands)

Related information:

"DSN9023I" on page 1145

DSNJ336I   UNABLE TO ALLOCATE TAPE UNIT FOR conn-id corr-id, x ALLOCATED, y ALLOWED.

Explanation: An attempt to allocate a tape unit for connection ID 'conn-id' failed. The current maximum tape unit specified is 'y', but only 'x' are physically available.

System action: The 'conn-id corr-id' process is queued until either an allocated tape unit becomes free or more tape units are varied online and made available to the DB2 archive read task.

Operator response: This situation rectifies itself over time as currently allocated tape units become available. To improve throughput, vary additional tape units online and make them available to DB2. Note that a queue rescan is not attempted until the -SET ARC READ COUNT command is issued or when an allocated tape dismounts.

DSNJ337I   SET ARCHIVE COMMAND COMPLETED, COUNT (count) TIME (minute,second).

Explanation: This message indicates that the -SET ARCHIVE command processed successfully and displays the current values for COUNT and TIME.

count   The value for the COUNT parameter

minute The 'minute' value for the TIME parameter

second The 'second' value for the TIME parameter

DSNJ339I   *csect-name* **SET LOG COMMAND COMPLETED,** *frequency-type*(*frequency-unit*)

Explanation: This message indicates that the SET LOG command processed successfully, and displays the current checkpoint frequency. The checkpoint frequency remains in effect until DB2 is restarted or until the value is modified by a SET LOG or SET SYSPARM command.

csect-name
> The name of the control section that issued the message.

frequency-type
> Identifies how the checkpoint frequency is determined. Expected values are CHKTIME (for a frequency based on elapsed time) and LOGLOAD (for a frequency based on the number of log records that are written).

frequency-unit
> Specifies the unit for the associated *frequency-type* value.

The checkpoint frequency is controlled by the following fields on installation panel DSNTIPL1:
- CHECKPOINT TYPE (CHKTYPE subsystem parameter)
- RECORDS/CHECKPOINT (CHKFREQ and CHKLOGR subsystem parameters)
- MINUTES/CHECKPOINT (CHKFREQ and CHKMINS subsystem parameters)

Multiple forms: This message has two forms. The short form is shown above. The long form is:*csect-name* SET LOG COMMAND COMPLETED, *frequency-type*(*frequency-unit*) OR *frequency-type*(*frequency-unit*) The long form is used when both the number of minutes and number of log records were changed by the SET LOG command. The message shows the *frequency-type* and *frequency-unit* values for both.

System action: Processing continues, and DB2 issues a DSNJ371I message to show the current settings.

Related concepts:

➡️ Checkpoint log records (DB2 Administration Guide)

Related tasks:

➡️ Choosing a checkpoint frequency (DB2 Performance)

Related reference:

➡️ -SET LOG (DB2) (DB2 Commands)

➡️ -SET SYSPARM (DB2) (DB2 Commands)

➡️ Checkpoint parameters: DSNTIPL1 (DB2 Installation and Migration)

Related information:
"DSNJ371I" on page 287

DSNJ340I   *csect-name* **COMMAND KEYWORD** *xxx* **IS ONLY VALID FOR DATA SHARING**

Explanation: An attempt was made to issue a DB2 command with KEYWORD *xxx* on a DB2 that was not initialized to participate in data sharing.

System action: The command request is rejected.

DSNJ341I   *csect-name* **-ARCHIVE LOG COMMAND NOT ALLOWED, LOGGING IS SUSPENDED**

Explanation: An attempt to issue an -ARCHIVE LOG command failed because a -SET LOG SUSPEND command has suspended logging.

System action: Command processing terminates for the -ARCHIVE LOG command. You may reissue the command when logging has been resumed by the -SET LOG RESUME command.

DSNJ350I   *csect-name* **(A)SYNCHRONOUS ARCHIVE LOG COMMAND QUIESCE PROCESSING STARTED IN DATA SHARING**

Explanation: The DB2 subsystem of a data sharing group accepted an ARCHIVE LOG command with the MODE(QUIESCE) option.

Each active DB2 member in the data sharing group performs the quiesce operation. The member originating the quiesce issues message DSNJ350I; all other members issue DSNJ310I as they start quiescing.

DSNJ351I   *csect-name* **MEMBER** *member name* **STATUS=***member status*

Explanation: This message is one of a list of elements that display archive related status of the indicated member of the data sharing group. These member status messages are preceded by the message DSNJ359I and are followed by DSNJ361I.

member name identifies the member of the data sharing group.

member status indicates archive related status of the named member. It appears as one of the following:.

ABNORMALLY TERMINATING
> The named member was undergoing abnormal subsystem termination at the time ARCHIVE LOG processing was active for the data sharing group. ARCHIVE LOG command processing for the remainder of the data sharing group terminates unsuccessfully.

ARCHIVE ALREADY ACTIVE
> ARCHIVE LOG command processing was running at the named member at the time a succeeding ARCHIVE LOG command was issued. The succeeding ARCHIVE LOG command terminates unsuccessfully for the remaining members of the data sharing group.

CANCELED
> ARCHIVE LOG command processing for the named member was canceled due to a problem encountered elsewhere in the data sharing group. The quiesce operation is interrupted and terminated and update activity against DB2 resources is resumed before command processing is terminated for the member.

Chapter 8. DSNJ messages   **283**

COMMUNICATIONS FAILURE

A communications failure occurred between the named member and one or more other members in the data sharing group. The quiesce operation is interrupted and update activity against DB2 resources is resumed before command processing is terminated for the member. ARCHIVE LOG command processing for the remainder of the data sharing group terminates unsuccessfully.

FAILED

The named member was inactive at the time the ARCHIVE LOG command was issued. Termination of the DB2 subsystem for this member was abnormal. This status, if displayed for any member, will cause ARCHIVE LOG command processing for the entire data sharing group to terminate unsuccessfully.

NOT AVAILABLE

Archive command status for the named member could not be determined. This condition typically occurs when the named member fails to respond during ARCHIVE LOG command processing and is usually accompanied by message DSNJ353I and/or message DSNJ355I.

QUIESCED

The named member was active at the time the ARCHIVE LOG command was issued. Update activity against DB2 resources had successfully quiesced for that member (no active URs) in response to specifying MODE(QUIESCE).

QUIESCED/OFFLOADING

The named member was active at the time the ARCHIVE LOG command was issued. Update activity against DB2 resources had successfully quiesced for that member (no active URs) in response to specifying MODE(QUIESCE). The active log at the time was truncated and an offload process was initiated after all members in the data sharing group had reached a quiesce point.

STOPPED

The named member was inactive at the time the ARCHIVE LOG command was issued. Termination of the DB2 subsystem for this member was normal with no incomplete URs outstanding. Log truncation and offload processing are scheduled to be performed automatically the next time the named member is started. When the member is started, it is important to ensure that archive processing occurs automatically. Certain error conditions might require entering an ARCHIVE LOG command via the operator console to cause log truncation and offload processing to be performed. This member status is accompanied by warning message DSNJ362A.

TIME EXCEEDED

The named member could not quiesce its update activity within the allotted period of time. The incomplete quiesce operation is interrupted and update activity is resumed before command processing is terminated for the member. ARCHIVE LOG command processing for the remainder of the data sharing group is unsuccessfully terminated.

DSNJ352W   *csect-name* **MEMBERS WERE NOT OFFLOADED**

Explanation: DB2 issues this message during processing of an ARCHIVE LOG command with the MODE(QUIESCE) option. One or more quiesced members were not available to receive the archive request. DSNJ351I messages follow to display the quiesced members.

At the next start up of the quiesced member, DB2 switches (truncates) the current active log and initiates the archive task.

DSNJ353I   *csect-name* **DSNJ353I MEMBER** *member name* **NOT RESPONDING.**

Explanation: This message is displayed when a response is needed but not received within an expected period of time during ARCHIVE LOG MODE(QUIESCE) command processing. This can occur as a result of abnormal subsystem termination under the named member or, possibly from communications problems among members of the data sharing group.

Operator response: Examine the console log of the named member for operator information messages that might relate to the problem.

DSNJ354I   *csect-name* **DSNJ354I ARCHIVE LOG: ALL ACTIVE MEMBER(S) STARTED OFFLOAD TASKS**

Explanation: This message is displayed when MODE(QUIESCE) is specified with the ARCHIVE LOG command, after all active members in the data sharing group have successfully quiesced their update activity within the allotted time period. For each active member in the data sharing group, the active log is truncated and switched, and an offload task is initiated to begin the offload process. Log truncation, switch and offload processing for stopped (and quiesced) members is performed the next time they are started.

DSNJ355I   *csect-name* **ARCHIVE LOG: ONE OR MORE ACTIVE MEMBERS FAILED TO RESPOND OR START OFFLOAD TASKS**

Explanation: This message is displayed when MODE(QUIESCE) is specified with the ARCHIVE LOG command, and one or more active members could not successfully quiesce their update activity in the time allotted. Subsystem activity already quiesced is resumed, and ARCHIVE LOG command processing terminates unsuccessfully for all active members in the data sharing group. Message DSNJ353I might also appear with this message.

Operator response: See the list of DSNJ351I member status messages following message DSNJ359I to determine the reason for the archive command failure.

DSNJ356I   *csect-name* **COMMUNICATIONS FAILURE TERMINATED COMMAND**

Explanation: This message appears as a result of the failure of a communications request by one or more members of the data sharing group. This message is likely to be accompanied by DSN7xxx or other error messages and/or an SVC dump.

System action: ARCHIVE LOG command processing terminates unsuccessfully for all members in the group.

DSNJ357E   *csect-name* **SYSTEM QUIESCE FAILED**

Explanation: This message indicates the unsuccessful outcome of an ARCHIVE LOG MODE(QUIESCE) command. This message indicates that one or more members were unable to quiesce their update activity in the time allowed. It is usually accompanied by one or more related messages (e.g., DSNJ355I), and a list of member status messages indicating the reason for the failure.

Operator response: Refer to messages DSNJ359I, DSNJ351I and DSNJ361I to determine archive related status of members in the data sharing group.

DSNJ359I   **MEMBER RESPONSES**

Explanation: This message marks the beginning of a list of one or more DSNJ351I member status messages and precedes the first status message in the list.

DSNJ361I   *csect-name* **END OF MEMBER RESPONSES LIST**

Explanation: This message marks the end of a list of one or more DSNJ351I member status messages and follows the last status message in the list.

DSNJ362E   *csect-name* **WARNING: LOGS OF** *nn* **STOPPED OR FAILED MEMBERS NOT OFFLOADED.**

Explanation: This message is displayed when either MODE(QUIESCE) or SCOPE(GROUP) is specified with the ARCHIVE LOG command, and one or more members of the data sharing group either stopped or failed and were not available to receive the archive request. Active log truncation, log switch and offload processing for these inactive members is scheduled to be performed the next time those members are started.

Operator response: Verify that log switch and offload processing does automatically occur when the inactive members are started. Certain error conditions might require the ARCHIVE LOG command to be individually issued for a member if ARCHIVE LOG processing does not occur automatically. Refer to messages DSNJ359I, DSNJ351I, and DSNJ361I to identify the inactive members.

DSNJ363I   *csect-name* **COPY** *log-copy* **LOG DATA SET** *dataset-name* **ADDED TO THE ACTIVE LOG INVENTORY**

Explanation: The active log data set identified by *dataset-name* was successfully added to active log inventory with the SET LOG command. The log data set is now available for use by DB2.

csect-name
> The name of the control section that issued the message.

log-copy The new log copy.

dataset-name
> The name of the data set.

System action: Processing continues.

User response: No action is required.

Related reference:

➦ -SET LOG (DB2) (DB2 Commands)

DSNJ364I   **NEWLOG OPERATION FAILED. REASON:** *reason-string*

Explanation: The NEWLOG option of the SET LOG command has failed. The value of *reason-string* describes the cause of the failure:

OPEN ERROR
> An error occurred attempting to allocate or open the new log data set. Make sure that the NEWLOG data set name is correct and that the data set has been defined correctly with IDCAMS.

BSDS ERROR
> An error occurred attempting to update the BSDS.

DUPLICATE LOG DATA SET NAME
> The specified log data set name already exists in the current active log inventory. Make sure that the NEWLOG data set name is correct.

LOG COPY AT MAXIMUM NUMBER OF LOGS
> The specified log copy already contains the maximum number of active log data sets. You can specify up to a maximum of 93 log data sets per active log copy.

COPY2 IN SINGLE LOGGING MODE
> COPY(2) was specified but DB2 was started in single logging mode.

OUT OF SPACE IN BSDS
> There is no space in the BSDS to add the new log data set information. This failure indicates a possible problem with the BSDS control record. If the problem persists, contact IBM Software Support. Prepare an IDCAMS PRINT of the BSDS.

COMMAND ABENDED
> An ABEND occurred during command processing.

System action: The command terminates.

User response: Examine the console log for messages related to the failure.

Related reference:

➦ -SET LOG (DB2) (DB2 Commands)

DSNJ370I   *csect-name* **LOG DISPLAY CURRENT COPY1 LOG =** *dsname1* **IS** *pct***% FULL CURRENT COPY2 LOG = ** *dsname2* **IS** *pct***% FULL H/W RBA =** *hw-rba***, H/O RBA =** *ho-rba* **FULL LOGS TO OFFLOAD =** *nn* **OF** *mm***, OFFLOAD TASK IS** *status*

Explanation: This message, in combination with message DSNJ371I, provides the output from the DISPLAY LOG command.

dsname1
> The data set name of the current active log copy1 data set. If this copy of the log is not currently active, dsname1 will display INACTIVE.

dsname2
> The data set name of the current active log copy2 data set. If this copy of the log is not currently active, dsname2 will display INACTIVE.

pct   The percentage of log space within the active log data set that has been written.

hw-rba The RBA of the most recently written log record.

ho-rba The highest log RBA that has been offloaded to an archive log data set.

nn   The number of full active log data sets that have not been archived.

mm   The total number of active log data sets.

status Status of the offload task:

> **BUSY,ALLC**
> > Task is busy, allocating an archive log data set. This could indicate a tape mount request is outstanding.
>
> **BUSY,BSDS**
> > Task is busy, copying the BSDS data set.
>
> **BUSY,ACTV**
> > Task is busy, copying an active log data set.
>
> **BUSY**   Task is busy, other processing.
>
> **AVAILABLE**
> > Task is available for work.

DSNJ371I   **DB2 RESTARTED** *time-date-1* **RESTART RBA** *rstrt-rba* **CHECKPOINT FREQUENCY** *quantity-and-type* **LAST SYSTEM CHECKPOINT TAKEN** *time-date-2*

Explanation: This message, in combination with message DSNJ370I, provides the output from the DISPLAY LOG command.

When the number of minutes and the number of log records are both used to determine the checkpoint frequency, DB2 issues an alternate form of the message that shows both values:

```
DB2 RESTARTED time-date-1 RESTART RBA rstrt-rba CHECKPOINT FREQUENCY quantity-and-type OR
quantity-and-type LAST SYSTEM CHECKPOINT TAKEN time-date-2
```

time-date-1
> The time and date when DB2 started.

rstrt-rba
> The RBA that DB2 began logging when started.

quantity-and-type
> The number of minutes or the number of log records that determine the current checkpoint frequency.

time-date-2
> The time and date of the last system checkpoint.

DB2 also issues this message at the completion of a SET LOG command that specifies either the LOGLOAD or CHKTIME parameters.

Related concepts:

▶ Checkpoint log records (DB2 Administration Guide)

Related tasks:

▶ Choosing a checkpoint frequency (DB2 Performance)

Related reference:

▶ CHECKPOINT TYPE field (CHKTYPE subsystem parameter) (DB2 Installation and Migration)

▶ -SET LOG (DB2) (DB2 Commands)

▶ RECORDS/CHECKPOINT field (CHKFREQ and CHKLOGR subsystem parameters) (DB2 Installation and Migration)

Related information:

DSNJ372I *csect-name* **UPDATE ACTIVITY HAS BEEN SUSPENDED FOR** *aaaa* **AT RBA** *xxxxxxxxxxxx*, **LRSN** *yyyyyyyyyyyy*, **PRIOR CHECKPOINT RBA** *zzzzzzzzzzzz*

Explanation: This message indicates that the -SET LOG command with the SUSPEND option has completed successfully.

System action: All log and database update activity has been suspended for DB2 subsystem *aaaa*. *xxxxxxxxxxxx* contains the RBA of the last record written to the log. In a data sharing environment, *yyyyyyyyyyyy* contains the LRSN of the last record written to the log. Otherwise, *yyyyyyyyyyyy* contains the high written RBA. *zzzzzzzzzzzz* contains the RBA of the last system checkpoint.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Issue the -SET LOG command with the RESUME keyword when ready to resume update activity.

DSNJ373I *csect-name* **UPDATE ACTIVITY HAS BEEN RESUMED FOR** *xxxx*

Explanation: This message indicates that the -SET LOG command with the RESUME option has completed successfully.

System action: All log and database update activity has been resumed for DB2 subsystem *xxxx*.

DSNJ374I *csect-name* **SET LOG** *options* **COMMAND FAILED,** *reason*

Explanation: A -SET LOG command has failed. *options* were specified on the command. *reason* describes the cause of the failure:

ALREADY SUSPENDED
A -SET LOG SUSPEND command was issued while update activity was suspended by a previous command.

DB2 STOPPING
A -SET LOG SUSPEND command was issued, but DB2 is stopping.

QUIESCE ACTIVE
A -SET LOG SUSPEND command was issued, but a system quiesce that was initiated by an -ARCHIVE LOG command is in progress.

NOT SUSPENDED
A -SET LOG RESUME command was issued, but update activity is not suspended.

COMMAND ABEND
A -SET LOG command abnormally terminated. Update activity is not suspended.

LOCK TIMEOUT
A lock timeout has occurred. Update activity is not suspended.

System action: Command processing terminates.

DSNJ401I   *csect-name rrrr* **RECORD NOT FOUND**

Explanation: An attempt was made to read the 'rrrr' record from the BSDS. In so doing, the read routine (DSNJU01B) could not find the record.

System action: Utility processing continues. This message is written to the SYSPRINT data set, not to a console.

DSNJ402I   **UNABLE TO LOCATE aaaa TBLSPACE NAME IN SYSTEM CATALOG OR DIRECTORY DATABASES**

Explanation: The table space name specified on the TBLSPACE parameter of the change log inventory (DSNJU003) utility SYSTEMDB statement could not be found in any DB2 subsystem catalog or directory database. The table space name in error is named in the 'aaaa' message keyword.

System action: All subsequent DSNJU003 utility statements are bypassed, and DSNJU003 processing is terminated with a return code of 4. The DB2 SYSTEMDB record is not updated.

This message is written to the SYSPRINT data set, not to a console.

User response: Correct the table space name specified on the TBLSPACE parameter, and resubmit the change log inventory (DSNJU003) utility job.

DSNJ403I   **CAPACITY OF BSDS SYSTEMDB RECORD EXCEEDED**

Explanation: The BSDS SYSTEMDB record contains VSAM passwords for data sets in the DB2 catalog and directory, and has a maximum capacity of 168 entries. This message is issued when the number of eligible DB2 catalog and directory data sets exceeds the maximum capacity of the BSDS SYSTEMDB record. This is an internal error.

System action: All subsequent change log inventory (DSNJU003) utility statements are bypassed, and DSNJU003 processing is terminated with a return code of 4. The DB2 SYSTEMDB record is not updated.

This message is written to the SYSPRINT data set, not to a console.

User response: Notify the systems programmer.

DSNJ404E   **keyword NOT ALLOWED FOR operation OPERATION**

Explanation: An invalid keyword was used during the operation named 'operation'.

System action: The current utility processing is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ405E   **keyword1 KEYWORD MUTUALLY EXCLUSIVE WITH keyword2 KEYWORD**

Explanation: 'keyword1' and 'keyword2' are mutually exclusive and cannot appear on the same control statement.

System action: The current utility processing is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Determine the extraneous keyword and remove it from the control statement. Rerun the utility.

DSNJ406E   **EITHER keyword1 KEYWORD OR keyword2 KEYWORD MUST BE SPECIFIED**

Explanation: A required keyword was not used on the control statement. Use either 'keyword1' or 'keyword2' with the attempted control statement type.

System action: The current utility processing is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ407I   *csect-name* **NO VALID CHECKPOINT RBA FOUND. LOG WILL BE SCANNED AT RESTART**

Explanation: After completing its search through the resource manager status table and the checkpoint queue, no valid checkpoint RBA was found within the specified range.

System action: Utility processing continues, and will set a return code of 4. This message is written to the SYSPRINT data set, not to a console.

User response: The last 100 checkpoints are recorded in the BSDS, including the log STARTRBA and log ENDRBA of the checkpoint range. When either STARTRBA or ENDRBA/ENDLRSN are specified, the change log inventory (DSNJU003) utility attempts to locate a valid checkpoint in the range. In this case, DSNJU003 was unsuccessful in finding a valid checkpoint. When DB2 is restarted, the log will be scanned to locate the checkpoint prior to the log truncation point.

DSNJ408I   *csect-name* **CHECKPOINT RBA FOUND, RBA =** *rba*, **TIME =** *time*

Explanation: After completing its search through the resource manager status table and the checkpoint queue, 'rba' was the most recent checkpoint RBA in the specified range and 'time' was the time of the checkpoint.

System action: Utility processing continues.

DSNJ409I   **UNABLE TO WRITE ACTIVE LOG RECORD TO THE BSDS DATASET. DDNAME=xxxxxx**

Explanation: A program encountered an I/O error when attempting to write a BSDS active log record to the BSDS data set. 'xxxxxx' is the DDNAME of the BSDS data set being accessed.

System action: The program will terminate.

Operator response: Notify the system programmer.

System programmer response: One or more error messages describing the specific error preceded this message. This error is considered unrecoverable and will leave the BSDS data set in an uncertain state. It will be necessary to either: re-create the data set using access method service routines and then use change log inventory (DSNJU003) to make the active and archive log data set(s) known to the BSDS, or recover the BSDS using a backup copy.

DSNJ410E   *csect-name* **I/O ERROR DURING** *operation-type* **PROCESSING OF** *error-record* **RECORD**

Explanation: An attempt was made to read, write, or erase a record, but an I/O error occurred.

csect-name
> The name of the control section that issued the message.

operation-type
> The type of operation: READ, WRITE, or ERASE.

error-record
> The record that could not be processed:
> - Conditional restart control record
> - Checkpoint queue record
> - Archive log command history record
> - Resource manager status table
> - BSDS member record (BMEM)

System action: The current utility processing is terminated. Message DSNJ410E is written to the SYSPRINT data set, and is accompanied by message DSNJ212I or DSNJ213I. The message is not written to console.

User response: Use the information provided in message DSNJ212I or DSNJ213I to determine the cause of the I/O error. Correct the problem, and resubmit the job.

Related information:

"DSNJ212I" on page 263

"DSNJ213I" on page 264

DSNJ411I   *csect-name* **CRESTART** *nnnn* **FOR CRCRID =** *yyyy*, **DDNAME =** *ddname*

Explanation: A CRESTART request has just completed. 'nnnn' is either CREATE or CANCEL. 'yyyy' is the restart control record hexadecimal identifier and 'ddname' is the BSDS data set (SYSUT1 or SYSUT2) associated with the request.

System action: Current utility processing continues. This message is written to the SYSPRINT data set, not to a console.

User response: Note the record identifier for future reference.

DSNJ412E   *csect-name* **RESTART CONTROL RECORD NOT FOUND IN BSDS**

Explanation: The CRESTART CANCEL keyword was specified but the conditional restart control record does not exist in the BSDS data set.

System action: Current utility processing is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: CREATE the conditional restart control record prior to attempting the CANCEL function.

DSNJ413E   *csect-name* **INVALID LOG RANGE SCOPE OR CHECKPOINT SPECIFIED**

Explanation: The values specified thru the STARTRBA, ENDRBA, and CHKPTRBA keywords are invalid.

System action: Current utility processing is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Ensure that the log range values are correct and correspond to the other log range values either specified or defaulted. A valid log range is where STARTRBA is less than or equal to CHKPTRBA, and CHKPTRBA is less than ENDRBA. The STARTRBA most be less than or equal to the ENDRBA.

DSNJ414I   *csect-name* **COLD START WILL RESULT FROM THIS RESTART CONTROL RECORD. FORWARD AND BACKOUT SET TO NO.**

Explanation: STARTRBA and ENDRBA are equal. A cold start will result if this restart control record is used during DB2 restart. No forward or backout processing will be performed.

System action: CRESTART processing continues.

User response: No additional actions are required if a cold start of the system will be performed. If a cold start will not be performed, reissue the CRESTART and either CANCEL the current restart control record, or CREATE a new restart control record.

DSNJ415E   *csect-name* **ENDRBA =** *rba* **IS INVALID, MUST BE A MULTIPLE OF 4K**

Explanation: The specified ENDRBA at 'rba' is not a multiple of 4KB.

System action: CRESTART processing is terminated.

User response: Correct the ENDRBA value on the CRESTART statement and rerun the utility.

DSNJ416I   **WARNING - BSDS UTILITY TIMESTAMP MISMATCH DETECTED. PROCESSING CONTINUES.**

Explanation: As a result of a change log inventory (DSNJU003) update, it was discovered that the SYSUT1 BSDS and SYSUT2 BSDS timestamps are unequal. Being unequal indicates the possibility of a BSDS mismatch.

System action: Current utility processing continues.

User response: Run the print log map (DSNJU004) utility against the SYSUT1 BSDS and SYSUT2 BSDS. Determine if each BSDS is current. If each BSDS is current, this warning can be ignored. If either BSDS is not current, delete the obsolete data set and define a replacement data set, then copy the current BSDS into the replacement data set.

DSNJ417I　　REQUIRED *xxxxxxxx* PARAMETER FOR *yyyyyyyy* OPERATION IS MISSING

Explanation:　A required parameter 'xxxxxxxx' for a Log utility operation was missing from the Log utility control statement. The attempted operation is 'yyyyyyyy'.

System action:　The Log utility 'yyyyyyyy' operation does not perform its function. All subsequent Log utility control statements are processed. A nonzero return code is issued by the utility.

DB2 writes this message to the SYSPRINT data set, not to a console.

User response:　Add the missing parameter to the control statements associated with the specified operation and rerun the utility.

DSNJ418I　　WARNING - A NOTREUSEABLE ACTIVE LOG DATA SET WAS DELETED FROM BSDS LOG INVENTORY, STARTRBA=...,ENDRBA=...

Explanation:　The data set name specified on the DSNAME parameter of the change log inventory (DSNJU003) utility DELETE statement was a NOTREUSABLE active log.

System action:　DSNJU003 processing continues, and will complete with a return code of 4. This message is written to the SYSPRINT data set, not to a console.

User response:　No additional actions are required if a deletion of a NOTREUSABLE active log was intended. Otherwise, re-create the deleted log by using the NEWLOG statement with the RBA values specified in the warning message.

DSNJ419I　　UNABLE TO WRITE ARCHIVE LOG RECORD TO THE BSDS DATASET. DDNAME=*xxxxxx*

Explanation:　A program encountered an I/O error when attempting to write a BSDS archive record to the BSDS data set. 'xxxxxx' is the DDNAME of the BSDS data set being accessed.

System action:　The program will terminate.

Operator response:　Notify the system programmer.

System programmer response:　One or more error messages describing the specific error preceded this message. This error is considered unrecoverable and will leave the BSDS data set in an uncertain state. It will be necessary to either: re-create the data set using access method service routines and then use change log inventory (DSNJU003) utility to make the active and archive log data set(s) known to the BSDS, or recover the BSDS using a backup copy.

DSNJ420I　　UNABLE TO WRITE BCR1 RECORD TO BSDS DATASET. DDNAME=*xxxxxx*

Explanation:　A program encountered an I/O error when attempting to write the BSDS control record to the BSDS data set. 'xxxxxx' is the DDNAME of the BSDS data set being accessed.

Operator response:　Notify the system programmer.

System programmer response:　One or more error messages describing the specific error preceded this message. Refer to the specific message(s) for error analysis and the appropriate action to take.

DSNJ421I　　THE BSDS IS NOT AT THE PROPER RELEASE LEVEL. DDNAME=*xxxxx*

Explanation:　This message is issued when a program attempting to access The BSDS data set is not at a compatible release level. This can be one of the following:

- The program accessing the BSDS data set is at a NEW release level and the BSDS is at the old level.
- The program accessing the BSDS data set is at an OLD release level and the BSDS is at the NEW level.
- The program encountered an UNKNOWN release level on the BSDS data set.

System action:　The program will terminate with no action taken.

System programmer response:　If the program that encountered the error was one of the BSDS conversion programs (DSNJMIG0 or DSNJFAL0) an unrecoverable error was found on the BSDS data set and it must be re-created. If the error was encountered running some other utility program (i.e., change log inventory or print log map), one of the following actions may be taken:

- Run a version of the program that is compatible with the release level of the BSDS.

- Run the appropriate formatting program (DSNJMIG0 or DSNJFAL0) to convert the BSDS data set to a release level compatible with the running program. If this was successful, rerun the original program.

DSNJ422I **LOG RECORDS HAVE ALREADY BEEN FORMATTED - THIS UTILITY WILL TERMINATE**

Explanation: This message occurs when one of the BSDS formatting programs detects the BSDS had already been reformatted or was already at the required format level.

System action: The program will terminate without any action being taken.

System programmer response: If the appropriate BSDS formatting program was being run, no further action is necessary.

DSNJ423I **ACTIVE LOG MAY NOT HAVE BEEN ARCHIVED. DSN=*xxxxxx***

Explanation: The program was attempting to convert an old BSDS data set with more than 31 active log data sets to the new format. The program will drop the oldest active log data set(s) to get to the new maximum (31) limit. When dropping one of the old active log data sets from the BSDS, it was found to have been flagged as NOT REUSABLE (i.e., this data set MAY NOT have been archived). 'xxxxxx' will contain the data set name of the active log in question.

System action: The program will remove this active log data set from the BSDS and continue.

System programmer response: Check to see if this data set had been archived. The active log data set was not deleted on disk but removed only from the BSDS data set, so it may still be archived, if needed. Use access method service routines to copy the VSAM active log data set to a sequential archive log data set.

DSNJ424I *csect-name* **INVALID CONTROL STATEMENT. THE SECOND WORD MUST BE 'DDNAME' OR '*' OR MUST START WITH A PARENTHESIS.**

Explanation: The control statement is invalid.

System action: The control statement is ignored. DB2 uses the SYSUT1 DD statement to locate the BSDS to be mapped. If the SYSUT1 DD statement is not available, the utility terminates.

User response: Correct or remove the control statement and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related reference:

➩ DSNJU004 (print log map) (DB2 Utilities)

➩ Sample DSNJU004 control statement (DB2 Utilities)

DSNJ425I **INVALID *xxxxxx* VALUE/FORMAT FOR *yyyyyyyy* PARAMETER (YYYYDDDHHMMSST)**

Explanation: 'xxxxxx' is the field in the 'yyyyyyyy' parameter that contains an incorrect value or incorrect format.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ426I **ENDTIME VALUE CANNOT BE LESS THAN STARTIME VALUE**

Explanation: The 'STARTIME' and 'ENDTIME' parameters specify a time range. Hence, the ENDTIME value must be equal to or greater than STARTIME value.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ427I **CHECKPOINT RECORD ADDED TO QUEUE**

Explanation: The checkpoint record specified has been added to the checkpoint queue.

System action: The BSDS checkpoint queue is modified as stated. This message is written to the SYSPRINT data set, not to a console.

DSNJ428I **CHECKPOINT RECORD DELETED FROM QUEUE, STARTRBA=***startrba***, ENDRBA=***endrba*

Explanation: The checkpoint record specified has been deleted from the Checkpoint Queue. 'startrba' and 'endrba' was the RBA range indicated in the deleted checkpoint record.

System action: The BSDS checkpoint queue is modified as stated. This message is written to the SYSPRINT data set, not to a console.

DSNJ429I **RBA RANGE CONFLICTS WITH EXISTING CHECKPOINT RECORD RBA RANGE**

Explanation: The specified RBA range for the new checkpoint record either exists or overlaps an existing RBA range in the checkpoint queue.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Run the print log map (DSNJU004) utility against the SYSUT1 BSDS and SYSUT2 BSDS. Determine the correct RBA range and correct the STARTRBA and ENDRBA parameters and rerun the utility.

DSNJ430I **SPECIFIED ENTRY CANNOT BE ADDED WITHOUT OVERLAYING EXISTING LOWEST ENTRY**

Explanation: The specified RBA range for the new checkpoint record is less than the existing lowest entry. The checkpoint queue is currently full and cannot add the new entry without overlaying the lowest entry.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Run the print log map (DSNJU004) utility against the SYSUT1 BSDS and SYSUT2 BSDS. Determine the lowest existing entry and either change the STARTRBA and ENDRBA parameters or delete the existing lowest entry and add the new low checkpoint entry and rerun the utility.

DSNJ431I **STARTRBA SPECIFIED NOT FOUND IN CHECKPOINT QUEUE**

Explanation: The specified STARTRBA could not be located in the checkpoint queue.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Run the print log map (DSNJU004) utility against the SYSUT1 BSDS and SYSUT2 BSDS. Determine the correct STARTRBA value and correct the STARTRBA parameter and rerun the utility.

DSNJ432I **THE OFFLRBA VALUE MUST END WITH 'FFF'**

Explanation: The OFFLRBA value is invalid. It must end with 'FFF'.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ433I *csect-name* **INVALID CONTROL STATEMENT. A MEMBER LIST MUST CONTAIN AT LEAST ONE MEMBER NAME.**

Explanation: The control statement is invalid. The member list is either empty or in the wrong format.

System action: The control statement is ignored. DB2 uses the SYSUT1 DD statement to locate the BSDS to be mapped. If the SYSUT1 DD statement is not available, the utility terminates.

User response: Correct or remove the control statement and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related reference:

DSNJU004 (print log map) (DB2 Utilities)

Sample DSNJU004 control statement (DB2 Utilities)

DSNJ434I   *csect-name* **SYSIN MUST CONTAIN AT LEAST ONE RECORD.**

Explanation: A SYSIN DD statement was provided, but the data set was null, or the JCL statement "//SYSIN DD *" was immediately followed by the statement "/*".

System action: The error is ignored. DB2 uses the DD SYSUT1 statement to locate the BSDS to be mapped. If the SYSUT1 DD statement is not available, the utility terminates.

User response: Correct or remove the control statement and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related reference:

DSNJU004 (print log map) (DB2 Utilities)

Sample DSNJU004 control statement (DB2 Utilities)

DSNJ435I   *csect-name* **THE MEMBER LIST OF A CONTROL STATEMENT MUST BE TERMINATED WITH A PARENTHESIS.**

Explanation: The control statement contained the correct format for starting a list of members, but the terminating parenthesis was not found.

System action: The control statement is accepted with an assumed parenthesis after the last member name recognized, and processing continues.

User response: Correct or remove the control statement and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related reference:

DSNJU004 (print log map) (DB2 Utilities)

Sample DSNJU004 control statement (DB2 Utilities)

DSNJ436I   *csect-name* **THE DD STATEMENT FOR** *ddname* **IS MISSING.**

Explanation: The print log map (DSNJU004) utility requested information about the identified *ddname* from the MVS Dynamic Allocation facility, but the DD statement was not found in the job.

System action: The job is terminated.

User response: Correct the JCL and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information

about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Related concepts:

➦ Invoking stand-alone utilities (DB2 Utilities)

Related reference:

➦ DSNJU004 (print log map) (DB2 Utilities)

DSNJ437I   *csect-name* **UNABLE TO ALLOCATE BSDS** *dsname*. **SVC 99 RTN CODE** *rc*, **RSN CODE** *rsncd*, **INFO** *infocd*.

Explanation: The print log map (DSNJU004) utility attempted to allocate a BSDS data set using the MVS Dynamic Allocation facility, but the system was unable to complete the allocation. The return code, reason code, and information code were returned by the system's SVC 99 facility. See *MVS/ESA Programming: Authorized Assembler Services Guide* for an explanation of these codes.

System action: The processing of the BSDS named in the message is terminated. If this is the last (or only) BSDS to be processed by this job, the job is terminated.

User response: Correct the JCL and rerun the job.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

.

Related concepts:

➦ Invoking stand-alone utilities (DB2 Utilities)

Related reference:

➦ DSNJU004 (print log map) (DB2 Utilities)

DSNJ438I   *csect-name* **THE** *xxx* **VALUE MUST END WITH** *'yyy'*

Explanation: The STARTRBA or ENDRBA value specified on the NEWLOG statement is invalid. *'xxx'* is either STARTRBA or ENDRBA, depending on which parameter is invalid. If *'xxx'* equals STARTRBA, then *'yyy'* equals *'000'* and if *'xxx'* equals ENDRBA, then *'yyy'* equals *'FFF'*.

System action: The current utility is terminated. This message is written to the SYSPRINT data set, not to a console.

User response: Correct the control statement and rerun the utility.

DSNJ439I   *csect_name* **NEW FUNCTION MODE REQUIRED FOR** *xxxxxx* **OPERATION**

Explanation: An attempt was made to run a log utility requiring new-function mode while DB2 was in conversion mode or enabling-new-function mode.

DB2 must be in new-function mode before the utility program can run successfully. *xxxxxx* is the name of the attempted operation requiring new-function mode.

System action: The program will terminate with no action taken.

System programmer response: Convert DB2 to new-function mode and retry the program.

DSNJ440I   BSDS HAS ALREADY BEEN CONVERTED, DDNAME=*dd-name*

Explanation: This message occurs when a BSDS conversion program detects that the BSDS data set that was specified by *dd-name* was already converted.

System action: If only SYSUT1 was specified, or both SYSUT1 and SYSUT2 data sets were already converted, the program terminates with no action taken. In the case where both are specified, but only SYSUT1 was previously converted, the program continues to process the conversion for SYSUT2.

System programmer response: No further action is necessary.

Related reference:

DSNJCNVB (DB2 Utilities)

DSNJCNVT (DB2 Utilities)

DSNJ441I   INVALID BSDS CONVERSION, DDNAME=*dd-name*

Explanation: This message is issued when a program, attempting to access the BSDS data set, encounters an invalid BSDS. An invalid BSDS is the result of a failure during a prior attempt to run a conversion program.

dd-name
        The DDNAME of the BSDS data set being accessed.

System action: The program terminates with no action taken.

System programmer response: The procedure for running the program includes making a copy of the BSDS before conversion. Restore the BSDS to the original pre-conversion copy, and try the conversion again. After the conversion is successful, try the program that issued the error message again.

Related reference:

DSNJCNVB (DB2 Utilities)

DSNJCNVT (DB2 Utilities)

DSNJ442I   *csect-name* RECEIVED ERROR STATUS *nnn* FROM *macro-name*

Explanation: DB2 encountered an error while issuing macro *macro-name*. Error status is the return code from the specified macro.

System action: The requested operation fails.

User response: If the error is from the CONVTOD macro, ensure that the ENDTIME timestamp value is valid and in the correct format.

DSNJ443I   *csect-name* ENDTIME LOG TRUNCATION POINT = *xxxxxxxxxxx*

Explanation: The ENDTIME timestamp value results in a log truncation at LRSN or STCK value *xxxxxxxxxxx*.

System action: Utility processing continues.

DSNJ444I   *csect-name* ENDTIME TIMESTAMP IS GREATER THAN THE CURRENT TIMESTAMP

Explanation: The ENDTIME timestamp must be a system time-of-day clock timestamp for which there exists a log record with a timestamp that is greater than, or equal to, the specified timestamp value.

System action: The utility terminates.

User response: Specify an ENDTIME timestamp value that is valid for log truncation.

DSNJ445I   *action-type* REQUEST FAILED: *error-description*

Explanation: A change log inventory utility (DSNJU003) operation failed.

action-type
        The DSNJU003 operation: DELMBR or RSTMBR.

DSNJ445I

error-description
> A description of the error that occurred:

System action:

CRESTART FAILED: SYSPITR LRSN MUST BE GREATER THAN *minimum-value* **OF LAST MEMBER DESTRUCTION**
> An attempt was made to use a CRESTART statement to create a conditional restart record for system point in time recovery. However, the CRESTART statement specified a log truncation point LRSN that is prior to the most recent member destruction. The logs of the destroyed member are no longer available. The minimum LRSN value is indicated in the message as *minimum-value*
>
> **User response:** Specify a LRSN that is greater than the minimum value indicated in the message.

CRESTART FAILED: SYSPITRT TIMESTAMP MUST BE GREATER THAN *minimum-value* **OF LAST MEMBER DESTRUCTION**
> An attempt was made to use a CRESTART statement to create a conditional restart record for system point in time recovery. However, the CRESTART statement specified a log truncation point timestamp that is prior to the most recent member destruction. The logs of the destroyed member are no longer available. The minimum timestamp value is indicated in the message as *minimum-value*.
>
> **User response:** Specify a timestamp that is more recent than the minimum value indicated in the message.

DEACTIV ON A MEMBER ALREADY DEACTIVATED
> DELMBR DEACTIV was requested for a member that has already completed deactivation processing.

DEACTIV ON A MEMBER ALREADY DESTROYED
> DELMBR DEACTIV was requested for a member that has already completed destruction processing.

DEACTIV ON A MEMBER ALREADY MARKED FOR DEACTIVATION
> DELMBR DEACTIV was requested for a member that has already been marked for deactivation, but the deactivation process has not yet completed.
>
> **System action:** The state of the member is unchanged.

DEACTIV ON A MEMBER ALREADY MARKED FOR DESTRUCTION
> DELMBR DEACTIV was requested for a member that has already been marked for destruction by a previous DESTROY request.

DEACTIV ON A NON-EXISTENT MEMBER IN NON-DATA SHARING
> An attempt was made to deactivate a member from the BSDS of a non-data sharing DB2 subsystem. Deactivation is allowed in data sharing mode only.
>
> **User response:** Verify the member ID.

DESTROY ON A MEMBER ALREADY DESTROYED
> DELMBR DESTROY was requested for a member that has already completed destruction processing.

DESTROY ON A MEMBER ALREADY MARKED FOR DESTRUCTION
> DELMBR DESTROY was requested for a member that has already been marked for destruction but the destruction process has not yet completed.
>
> **System action:** The state of the member is unchanged.

DESTROY ON A NON-EXISTENT MEMBER IN NON-DATA SHARING
> An attempt was made to destroy a member from the BSDS of a non-data sharing DB2 subsystem. Destruction is allowed in data sharing mode only.
>
> **User response:** Verify the member ID.

INVALID AFTER A NON-DELMBR OR NON-RSTMBR OPERATION

A control statement other than DELMBR or RSTMBR preceded the current DELMBR or RSTMBR statement. The statement is not allowed if the BSDS being processed belongs to the target member. The DELMBR or RSTMBR statement is not processed.

User response: Restart the member that owns the BSDS so that previous changes to the BSDS changes can be processed.

INVALID AFTER DELMBR OR RSTMBR OPERATION

A control statement other than DELMBR or RSTMBR was issued for a data sharing group member that is queued for deletion. When DELMBR or RSTMBR is input to DSNJU003, no other operations are allowed if the BSDS that is being processed belongs to the target member.

System action: The other control statements are not processed. If the BSDS that belongs to the target member has been marked as deactivated or destroyed by a previous invocation, no further updates to the BSDS are allowed unless the target member was deactivated and a RSTMBR operation is requested.

INVALID WITH HOST MEMBER IN DEACTIV/DESTROY STATE

A control statement other than RSTMBR was issued for a data sharing group member that has been deactivated. When a data sharing group member has been deactivated, the only control statement that DSNJU003 accepts for that member's BSDS is RSTMBR. When a member is marked as destroyed in its own BSDS, DSNJU003 rejects all statements that would alter the BSDS contents.

MEMBER ALREADY DESTROYED

RSTMBR has been requested for a member that has been destroyed. RSTMBR can be used only when a member has been deactivated or marked for deactivation.

MEMBER ALREADY MARKED FOR DESTRUCTION

RSTMBR has been requested for a member that has been marked for destruction but not yet completely destroyed. RSTMBR can be used only when a member has been deactivated or marked for deactivation.

User response: If a backup copy of the BSDS is available from before DELMBR DESTROY was used, restore the backup.

MEMBER HAS NOT BEEN DEACTIVATED

Either RSTMBR has been requested for a member that is not deactivated or marked for deactivation or DELMBR DESTROY was requested for a member that has not been deactivated or marked for deactivation. A DELMBR DESTROY can immediately follow a DELMBR DEACTIV for the same member ID, but a DELMBR DEACTIV is always required first.

User response: Either complete the deactivation process or use DELMBR DEACTIV followed by DELMBR DESTROY for the target member.

MEMBER HAS NOT BEEN QUIESCED

DELMBR was requested for a member that has not been quiesced or was active when the member that owns the BSDS was shut down. A member must be quiesced before a DELMBR command will be accepted.

User response: Quiesce the target member. To update the state of the target member, restart the member that owns the BSDS that DSNJU003 was running against.

NON-EXISTENT MEMBER IN NON-DATA SHARING

RSTMBR was requested for a nonexistent member and the BSDS indicates the member that owns the BSDS is not part of a data sharing group. In a non-data sharing environment, RSTMBR can be used only for a known member that has been marked for deactivation through the DELMBR DEACTIV statement.

User response: Indicate a different member number.

UTILITY TIMESTAMP GREATER THAN SYSTEM TIMESTAMP. MEMBER RESTART REQUIRED.

DELMBR was requested, but failed because an earlier change to the BSDS updated the utility timestamp. The member must be restarted before DELMBR can be used.

> **User response:** Restart the member that owns the BSDS before using DELMBR DEACTIV.

VALID MEMBER ID IS 1 TO 32 DECIMAL
> An invalid member ID was specified. The valid range is 1 to 32, inclusive.

> **User response:** Specify a member ID between 1 and 32.

System action: The change log inventory utility terminates. Message DSNJ445I is written to the SYSPRINT data set.

Additional actions occur for some *error-description* values, and are described with the explanation of the value.

User response: Take the action indicated for the *error-description* value.

Related reference:

DSNJU003 (change log inventory) (DB2 Utilities)

DSNJ446I   *operation-type* **SUCCESSFUL FOR MEMBER ID** *member-id*, **MEMBER NAME** *member-name*

Explanation: The change log inventory utility (DSNJU003) successfully completed a DELMBR or RSTMBR request.

operation-type
> The type of operation: DELMBR DEACTIV, DELMBR DESTROY, or RSTMBR

member-id
> The ID number of the member that was the target of the request.

member-name
> The name of the member that was the target of the request.

System action: Processing completes. Message DSNJ446I is written to the SYSPRINT data set.

User response: No action is required.

Related reference:

DSNJU003 (change log inventory) (DB2 Utilities)

DSNJ447I   **KEYWORD** *required-keyword* **IS REQUIRED WHEN KEYWORD** *specified-keyword* **IS SPECIFIED**

Explanation: A change log inventory utility (DSNJU003) control statement specified a keyword that requires an additional keyword that was not specified.

required-keyword
> The name of the keyword that is required when *specified-keyword* is used.

specified-keyword
> The name of the keyword that was specified. This keyword cannot be used without *required-keyword*.

System action: The statement is ignored. The utility terminates with RC=4. Message DSNJ447I is written to the SYSPRINT data set.

User response: Correct the syntax of the control statement, and resubmit the utility job.

Related reference:

DSNJU003 (change log inventory) (DB2 Utilities)

DSNJ448I   **ERROR DURING BSDS CONVERSION.** *error-condition*

Explanation: An error occurred during BSDS format conversion.

error-condition
> Text that describes the error that occurred.

> **OUTPUT BSDS CI SIZE NOT CORRECT, DDNAME=***dd-name*
> > The CI size of the output BSDS should be 8192.

> > *dd-name*
> > > The JCL DD name of the data set with the incorrect CI size. Delete and redefine the output BSDS.

log-type **LOG DATA SET** *dataset-name* **HAS AN ASSOCIATED PASSWORD**
DB2 does not support active or archive log data sets with passwords defined.

log-type   Indicates whether an active log or archive log data set has a password.

dataset-name
The name of the data set with the password.

INPUT BSDS NOT IN CORRECT FORMAT, DDNAME=*dd-name*

dd-name
Specifies the data set that was not in the format that supports 93 active logs and 10,000 archive logs. If you have not yet run DSNJCNVB against the BSDS, run DSNJCNVB first and then run DSNJCNVT.

INPUT AND OUTPUT BSDS ARE THE SAME. INPUT=*input-name*, **OUTPUT=***output-name*
The converted BSDS cannot be the same as the BSDS to be converted.

input-name
The JCL input name of the data set.

output-name
The JCL output name of the data set.

UNRECOGNIZED BSDS RECORD, KEY=*key-value*
During conversion of the BSDS, a record was found that is not a known format.

key-value
The VSAM KSDS key of the BSDS record that was not recognized. IDCAMS PRINT that specifies the key value that is listed can be used to determine what inserted the record to the BSDS. If the record is not needed, delete it and rerun the BSDS conversion

RETURN CODE *return-code*, **REASON CODE** *reason-code*
An error condition was found. For more information, see the reason code explanation.

System action: The program terminates. The output BSDS is not usable. The input BSDS is unaltered.

Programmer response: Correct the error condition and rerun the conversion program.

Related reference:

➡ DSNJCNVT (DB2 Utilities)

DSNJ700I   *csect_name* **ERROR ATTEMPTING TO ACCESS BSDS FOR PEER MEMBER ID=** *member-id*, **MEMBER NAME=** *member-name*, **REASON-CODE=** *reason-code*.

Explanation: DB2 issues this message to identify the peer member ID, and if known, the peer member name associated with listed DB2 reason code.

System action: An execution unit issues this message and then, if this was an unconditional request, writes a record to SYS1.LOGREC and requests an SVC dump. The execution unit then terminates.

Operator response: Collect all relevant diagnostic materials, including SYS1.LOGREC, SVC dump, and console output of peer and host.

System programmer response: Review the explanation for the associated DB2 reason code for further action.

DSNJ701E   *csect-name* **MEMBER NAME (***membername***) IN BSDS DOES NOT MATCH MEMBER NAME (***membername***) IN DB2 SYSTEM PARAMETERS** *dsnzparm*

Explanation: During startup, DB2 detected that the member name listed in its bootstrap data set (BSDS) does not match the member name in the DB2 system parameters load module.

Operator response: Notify the system programmer.

System programmer response: Restore the BSDS or specify a different system parameters load module on the START DB2 command.

DSNJ702E   *csect_name* **GENERIC LU NAME** *lu-name* **IN BSDS DOES NOT MATCH THE GENERIC LU NAME** *lu-name* **ASSOCIATED WITH THE DATA SHARING GROUP.**

Explanation: DB2 issues this message during startup when it detects that the generic LU name in the bootstrap data set (BSDS) does not match the generic LU name in the shared communications area (SCA) structure.

System action: DB2 startup continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

System programmer response: Depending on the accuracy of the BSDS value, take one of the following actions:
- BSDS value is correct: DB2 startup will automatically replace the generic LU name in the SCA with the correct value. The BSDS for each member of this DB2 data sharing group must specify this same value for generic LU name.If any member specifies a different value for generic LU name, use the change log inventory utility (DSNJU003) to change the generic LU name.
- BSDS value is not correct: DB2 startup will replace the generic LU name in the SCA. The SCA value will be incorrect when startup is complete. Take the following actions to correct this situation:
  - Stop this DB2 subsystem.
  - Use the change log inventory utility (DSNJU003) to change the generic LU name of this member.
  - Issue START DB2 for this member. START DB2 will correct the generic LU name contained in the SCA.

Related reference:

↪ DSNJU003 (change log inventory) (DB2 Utilities)

DSNJ703E   *csect-name* **GROUP NAME (***groupname***) IN BSDS DOES NOT MATCH GROUP NAME (***groupname***) IN DB2 SYSTEM PARAMETERS** *dsnzparm*

Explanation: During startup, DB2 detected that the group name listed in its bootstrap data set (BSDS) does not match the group name in the DB2 system parameters load module.

Operator response: Notify the system programmer.

System programmer response: Restore the BSDS or specify a different system parameters load module on the START DB2 command.

DSNJ704I   *csect-name* **NEW MEMBER(***member-name***) START-UP NOT ALLOWED WHEN GROUP-RESTART IS REQUIRED FOR GROUP(***group-name***).**

Explanation: One of the following occurred:
- A new member *member-name* tried to start when a group restart of data sharing group *group-name* is required.
- A member other than the originating member of a data sharing group attempted to start the new DB2 data sharing group. The originating member of a group is either an existing DB2 subsystem which formed the data sharing group or a new DB2 subsystem which was installed to form the group.

System action: The DB2 subsystem stops.

Operator response: Start a previously active member of this group. When the required group restart is completed, start the new member again.

If this is the very first startup of the group, you must start the originating member first before starting any other member.

DSNJ705I   *csect-name* **THE DB2 SYSTEM PARAMETERS SPECIFY DATA SHARING, BUT THE BSDS SPECIFIES NON-DATA SHARING.**

Explanation: The system parameters passed on the DB2 start command specified data sharing, but the bootstrap data set (BSDS) did not indicate this DB2 is a data sharing member.

System action: The DB2 subsystem stops.

Operator response: If this DB2 should be a member of a data sharing group, inform the system programmer. If this DB2 is not a member of a data sharing group, restart DB2 with the correct system parameters load module

System programmer response: If this DB2 should be a member of a data sharing group, execute the installation job DSNTIJUZ, which updates the BSDS for data sharing.

If this DB2 is not a member of a data sharing group, reassemble the system parameters load module and specify DSHARE=NO.

If this DB2 is attempting to disable the data sharing group, then reassemble the system parameters load module and specify DSHARE=NO, then execute the installation job DSNTIJGF, and then restart this DB2. Job DSNTIJGF must be used with caution, this will disable the data sharing group.

Related tasks:

➡ Disabling data sharing (DB2 Installation and Migration)

➡ Migration step 11: Define DB2 initialization parameters: DSNTIJUZ (DB2 Installation and Migration)

➡ Installation step 5: Define DB2 initialization parameters: DSNTIJUZ (DB2 Installation and Migration)

Related reference:

➡ DATA SHARING field (DSHARE subsystem parameter) (DB2 Installation and Migration)

DSNJ706I   *csect-name* **THE DB2 SYSTEM PARAMETERS SPECIFY NON-DATA SHARING, BUT THE BSDS SPECIFIES DATA SHARING.**

Explanation: The system parameters passed on the DB2 START command specified non-data sharing, but the bootstrap data set (BSDS) indicates this is a data sharing member.

System action: The DB2 subsystem stops.

Operator response: If this DB2 is a member of a data sharing group, restart DB2 with the correct system parameters load module. If this DB2 is not a member of a data sharing group, inform the system programmer.

System programmer response: If this DB2 should be a member of a data sharing group, reassemble the system parameters load module and specify DSHARE=YES. If this DB2 is not a member of a data sharing group, execute the installation job DSNTIJGF, which updates the BSDS for non-data sharing.

DSNJ707E   *csect_name* **LOCATION NAME** *location-name* **IN BSDS DOES NOT MATCH THE LOCATION NAME** *location-name* **ASSOCIATED WITH THE DATA SHARING GROUP.**

Explanation: DB2 issues this message during startup when it detects that the LOCATION name in the bootstrap data set (BSDS) does not match the LOCATION name in the system communications area (SCA) structure.

System action: DB2 startup continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Notify the system programmer.

System programmer response: Depending on the accuracy of the BSDS value, take one of the following actions:
- BSDS value is correct: DB2 startup will automatically replace the LOCATION name in the SCA with the correct value. The BSDS for each member of this DB2 data sharing group must specify this same value for LOCATION. If any member specifies a different value for LOCATION, use the change log inventory utility (DSNJU003) to change the LOCATION name.
- BSDS value is not correct: DB2 startup will replace the LOCATION name in the SCA. The SCA value will be incorrect when startup is complete. Take the following actions to correct this situation:
  – Stop this DB2 subsystem.
  – If any member specifies a different value for LOCATION, use the change log inventory utility (DSNJU003) to change the LOCATION name.
  – Issue START DB2 for this member. START DB2 will correct the LOCATION name contained in the SCA.

Related concepts:

⇨ Update the BSDS with the DSNJU003 utility (DB2 Data Sharing Planning and Administration)

Related reference:

⇨ DSNJU003 (change log inventory) (DB2 Utilities)

DSNJ708I   *csect-name* **NEW MEMBER START-UP IS NOT ALLOWED WHEN THE DATA SHARING GROUP IS IN SYSTEM RECOVER PENDING MODE**

Explanation: A new data sharing member attempted to join the data sharing group when the data sharing group is in System Recover Pending mode.

System action: DB2 startup terminates.

Operator response: When the data sharing group is no longer in System Recover Pending mode, start the new member again.

DSNJ709I   *csect-name RBA/LRSN value* **WILL BE NEAR THE END OF THE LOG RANGE. REPLY Y TO CONTINUE, N TO CANCEL**

Explanation: This message is issued during enable, disable, or re-enable of data sharing mode if the resulting RBA or LRSN value is close enough to the end of the log range that normal SQL processing is not allowed.

csect-name
   The name of the control section that issued the message.

RBA/LRSN
   If you are disabling data sharing, the string RBA is displayed. Otherwise, the string LRSN is displayed.

value   The value of the RBA or LRSN if the operation continues.

System action: If Y is entered, the operation proceeds. If N is entered, the restart is terminated.

Operator response: Consult with the system programmer or database administrator to determine the correct reply.

System programmer response: Determine whether the RBA or LRSN value is acceptable. When the RBA or LRSN log is nearing the end of the RBA or LRSN range, normal SQL updates fail with resource unavailable errors for objects that are in basic 6-byte format. Convert database objects to extended 10-byte format before the end of the log range is reached.

DSNJ711E   *csect-name* **SCA OPEN FAILURE ENCOUNTERED.**

Explanation: A failure occurred during command processing in an attempt to open an object in the SCA.

System action: Command processing terminates unsuccessfully.

System programmer response: Obtain SYS1.LOGREC and possible SVC dump for analysis.

Problem determination: Review the DB2 system console for possible DSN7xxxx and/or SVC dump messages that could relate to this problem.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNJ712E   *csect-name* **SCA READ FAILURE ENCOUNTERED.**

Explanation: A failure occurred during command processing attempting to read from the SCA.

System action: Command processing terminates unsuccessfully.

System programmer response: Obtain SYS1.LOGREC and possible SVC dump for analysis.

Problem determination: Review the DB2 system console for possible DSN7xxxx and/or SVC dump messages that could relate to this problem.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNJ713E   *csect-name* **SCA WRITE FAILURE ENCOUNTERED.**

Explanation: A failure occurred during command processing attempting to write to the SCA.

System action: Command processing terminates unsuccessfully.

System programmer response: Obtain SYS1.LOGREC and possible SVC dump for analysis.

Problem determination: Review the DB2 system console for possible DSN7xxxx and/or SVC dump messages that could relate to this problem.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNJ990I   *csect-name* **SHOWCB ERROR, DDNAME =** *dd-name* **RETURN CODE =** *error-code* **REASON CODE =** *reason-code*

Explanation: SHOWCB failed with *error-code* and *reason-code* when processing the data set that is associated with the *dd-name*.

System action: Processing terminates.

System programmer response: Follow standard VSAM diagnostic procedure. Refer to *z/OS DFSMS: Macro Instructions for Data Sets*. If the log data set was allocated greater than 4 gigabytes in size, reallocate the data set less than 4 gigabytes in size.

User response: Notify the system programmer.

Severity: 8 (error)

DSNJ991I   *csect-name* **START OF LOG DATA SET PREFORMAT FOR JOB** *job-name step-name*

Explanation: The active log data set preformat utility has started.

Severity: 0 (informational)

DSNJ992I   *csect-name* **DDNAME =** *dd-name* **LOG DATA SET NAME =** *data-set-name*

Explanation: The indicated *data-set-name* is to be preformatted.

The DDNAME *dd-name* was used to process the data set. This message is issued multiple times if multiple active log data sets are being preformatted.

Severity: 0

DSNJ993I   *csect-name* **LOG DATA SET IS NOT A VSAM DATA SET**

Explanation: The input log data set is not a VSAM data set.

System action: Processing terminates.

User response: Use the AMS IDCAMS Service to define the active log data set as a VSAM Linear data set.

Severity: 8 (error)

DSNJ994I   *csect-name* **VSAM OPEN ERROR, DDNAME =** *dd-name* **ACBERRFLG =** *error-code*

Explanation: VSAM OPEN failed with the indicated ACB *error-code* for the indicated *dd-name*.

System action: Processing terminates if the error code is greater than or equal to 128; otherwise, processing continues normally.

System programmer response: Follow standard VSAM diagnostic procedure. Refer to *z/OS DFSMS: Macro Instructions for Data Sets*. See also *MVS/DFP V3R3 Diagnosis Guide* for more information about the VSAM procedures.

User response: Notify the system programmer.

Severity: 8 (error) if error is greater than or equal to 128; otherwise, 4 (warning)

| **DSNJ995I** *csect-name* **VSAM PUT ERROR, DDNAME =** *dd-name*, **RPLERREG =** *return-code*, **RPLERRCD =**
| *reason-code*

| **Explanation:** VSAM PUT failed with the indicated *return-code* and *reason-code* for the listed *dd-name*.

System action: Processing terminates.

System programmer response: Follow standard VSAM diagnostic procedure. Refer to *z/OS DFSMS: Macro Instructions for Data Sets*. See also *MVS/DFP V3R3 Diagnosis Guide* for more information about the VSAM procedures.

User response: Notify system programmer.

Severity: 8 (error)

| **DSNJ996I** *csect-name* **LOG PREFORMAT COMPLETED SUCCESSFULLY,***nn* **RECORDS FORMATTED FOR**
| **DDNAME** *dd-name*

| **Explanation:** The active log data set was preformatted successfully. *nn*records (or VSAM control intervals) were
| formatted for the indicated *dd-name*. Multiple messages are issued if multiple log data sets are being preformatted
| concurrently.

Severity: 0

DSNJ997I *csect-name* **LOG PREFORMAT TERMINATED**

Explanation: Preformatting of the active log data set did not complete successfully.

System action: Processing terminates.

User response: Examine other error messages associated with this job.

Severity: 8 (error)

| **DSNJ998I** *csect-name* **LOG DATA SET IS NOT EMPTY, DDNAME =** *dd-name*

| **Explanation:** The input log data set that is designated by *dd-name* is not an empty data set. The input data set must
| be empty to prevent the loss of actual log data.

System action: Processing terminates.

User response: Use the AMS IDCAMS Service to delete and redefine the active log data set or correct the input log data set name.

Severity: 8 (error)

Chapter 9. DSNL messages

Message identifiers that begin with DSNL communicate information about the distributed data facility.

DSNL001I   **DDF IS ALREADY STARTED**

Explanation: The distributed data facility (DDF) was already active when you issued -START DDF command.

System action: The issued command is not executed.

DSNL002I   **DDF IS ALREADY STOPPED**

Explanation: The distributed data facility (DDF) had already been stopped when the -STOP DDF command was issued.

System action: The issued command is not executed.

DSNL003I   **DDF IS STARTING**

Explanation: The distributed data facility (DDF) -START command is starting.

System action: DB2 begins processing the -START DDF command.

DSNL004I   *csect-name* **DDF START COMPLETE LOCATION** *location-name* **LU** *network-name.luname* **GENERICLU** *network-name.generic-luname* **DOMAIN** *domain-name* **TCPPORT** *incoming-port* **SECPORT** *secure-port* **RESPORT** *resynch-port* **IPNAME** *ip-name* **OPTIONS: PKGREL =** *pkgrel-value*

Explanation: The distributed data facility (DDF) started.

csect-name
> The name of the control section that issued the message.

location-name
> The location of the DB2 subsystem.

network-name.luname
> The fully qualified LU name for the DB2 subsystem.
>
> The value '- NONE' indicates that the DB2 subsystem has not activated its SNA/APPC communications support.

network-name.generic-luname
> The fully qualified VTAM generic LU name for the DB2 subsystem.
>
> The value '-NONE' indicates that no VTAM generic LU name was defined to be used by the DB2 subsystem or that SNA was not activated.

domain-name

> For non-data sharing systems, the member-specific domain name that maps to the member-specific IP address.
>
> For data sharing systems, the group domain name that maps to the group IP address.
>
> The domain name can be specified in the DNS or a local host file.
>
> The value '-NONE' indicates that TCP/IP is not supported, TCP/IP initialization failed, or DB2 has been started in restricted access mode.

incoming-port
> The TCP/IP port number that is used by this DB2 subsystem to accept incoming DRDA® connection requests.

DSNL004I

- A value of 0 indicates that TCP/IP is not supported. The BSDS DDF record is not configured with a PORT.
- A value of '-NONE' indicates that DB2 will not accept connections on this port. This condition can be caused by a value of 0 for the MAXDBAT subsystem parameter.

secure-port

The TCP/IP port number that is used by this DB2 subsystem to accept incoming secure DRDA connection requests.

A value of 0 indicates that the BSDS DDF record is not configured with a SECPORT.

resynch-port

The TCP/IP port number that is used by this DB2 subsystem to accept incoming two-phase commit resynchronization requests.

A value of 0 indicates that TCP/IP is not supported. The BSDS DDF record is not configured with a RESPORT.

ip-name The name given to this DB2 subsystem for use as the network ID portion of the unit of work identifier.

- A value of '-NONE' indicates that either the BSDS DDF record does not have a value for IPNAME or that DB2 is not running in new-function-mode.
- Any other value indicates that DDF has activated TCP/IP communications support in such a way that the value of *network-name.luname* will be '-NONE'.

pkgrel-value

The current value of the DDF PKGREL processing option. Possible values are:

BNDOPT

Packages are allocated based on the RELEASE bind option value.

If a database access thread (DBAT) has processed RELEASE(DEALLOCATE) packages at some point, then the thread will remain active with its connection when a unit of work finishes. If the DBAT was to have been pooled and the connection made inactive, an accounting record is made and the current WLM enclave is deleted.

Access threads that are not actively processing any units of work might continue to hold package allocation locks while waiting for the next request.

COMMIT

Packages are always allocated under RELEASE(COMMIT) rules regardless of their package RELEASE bind option value.

When an access thread has completed a unit of work, and the DBAT can be pooled, DDF pools the DBAT, makes an accounting record, deletes the current WLM enclave, and inactivates the connection. Pooled access threads release all package allocation locks.

If the CMTSTAT subsystem parameter is set to ACTIVE, then the message will always indicate that PKGREL IS SET TO COMMIT, regardless of the value specified on the command.

System action: The DDF start up process completed. However, it might fail for reasons specified by messages DSNL013I, DSNL015I, DSNL016I, DSNL017I, DSNL018I, DSNL019I, DSNL700I, or DSNL705I.

System programmer response: Use the recommended domain name to configure network definitions for TCP/IP requesters.

Related concepts:

Distributed data facility (Introduction to DB2 for z/OS)

Related tasks:

Starting DDF (DB2 Administration Guide)

Related reference:

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

RELEASE bind option (DB2 Commands)

-START DDF (DB2) (DB2 Commands)

DSNL005I   **DDF IS STOPPING**

Explanation: The distributed data facility (DDF) -STOP command has been initiated.

System action: The -STOP DDF command is initiated.

DSNL006I   **DDF STOP COMPLETE**

Explanation: The distributed data facility (DDF) -STOP command has completed.

System action: The DDF has stopped.

DSNL007I   **DDF IS ABNORMALLY TERMINATING**

Explanation: Abnormal termination of the distributed data facility (DDF) has started.

System action: Abnormal termination of the DDF is starting.

DSNL008I   **DDF ABNORMAL TERMINATION COMPLETE**

Explanation: Abnormal termination of the distributed data facility (DDF) has completed.

System action: The DDF has stopped.

DSNL009I   **INVALID DDF THREAD** *logical-unit-of-work-id* **SPECIFIED**

Explanation: No distributed transaction with the specified 'logical-unit-of-work-id' exists.

System action: The specified thread is not canceled.

User response: Determine the correct 'logical-unit-of-work-id' of the distributed thread that is to be canceled and then reissue the command.

Problem determination: Determine the correct value. The -DISPLAY THREAD command can be used to display all distributed activity.

DSNL010I   **DDF THREAD** *luw-id* **HAS BEEN CANCELED**

Explanation: The distributed data facility (DDF) thread with the specified logical unit of work ID was canceled.

luw-id   The logical unit of work ID.

In the following cases, more than one DDF thread might be canceled:
- If the logical unit of work ID is an LU network name and an LUW instance number, instead of an LUWID token, and an application accesses the same DB2 server by different access paths (for example, A->B->C and A->D->C). In this case, multiple threads with the same LUWID exist at the server (for example, C). If CANCEL DDF THREAD is issued at the server, specifying this LUWID, every thread at the server with the LUWID is canceled. This case occurs rarely.

| • If the logical unit of work ID is an LUWID token, and multiple server threads have the same logical unit of work
| ID. This happens when threads are processing secondary connections in support of an application that uses
| package-based continuous block fetch. All server threads that are associated with the secondary connections are
| canceled.

| **System action:** The specified thread is canceled.

DSNL011I   VTAM STOPPING DDF

Explanation: The operator issued a HALT NET command to terminate VTAM.

System action: The distributed data facility (DDF) -STOP command is initiated.

DSNL012I   VTAM TERMINATING DDF

Explanation: The operator issued a HALT NET QUICK command or a HALT NET CANCEL command, or VTAM is
terminating abnormally. This message can be issued any time after -START DDF and occurs when VTAM is stopped,
which can occur prior to DDF termination. This message is not issued when DDF terminates normally.

System action: Termination of the distributed data facility is initiated.

DSNL013I   VTAM OPEN ACB FAILED ERROR=*error-code*

Explanation: VTAM returned a nonzero return code after DB2 invoked the OPEN ACB macro instruction during
termination of the distributed data facility (DDF). VTAM return codes for OPEN error conditions are defined in *z/OS
Communications Server SNA Programming*.

System action: Termination of the distributed data facility is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the OPEN ACB failure, correct the error, and issue the
-START DDF command.

Problem determination: Refer to *z/OS Communications Server SNA Programming* for failure analysis during an OPEN
macro instruction.

DSNL014I   VTAM CLOSE ACB FAILED ERROR=*error-code*

Explanation: VTAM returned a nonzero return code after DB2 invoked the CLOSE ACB macro instruction during
termination of the distributed data facility (DDF). VTAM return codes for CLOSE error conditions are defined in *z/OS
Communications Server SNA Programming*.

System action: Termination of DDF continues.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the failure, fix the error condition, and then issue the -START
DDF command.

Problem determination: Refer to *z/OS Communications Server SNA Programming* for failure analysis during a CLOSE
macro instruction.

DSNL015I   VTAM SETLOGON FAILED RTNCD=*error-code* **FDBK2=***open-flags*

Explanation: VTAM returned a nonzero return code after DB2 invoked the SETLOGON macro during initialization
of the distributed data function (DDF). VTAM return codes are defined in *z/OS Communications Server SNA
Programming*.

System action: Termination of the DDF is started.

Operator response: Notify the system programmer.

System programmer response: Refer to *z/OS Communications Server SNA Programming* to determine the cause of the
failure. After the error is fixed, issue the -START DDF command.

DSNL016I   DDF RECORD NOT FOUND IN BSDS

Explanation: DDF cannot find the DDF communication record in the BSDS.

System action: DDF termination process is initiated.

Operator response: Notify the system programmer.

System programmer response: Use the change log inventory to add the DDF communication record to the BSDS.

Problem determination: Collect the following diagnostic item:
- Print the contents of the BSDS using the print log map utility.

DSNL017I   INVALID DDF RECORD FOUND IN BSDS REASON = *reason-code*

Explanation: An invalid or incomplete DDF communication record was found in the BSDS.

System action: DDF begins to terminate processing.

Operator response: Notify the system programmer.

System programmer response: See the information for the associated *reason-code*. Use the change log inventory utility to correct the DDF communication record in the BSDS.

Problem determination: Collect the following diagnostic item:
- Print the contents of the BSDS using the print log map utility.

DSNL018I   VSAM ERROR READING DDF RECORD IN BSDS RPLERRCD=*error-code*

Explanation: A VSAM GET command was issued that resulted in a nonzero return code. RPLERRCD 'error-code' is the VSAM return code.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the failure, fix the error condition, and issue the -START DDF command.

The cause of the failure is dictated by the VSAM return code. The BSDS may have to be recovered by using a backup copy.

Problem determination: Refer to the appropriate VSAM publications.

DSNL019I   VTAM DOES NOT SUPPORT APPC FUNCTIONS

Explanation: The installed release of VTAM does not support the advanced program-to-program communications (APPC) application program interface (API). The APPC API bit is not set in the VTAM function-list vector in the access-method-support vector list.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine whether the installed release of VTAM supports the APPC API. If VTAM does not support the APPC API, a VTAM release that supports the APPC API must be installed before the DDF can be started.

DSNL020I   KEYWORD 'THREAD' PARAMETER '*parameter***' FORMAT IN ERROR**

Explanation: The parameter value specified for the logical unit of work ID on the CANCEL DDF THREAD has an invalid format.

parameter
        The invalid parameter string.

The correct format for the parameter is either *network-id.luname.luwunique* or *luname.luwunique,* using the following values:

network-id
An optional 1- to 8-character network ID.

luname   A 1- to 8-character logical unit name.

luwunique
A 12-character hexadecimal number.

System action: The specified thread is not canceled.

User response: Determine the correct logical unit of work ID of the distributed thread that is to be canceled and reissue the command.

Problem determination: Use the -DISPLAY THREAD command to display all distributed activity.

DSNL021I   *command* **ACCEPTED**

Explanation: The 'command' command has been accepted and asynchronous processing of 'command' has been initiated. The command can only be -START or -STOP the distributed data facility.

System action: The command was accepted and asynchronous processing has been initiated.

DSNL022I   *csect-name* **DDF THREAD '*luw-id*' WAS PREVIOUSLY CANCELED**

Explanation: A command to cancel the distributed data facility (DDF) thread *luw-id* was issued, but a command to cancel the same thread has already been issued.

csect-name
The name of the control section that issued the message.

luw-id   The logical unit of work ID.

Either the CANCEL DDF THREAD or the CANCEL THREAD command was entered.

System action: The thread is currently being canceled.

User response: No action is required.

Programmer response: The CANCEL command terminates a thread only if the thread is processing in DB2 or is currently suspended by DB2. The thread might not be processing in DB2 in these situations:
- The thread is processing in the application and has not made a request to DB2. The DISPLAY THREAD command indicates this.
- The user placed the thread on an MVS resource using the ENQUE command. The CANCEL command has no effect on the thread until ENQUE completes and the thread returns to DB2.
- The thread is processing in VTAM and is waiting for a response from a partner. The DISPLAY THREAD DETAIL command indicates this. In this case the thread does not terminate until VTAM returns to DB2. VTAM commands, which will terminate the session or LU, can be used to cause VTAM to return to DB2.

DSNL023I   **DDF IS ALREADY IN THE PROCESS OF STARTING**

Explanation: The distributed data facility (DDF) was starting when the -START DDF or -STOP DDF command was issued.

System action: The issued command is not executed.

DSNL024I   **DDF IS ALREADY IN THE PROCESS OF STOPPING**

Explanation: The distributed data facility (DDF) was stopping when the -STOP DDF or -START DDF command was issued.

System action: The -START or -STOP DDF command is not executed.

DSNL025I   A DISTRIBUTED AGENT FOR *luw-id=token* EXISTS AT THIS LOCATION BUT NOT AT SITE WITH LOCATION *location* AND LU *luname*

Explanation: The DDF error monitor has detected that a distributed agent exists at this location but not at the named partner location.

Where

luw-id=token

> The logical unit of work ID of the thread followed by an '=' sign and a token which may be used in place of 'luw-id' in any DB2 command that accepts 'luw-id' as input.

location

> The location name of the partner site where a thread with the same 'luw-id' does not exist.

luname

> The 8-character LUNAME of the partner location.

Operator response: Determine if the inconsistency is due to network conditions or a DB2 error. If a DB2 error is suspected, acquire documentation to diagnose the problem.

1. Console dump of DB2 address spaces (*ssnm*DBM1, *ssnm*MSTR, *ssnm*DIST), VTAM, and user's address space at the requesting site.
2. Console dump of DB2 address spaces (*ssnm*DBM1, *ssnm*MSTR, *ssnm*DIST), and VTAM at the server site.
3. SYSLOG from requesting and server sites.
4. LOGREC from requesting and server sites.

If an indefinite inconsistency is suspected, use the CANCEL DDF THREAD command to terminate the thread. If CANCEL DDF THREAD fails to terminate the agent, use VTAM commands to terminate the thread conversations.

Problem determination: Use the DB2 DISPLAY THREAD command with DETAIL option at each DB2 subsystem to display information about the agent and its conversations. Use VTAM commands to display additional information about the conversations being used by the thread and also about network conditions.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

.

DSNL026I   LUW-ID *luw-id=token* APPEARS TO BE SUSPENDED PARTNER LOCATION *location* PARTNER LU *luname*

Explanation: The DDF error monitor has detected that a distributed agent with the specified 'luw-id' at this location and at the named partner 'location' may be suspended indefinitely.

Where

luw-id=token

> The logical unit of work ID of the thread followed by an '=' sign and a token which may be used in place of 'luw-id' in any DB2 command that accepts 'luw-id' as input.

location

> The location name of the partner site where a thread with the same 'luw-id' exists.

luname

> The 8-character LUNAME of the partner location.

Operator response: Determine if the inconsistency is due to network conditions or a DB2 error. If a DB2 error is suspected, acquire documentation to diagnose the problem.

1. Console dump of DB2 address spaces (*ssnm*DBM1, *ssnm*MSTR, *ssnm*DIST), VTAM, and hung user's address space at the requesting site.

2. Console dump of DB2 address spaces (*ssnm*DBM1, *ssnm*MSTR, *ssnm*DIST), and VTAM as the server site.

3. SYSLOG from requesting and server sites.

4. LOGREC from requesting and server sites.

If an indefinite inconsistency is suspected, use the CANCEL DDF THREAD command to terminate the thread. If CANCEL DDF THREAD fails to terminate the agent, use VTAM commands to terminate the thread conversations.

Problem determination: Use the DB2 DISPLAY THREAD command with DETAIL option at each DB2 subsystem to display information about the agent and its conversations. Use VTAM commands to display additional information about the conversations being used by the thread and also about network conditions.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

.

DSNL027I   *agent-type* **DISTRIBUTED AGENT WITH LUW-ID** *luw-id=token* **THREAD-INFO** *thread-information* **RECEIVED ABEND=***abend-code* **REASON=***reason-code*

Explanation: A distributed agent has abended at this location.

agent-type
> Information about the distributed agent that has abended:
>
> **REQUESTING**
>> The thread is an allied thread that has distributed activity.
>
> **SERVER**
>> The thread is a database access thread.

luw-id=token
> The logical unit of work identifier (LUWID) and a five-digit token value. The *token* value can be used in any DB2 command that accepts a LUWID as input.

thread-information
> Information about the thread. The information is presented in a colon-delimited list that contains the following segments:
>
> - The primary authorization ID that is associated with the thread.
>
>   In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.
>
> - The name of the user's workstation.
> - The ID of the user.
> - The name of the application.
> - The statement type for the previously executed statement: dynamic or static.
> - The statement identifier for the previously executed statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table. That statement might not have contributed to the current failure.
> - The name of the role that is associated with the thread.
> - The correlation token that can be used to correlate work at the remote system with work performed at the DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, separated by periods:
>   – A 3 to 39 character IP address.
>   – A 1 to 8 character port address.
>   – A 12 character unique identifier.

An asterisk (*) in any segment indicates that the information is not available.

abend-code
:   The abend code. A value of zeros indicates that the abend code could not be determined.

reason-code
:   A numeric value that indicates the reason for the failure. A value of zeros indicates that the reason code could not be determined.

This message might be followed by instances of message DSNL028I, each of which describes a remote location where the distributed agent also exists and where diagnostic information might also need to be collected.

Problem determination: Use the information provided for the *abend-code* value to determine what diagnostic information to gather.

Look for instances of message DSNL028I that identify other locations that are involved in the abend.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related information:
"DSNL028I"

DSNL028I   *luw-id=token* **ACCESSING DATA** *preposition* **LOCATION** *location nettype netvalue*

Explanation: This message follows DSNL027I and identifies a remote location where work might have been occurring for the distributed thread. This message is informational and with the DSNL027I message identifies a remote location where diagnostic information might have to be collected.

luw-id=token
:   The logical unit-of-work ID of the thread followed by an '=' sign and a token which may be used in place of 'luw-id' in any DB2 command that accepts 'luw-id' as input.

preposition
:   'FOR' if the thread is a database access thread and 'AT' if the thread is an allied thread that has distributed activity.

location
:   This field contains one of the following:
    - The requester's relational database name (location name)
    - The requester's VTAM LU name (<luname>)
    - The requester's IP address

nettype
:   Identifies the type of network used to access the remote site. There are two possible values:

    LU --   indicates that VTAM is used to connect to the remote site.

    IPADDR --
    :   indicates that TCP/IP is used to connect to the remote site.

netvalue
:   The network address of the remote location. *netvalue* can be blank.
    - If *nettype* is LU, *netvalue* specifies the VTAM LU name of the remote location.
    - If *nettype* is IPADDR, *netvalue* specifies the IP address of the partner.

Problem determination: See message DSNL027I.

DSNL029I   **DDF IS NOT INITIALIZED**

Explanation: A -START DDF or -STOP DDF command was issued before DDF was initialized during DB2 startup processing.

Operator response: Wait for the DSN9022I 'START DB2' NORMAL COMPLETION message and re-issue the command.

DSNL030I • DSNL031I

DSNL030I   *csect-name* **DDF PROCESSING FAILURE FOR LUWID=***luw-id* **REASON=***reason-code*
               THREAD-INFO=*thread-information*

Explanation: A requesting conversation was terminated because of DDF processing failure.

csect-name
> The name of the control section that issued the message.

luw-id   The logical unit of work ID of the requesting agent, if that information is available, or a description of the
> logical unit that made the request in the format "*netid.luname*.000000000000".

reason-code
> A numeric value that describes the nature of the DDF processing failure.

thread-information
> Information about the thread. The information is presented in a colon-delimited list that contains the
> following segments:
> - The primary authorization ID that is associated with the thread.
> - The name of the user's workstation.
> - The ID of the user.
> - The name of the application.
> - The statement type for the previously executed statement: dynamic or static.
> - The statement identifier for the previously executed statement, if available. The statement identifier can be
>   used to identify the particular SQL statement. For static statements, the statement identifier correlates to
>   the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement
>   identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table. That
>   statement might not have contributed to the current failure.
> - The name of the role that is associated with the thread.
> - The correlation token that can be used to correlate work at the remote system with work performed at the
>   DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three
>   components, separated by periods:
>   - A 3 to 39 character IP address.
>   - A 1 to 8 character port address.
>   - A 12 character unique identifier.
>
> An asterisk (*) in any segment indicates that the information is not available.

System action: Processing continues, but the system will eventually abend the database access agent.

Operator response: Notify the system programmer for analysis.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For
information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis
Guide and Reference.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
  time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNL031I   *csect-name* **DRDA EXCEPTION CONDITION IN RESPONSE FROM SERVER LOCATION=***location-*
               indicator **FOR THREAD WITH LUWID=***luwid* **REASON=***reason-code* **ERROR ID=***error-id*
               CORRELATION ID=*correlation-id* **CONNECTION ID=***connection-id* **IFCID=***ifcid* **SEE TRACE RECORD**
               WITH IFCID SEQUENCE NUMBER=*seq-no*

Explanation: A DRDA exception condition was detected in response data returned from the application server (AS).
Possible error conditions include:
- An error reply message and severity code received from the server.
- A parsing error detected while processing a reply message or reply data object
- A DRDA protocol violation, (for example, an object returned that is not valid for this command).

csect-name
>   The name of the control section that issued the message.

luw-id   The logical unit of work (LUW) ID that identifies the thread. This values is also included in the trace record.

location-indicator
>   An indicator of the location:
>   - The server's relational database name
>   - The server's VTAM LU name
>   - The server's IP address

reason-code
>   The DB2 reason code for this specific error.

error id   The module name concatenated with a 4-byte source id that uniquely determines the source within the module where the exception is encountered.

correlation-id
>   The correlation identifier of the thread that detected the DRDA exception condition. Correlation IDs beginning with 001 through 028, inclusive, identify system agents.

connection-id
>   The connection name of the thread that detected the DRDA exception condition.

ifcid-number
>   The IFCID number of the trace record that was written.

sequence-number
>   The IFCID sequence number associated with the trace record. The trace record contains information that can be used to determine what caused the DRDA exception condition, and is written to the destination for IFCID 191. The message and the trace record are written only the first time an error is detected. This avoids flooding the console and the trace. The sequence number is printed in hexadecimal.
>
>   If the sequence number is zero, then the trace data was not externalized.

System action: An alert was generated. Additionally, if the seq-no is not zero, the system wrote a trace record with IFCID=*ifcid-name* to the destination for IFCID 191 that can be used to diagnose the problem.

Problem determination: Determine the destination of the trace record and locate and print the trace record with SEQUENCE NUMBER=*sequence-number*. The description of the reason code can be used in conjunction with the publication Distributed Relational Database Architecture™ Reference to determine the type of DRDA exception that has been detected. Determine the problem and which system is responsible by using the procedures that are described in DB2 Diagnosis Guide and Reference.

Related concepts:

➦ DB2 reason codes (DB2 Codes)

DSNL032I   *csect-name* **DRDA EXCEPTION CONDITION IN REQUEST FROM REQUESTER LOCATION=***location-indicator* **FOR THREAD WITH LUWID=***luw-id* **REASON=***reason-code* **ERROR ID=***error-id* **IFCID=***ifcid-number* **SEE TRACE RECORD WITH IFCID SEQUENCE NUMBER=***sequence-number*

Explanation: A DRDA exception condition was detected while processing a command or command data received from the application requester (AR). Possible error conditions include errors detected while processing a command or a command data object.

csect-name
>   The name of the control section that issued the message.

location-indicator
>   An indicator of the location:
>   - The requester's relational database name
>   - The requester's VTAM LU name
>   - The requester's IP address

luw-id   The logical unit of work (LUW) ID that identifies the thread. This value is also included in the trace record, or "NETID.LUNAME.000000000000" if the error detected involves an invalid FMH5.

reason-code
          The DB2 reason code for this specific error.

error-id   The module name concatenated with a 4-byte source ID that uniquely determines the source within the module where the exception was encountered.

ifcid-number
          The IFCID number of the trace record that was written.

sequence-number
          The IFCID sequence number associated with the trace record. The trace record contains information that can be used to determine what caused the DRDA exception condition, and is written to the destination for IFCID 191. The message and the trace record are written only the first time an error is detected. This avoids flooding the console and the trace. The sequence number is printed in hexadecimal.

          If the sequence number is zero, then the trace data was not externalized.

System action: An alert was generated. Additionally, if the seq-no is not zero, the system wrote a trace record with IFCID=*ifcid-name* to the destination for IFCID 191 that can be used to diagnose the problem.

Operator response: Notify the system programmer for analysis.

System programmer response: Determine the destination of the trace record and locate and print the trace record with SEQUENCE NUMBER=*sequence-number*. The description of the reason code can be used in conjunction with the publication Distributed Relational Database Architecture Reference to determine the type of DRDA exception that has been detected. Determine the problem and which system is responsible by using the procedures that are described in DB2 Diagnosis Guide and Reference.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

DSNL033I   *csect-name* **DDF TERMINATION SCHEDULED DUE TO DETECTION OF ABEND** *abend* **REASON** *reason*

Explanation: A severe error was detected during DDF processing. DDF processing can no longer continue so DDF termination has been scheduled. 'Csect-name' is the DB2 module that detected the condition and requested DDF termination.

The abend and reason might also contain additional information about the cause.

abend   The abend code or zeros if no abend is associated with the termination of DDF.

          If the abend was a DB2 04E abend, the DB2 reason code will contain additional information about the abend.

reason   The DB2 reason code or zeros if no DB2 reason code is associated with the termination of DDF.

          If no abend is associated with the problem, the reason code describes the cause for the termination.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer for analysis.

System programmer response: Determine the cause of the failure. Correct the condition and restart DDF using the -START DDF command.

Problem determination: If an abend is associated with the failure, a dump might have been provided. LOGREC will also contain information about the abend.

DSNL034E   **DDF CANNOT BE STARTED BECAUSE OF BSDS INCONSISTENCIES, ORIGINAL LUNAME=***original-luname***, BSDS LUNAME=***bsds-luname***, ORIGINAL LOCATION=***original-location***, BSDS LOCATION=***bsds-location*

Explanation: The LUNAME or the LOCATION name stored in the bootstrap data set (BSDS) was changed after the distributed data facility (DDF) was last started, but DDF still has resynchronization responsibility with one or more

partner logical units using the old LUNAME and LOCATION name. Both the original and current BSDS values of LUNAME and LOCATION name are displayed.

original-luname
> The LUNAME required for resynchronization.

bsds-luname
> The LUNAME currently stored in the BSDS.

original-location
> The LOCATION name required for resynchronization.

bsds-location
> The LOCATION name currently stored in the BSDS.

System action: DDF cannot be started until the inconsistencies are corrected.

Operator response: Inform the database administrator.

Database administrator response: Take the appropriate action, depending on whether there is a requirement to change the LUNAME or LOCATION name in the BSDS:

Name changed in error
> Use the change log inventory (DSNJU003) utility to change the LUNAME, LOCATION name , or both back to the correct value. Then issue the START DDF command.

Name change required
> Remove all of the resynchronization responsibility that DB2 had using the old names. You can purge this information, allow automatic resynchronization to take place, or use a combination of both.
>
> You can purge resynchronization information by using the RESET INDOUBT command with the FORCE option. You can purge all information regardless of partner name (LOCATION (*)), or you can selectively purge by specifying a location name. DDF does not have to be started to use this command. You can use the DISPLAY THREAD command to generate a report of indoubt threads. This report indicates all resynchronization responsibility known by DB2.
>
> If you do not purge the resynchronization information, or choose to selectively purge, and you want automatic resynchronization to occur before changing the names, you must first use the change log inventory (DSNJU003) utility to change the LUNAME, the LOCATION name, or both back to the correct value. Then issue the START DDF. At this point, automatic resynchronization begins. You can monitor the progress by periodically using the DISPLAY THREAD command for TYPE (INDOUBT) threads. Resynchronization does not complete until all needed communications paths are available and all needed partner logical units are available. If you cannot wait for availability, use the RESET INDOUBT command with the FORCE option to purge information associated with unavailable partners.

Related reference:

-DISPLAY THREAD (DB2) (DB2 Commands)

-RESET INDOUBT (DB2) (DB2 Commands)

-START DDF (DB2) (DB2 Commands)

DSNJU003 (change log inventory) (DB2 Utilities)

DSNL035I   DDF CANNOT BE STOPPED BECAUSE AN INDOUBT DATA BASE ACCESS THREAD EXISTS

Explanation: DDF is the connection manager for a database access thread (DBAT) that is indoubt. A STOP DDF MODE(QUIESCE) command is rejected until the indoubt logical unit of work is resolved.

System action: DDF is not stopped.

Operator response: Inform the database administrator.

Database Administrator Action: You have the following options:

- Wait for automatic resynchronization to be performed. When this occurs, either message DSNL407I or DSNL415I is displayed.

  After the indoubt thread is resolved, reissue the STOP DDF command.

- Use the DISPLAY THREAD TYPE(INDOUBT) command to find the LOCATION that has coordinator responsibility for the indoubt LUW. Contact that location and determine whether the LUW committed or aborted. Then use the RECOVER INDOUBT command to cause the indoubt DBAT to be resolved with the same decision.

  After the indoubt thread is resolved, reissue the STOP DDF command.

- Issue the STOP DDF MODE(FORCE) command. The indoubt thread remains indoubt and can be resolved by using automatic resynchronization after DDF is started again, or it can be resolved manually as described above.

DSNL036I   DDF THREAD *logical-unit-of-work-id* **SPECIFIED IS INDOUBT**

Explanation: A database access thread that is indoubt cannot be canceled by using the -CANCEL DDF THREAD command.

System action: The specified thread is not canceled.

Operator response: Use the -DISPLAY THREAD TYPE(*) command to display all distributed activity.

DSNL037I   *csect-name* **RESET GENERICLU** *lu* **SUCCESSFUL**

Explanation: The RESET GENERICLU command successfully purged the VTAM sysplex information from shared memory.

lu   The value supplied on the RESET GENERICLU command

System action: The VTAM information contained in the sysplex shared memory for the indicated LU is purged.

DSNL038E   *csect-name* **RESET GENERICLU** *lu* **FAILED DUE TO VTAM RTNCD=**xx **and FDB2=**yy

Explanation: The VTAM CHANGE,OPTCD=(SYN,ENDAFFIN) macro returned an unexpected return code, which is described by the indicated VTAM RTNCD and FDB2 values.

lu   The value supplied on the RESET GENERICLU command

xx   The hexadecimal value of the VTAM RTNCD field

yy   The hexadecimal value of the VTAM FDB2 field

System action: VTAM was not able to purge the sysplex shared memory information for the indicated LU.

System programmer response: Refer to *z/OS Communications Server SNA Programming* for the meaning of the VTAM RTNCD and FDB2 values.

DSNL039E   *csect-name* **RESET GENERICLU** *lu* **FAILED DUE TO ACTIVE VTAM SESSIONS**

Explanation: The RESET GENERICLU command was not accepted because one or more active VTAM sessions exist to the indicated LU. All VTAM sessions must be terminated before the command is accepted.

lu   The value supplied on the RESET GENERICLU command

System action: The command is rejected.

System programmer response: Terminate the VTAM sessions to the remote LU and reissue the -RESET GENERICLU command.

DSNL040E   *csect-name* **RESET GENERICLU** *lu* **FAILED DUE TO INDOUBT RESOLUTION INFORMATION**

Explanation: The RESET GENERICLU command was not accepted because information is currently being retained for resolution of indoubt threads for the indicated partner LU. The DISPLAY THREAD TYPE(INDOUBT) command can be used to view this information.

lu   The value supplied on the RESET GENERICLU command

System action: The command is rejected.

System programmer response: Make sure that all indoubt threads at the remote site are resolved before issuing the RESET GENERICLU command.

When DB2 performs automatic indoubt thread resolution, the information contained in the DISPLAY THREAD

TYPE(INDOUBT) display is purged at the end of the resynchronization process.

If required, use the -RESET INDOUBT command to manually purge the indoubt resolution information.

DSNL041E   *csect-name* **DB2 GENERIC LU NAME REQUESTED, BUT REQUIRED VTAM LEVEL IS NOT INSTALLED**

Explanation: The DB2 GENERIC LUNAME installation option was specified. This option requires VTAM Version 4 Release 2 or a later release.

System action: The DB2 GENERIC LUNAME installation option is ignored.

System programmer response: If the DB2 GENERIC LUNAME is required, install VTAM Version 4 Release 2 or a later release.

DSNL042E   *csect-name* **DB2 SYSPLEX TP WAS INVOKED BY LU** *lu*, **BUT REQUIRED MVS LEVEL IS NOT INSTALLED**

Explanation: A request for the DB2 sysplex transaction program (TP) was received by a DB2 server that is a member of a data sharing group. The DB2 sysplex TP is used by the requesting systems to determine which DB2 systems within the group are currently available for DDF conversations.

The DB2 sysplex TP uses Workload Manager (WLM) functions in MVS Version 5 Release 2 or a later release to balance distributed workload among the DB2 systems in the group. Because the required MVS release is not installed or DB2 is not registered with WLM, the DB2 sysplex TP is not able to provide the requested information.

lu   The name of the LU in the network that invoked the DB2 sysplex TP

System action: The DB2 sysplex TP returns the NETID.LUNAME of the local DB2 system as the only available DB2 server in the group. The requester is not provided the names of other DB2 systems in the group, so all sessions from the requester are routed to the local DB2 server.

System programmer response: Install MVS Version 5 Release 2 or a later release if you want to have the DB2 sysplex TP return multiple NETID.LUNAME values to the requester or refer to the DSNL044I on why DB2 could not register with WLM.

DSNL043I   *csect-name* **RESET GENERICLU** '*luname*' **INVALID**

Explanation: The *luname* value specified is invalid.

csect-name
        The name of the control section that issued the message.

luname   The logical unit name.

The correct format for the logical unit name is either *luname* or *network-id.luname*, using the following values:

network-id
        An optional 1- to 8-character network ID.

luname   A 1- to 8-character logical unit name.

System action: The parameter is ignored. Processing continues with the next logical unit name, if any.

User response: Determine the correct value for *luname* and reissue the command.

DSNL044I   *csect-name* **MVS** *macro-name* **MACRO FAILED WITH RETCODE='***return-code***'X AND RSNCODE='***reason-code***'X**

Explanation: An MVS Workload Manager (WLM) macro returned an unexpected return code, which is described by the indicated RETCODE and RSNCODE values. DDF uses MVS Workload Manager macros to obtain data sharing sysplex routing information for incoming DDF work requests, and to establish an MVS enclave for the execution environment of the distributed thread.

csect-name
        The name of the control section that issued the message.

macro-name
> The name of the MVS macro that returned the unexpected return code. The possible values are:
> - IWMSRSRG: Register with WLM for sysplex routing.
> - IWMSRSRS: Request sysplex routing information.
> - IWMSRDRS: Deregister with WLM for sysplex routing.
> - IWMECREA: Create enclave.
>
> The WLM sysplex routing service macros will be appended with the contents of the LOCATION parameter passed to it. The contents of the LOCATION parameter will either be the member's location name or one of the alias-names being defined to subset the data sharing group. If the member's location name is displayed, then this member will not participate in WLM workload balancing. If a subsetting alias-name is displayed, then subsetting of the data sharing group by that alias-name will be disabled. However, a connection to this member of the data sharing group will be accepted when using the alias-name instead of the member's location name.

return-code
> The hexadecimal value of the MVS return code.

reason-code
> The hexadecimal value of the MVS reason code.

System action: If a sysplex routing macro failed, then sysplex routing may not be performed in which case, all SQL statements received from the client will be processed by this DB2 subsystem. If an enclave macro failed, the distributed thread may not be able to execute in an MVS enclave, however the distributed thread will continue to execute as an MVS SRB.

System programmer response: Use the MVS return and reason codes to determine the problem.

Related reference:

⇨ Setting up a Service Definition (MVS Programming: Workload Management Services)

DSNL045I   *csect-name* **ICSF** *function-name* **FUNCTION FAILED WITH RETCODE='***return-code***'X AND RSNCODE='***reason-code***'X**

Explanation: DB2 for z/OS made an Integrated Cryptographic Service Facility (ICSF) call because a connection to a remote server or a connection from a remote requester is using encryption. DB2 for z/OS uses ICSF to perform the following functions:

- Creating connection keys
- Generating a shared private key
- Encrypting data
- Decrypting data

The ICSF call returned an unexpected return code.

csect-name
> The name of the control section that issued the message.

function-name
> The name of the ICSF call that returned the unexpected return code. The possible values are:
> - CSNERNG: Random Number Generate
> - CSNECKM: Multiple Clear Key Import
> - CSNFPKB: Key Token Build
> - CSNFPKE: Key Encrypt
> - CSNEENC: Encipher data.
> - CSNEDEC: Decipher data
> - CSNESYE: Symmetric Encipher
> - CSNESYD: Symmetric Decipher
> - CSNEOWH: MD5 Hash

return-code
> The hexadecimal value of the ICSF return code.

reason-code
> The hexadecimal value of the ICSF reason code.

System action: The encryption function cannot be processed. If ICSF services were successfully loaded during DB2 startup processing, DB2 retries the ICSF services call periodically. Each time that the call fails, DB2 issue this message. If ICSF services were not successfully loaded, DB2 does not retry the ICSF services call.

System programmer response: If the encryption facility is not installed, install it before requesting encryption. If the encryption facility is installed, verify that it is working correctly. Look up the return code and reason code in the ICSF documentation for help in finding and fixing the problem.

Related reference:

➥ z/OS Cryptographic Services Integrated Cryptographic Service Facility System Programmer's Guide

---

DSNL046I   *csect-name* **ICSF is not enabled**

Explanation: This message indicates that a cryptographic service is required, but the Integrated Cryptographic Service Facility (ICSF) is unavailable.

System action: DB2 cannot continue with the encryption or decryption function.

System programmer response: If the cryptographic service facility is not installed, install it before requesting encryption functions. If it is configured, verify that the ICSF service is available and working correctly.

Related reference:

➥ z/OS Cryptographic Services Integrated Cryptographic Service Facility System Programmer's Guide

---

DSNL047I   **THE REQUESTED TRUSTED CONNECTION CANNOT BE ESTABLISHED TO THE LOCATION** *location-name*

Explanation: The trusted connection that you requested cannot be established to the location *location-name*.

System action: The attempt to create a trusted connection was unsuccessful. A normal connection was created without any attributes of a trusted context.

Problem determination: *location-name* identifies the name of the server that could not support the trusted connection.

DSNL048I   *csect-name* **SAF USER MAPPING FUNCTION** *function-name* **FAILED WITH RETURN VALUE=***return-value* **RETURN CODE=***return_code* **REASON CODE=***reason_code* **ERROR MESSAGE:***saf_error_message*

Explanation: The SAF user mapping plug-in function failed. When DB2 receives a user registry name from a remote client application, DB2 calls the SAF user mapping plug-in service to retrieve the z/OS user ID which will be used as the primary authorization ID for the remote connection.

function-name
> The name of the SAF user mapping function that failed. Possible functions can be safMappingInit(), safMappingLookup(), or safMappingTerm().

return-value
> SAF user mapping function return value.

return_code
> SAF user mapping function return code.

reason_code
> SAF user mapping function reason code.

saf_error_message
> SAF user mapping function error message detailing the error condition.

System programmer response: Refer to the *z/OS Security Server - RACF Messages and Codes* for the meaning of SAF user mapping function *return_value*, *return_code*. *reason_code*, and *saf_error_message*.

DSNL065I   *csect-name* **CURRENT DDF STATE DOES NOT PERMIT STOP DDF MODE(SUSPEND)**

Explanation: The current DDF state does not allow suspend processing to be initiated. DDF must be started for normal processing in order for suspend processing to be initiated.

System action: The command is not executed.

DSNL066I   *csect-name* **STOP DDF MODE(SUSPEND) COMPLETE**

Explanation: DDF suspend processing has completed successfully. DDF has suspended all inbound activity.

System action: Processing continues normally.

DSNL067I   *csect-name* **CURRENT DDF STATE DOES NOT PERMIT START DDF**

Explanation: The current DDF state does not allow START DDF processing to be initiated.

System action: The command is not executed.

DSNL068I   *csect-name* **START DDF (RESUME PROCESSING) COMPLETE**

Explanation: DDF has resumed normal processing.

System action: Processing continues normally.

DSNL069I   *csect-name* **DDF IS SUSPENDING**

Explanation: DDF has initiated suspend processing.

System action: Processing continues normally.

DSNL070I   *csect-name* **DDF IS RESUMING**

Explanation: DDF has initiated resume processing.

System action: Processing continues normally.

DSNL071I   *csect-name* **WAIT OR CANCEL REQUIRES KEYWORD SUSPEND**

Explanation: The WAIT(*n*) or CANCEL(*n*) keywords require that MODE(SUSPEND) be specified.

System action: The command is not executed.

User response: Add the missing keyword and reissue the command.
Related reference:

⮕ -STOP DDF (DB2) (DB2 Commands)

DSNL072I   *csect-name* **WAIT TIME EXPIRED, DDF RESUME PROCESSING INITIATED**

Explanation: The DDF suspend process has not completed successfully in the period of time specified for the WAIT keyword, so DDF resume processing has been initiated.

System action: DDF resume processing has been initiated.
Related reference:

⮕ -STOP DDF (DB2) (DB2 Commands)

DSNL073I   *csect-name* **CANCEL TIME EXPIRED, CANCEL ACTIVE DBAT PROCESSING HAS BEEN INITIATED**

Explanation: The DDF suspend process has not completed successfully in the period of time specified for the CANCEL keyword so DDF is canceling all active data base access threads (DBATS).

System action: DDF is attempting to terminate all active database access threads.

Related reference:

➡ -STOP DDF (DB2) (DB2 Commands)

DSNL074I   **NUMBER OF CLIENT CONNECTIONS HAS EXCEEDED** *exceeded-percent* **% OF THE MAXIMUM ALLOWED**

Explanation: The number of increasing client connections has exceeded either 80% or 90% relative of the value of the CONDBAT subsystem parameter.

percent-exceeded
> The percentage of the threshold specified by the value of the CONDBAT subsystem parameter that was exceeded.

The DSNL074I message is issued only when the DB2 subsystem is a member of a data sharing group.

System action: The health value reported to WLM is decreased to 50% (when the value of *exceeded-percent* is 80) or 25% (when the value of *exceeded-percent* is 90) of the calculated DB2 system health value. This action might enable remote client systems to direct work to other members which have more resources to process the work.

The modified health value remains until the number of client connections decreases below either the 80% or 90% threshold.

System programmer response: If a subsequent DSNL075I is displayed, no action is required.

Otherwise, investigate why the number of connections has exceeded the threshold set by the CONDBAT subsystem parameter. You can use the DISPLAY LOCATION command to determine which client locations are sending increasing numbers of connections to the DB2 subsystem. If the cause of the increasing connections in normal workload growth, then additional monitoring might be required. If the increasing connections are the result of and exception condition, additional investigation might be required to resolve the problem.

Related reference:

➡ MAX REMOTE CONNECTED field (CONDBAT subsystem parameter) (DB2 Installation and Migration)

➡ -DISPLAY LOCATION (DB2) (DB2 Commands)

Related information:

"DSNL075I"

DSNL075I   **NUMBER OF CLIENT CONNECTIONS NO LONGER EXCEEDS** *percent-exceeded***% OF THE MAXIMUM ALLOWED**

Explanation: The number of client connections has decreased such that either the 80% or 90% threshold relative to the MAX CONDBAT subsystem parameter is no longer being exceeded.

percent-exceeded
> The percentage of the threshold specified by the value of the CONDBAT subsystem parameter that was exceeded.

The health value reported to WLM will also be increased to 50% (when the value of *percent-exceeded* is 90) or 100% (when the value of *percent-exceeded* is 80) of the calculated DB2 system health value.

The DSNL075I message can only be issued when the DB2 subsystem is a member of a data sharing group.

System action: Processing continues.

User response: No action is required.

Related reference:

➡ MAX REMOTE CONNECTED field (CONDBAT subsystem parameter) (DB2 Installation and Migration)

Related information:

"DSNL074I"

DSNL080I *csect-name* **DISPLAY DDF REPORT FOLLOWS:**

Explanation: This message displays output from the DISPLAY DDF command. The output includes the status and configuration of DDF for the local subsystem or member. It also includes statistics for the connections and threads that are controlled by DDF.

csect-name
> The name of the control section that issued the message.

The remaining output consists of one or more of the following messages, in the indicated order:

- DSNL081I: DDF status
- DSNL082I and DSNL083I: Location information
- DSNL084I: TCP/IP port information
- DSNL085I: IP address
- DSNL086I: TCP/IP domain name
- DSNL087I and DSNL088I: Aliases
- DSNL089I: Member IP address
- DSNL090I: Subsystem parameter information (CMTSTAT, CONDBAT, and MAXDBAT)
- DSNL091I: Subsystem parameter information (MAXCONQN and MAXCONQW)
- DSNL092I and DSNL093I: DBAT and connection information
- DSNL094I: Closed client connections
- DSNL096I: Detailed information for a specific alias
- DSNL100I, DSNL101I, DSNL102I: Server list information
- DSNL105I and DSNL106I: DDF options
- DSNL099I: End of output

DSNL081I: DDF status:

The introduction to the output is followed by the operational status of DDF:

```
DSNL081I STATUS=status
```

status   The DDF status, which can be one of the following values:

> **STARTD**
> > DDF is started.
>
> **STARTG**
> > DDF is starting.
>
> **STOPDF**
> > DDF was stopped abnormally, for example, by the STOP DDF command with the MODE(FORCE) option.
>
> **STOPDQ**
> > DDF was stopped normally or is not yet started. DDF can be stopped normally by using the STOP DDF command with the MODE(QUIESCE) option.
>
> **STOPGF**
> > DDF is in the process of stopping abnormally, for example, by the STOP DDF command with the MODE(FORCE) option.
>
> **STOPGQ**
> > DDF is in the process of stopping normally, for example, by the STOP DDF command with the MODE(QUIESCE) option
>
> **SUSPND**
> > DDF is suspended, for example, by the STOP DDF command with the MODE(SUSPEND) option.
>
> **SUSPNG**
> > DDF is in the process of suspending, for example, by the STOP DDF command with the MODE(SUSPEND) option.

> **NOINIT**
> > DDF is not configured. The value of the DDF subsystem parameter is NO.

Related information:

-STOP DDF (DB2) (DB2 Commands)

DDF STARTUP OPTION field (DDF subsystem parameter) (DB2 Installation and Migration)

DSNL082I and DSNL083I: Location information:

The DDF status information is followed by location information:

```
DSNL082I LOCATION          LUNAME            GENERICLU
DSNL083I location          luname            genericlu
```

location The location name of DDF, as recorded in the bootstrap data set (BSDS). The location name can be specified by one of the following methods:

- The DB2 LOCATION NAME field on the DSNTIPR installation panel
- The LOCATION option in the DDF statement of the DSNJU003 (change log inventory) utility

luname The DDF LU name, as recorded in the BSDS. The LU name can be specified by one of the following methods:

- The DB2 NETWORK LUNAME field on the DSNTIPR installation panel
- The LUNAME option in the DDF statement of the DSNJU003 (change log inventory) utility

A value of -NONE indicates one of the following situations:

- An LU name is not configured.
- The DDF record in the BSDS already has a value for IPNAME. Therefore, DDF did not activate its Systems Network Architecture (SNA) or Advanced Program to Program Communication (APPC) support.

If an LU name is configured, the *luname* field contains the fully qualified LU name, which consists of a network ID, a period delimiter, and the LU name. The network ID is determined from VTAM after DDF is started. A network ID of NONE indicates that DDF is not yet started, or initialization with VTAM failed.

genericlu

The DDF generic LU name, as recorded in the BSDS. The generic LU name can be specified by one of the following methods:

- The DB2 GENERIC LUNAME field on the DSNTIPR installation panel
- The GENERIC option in the DDF statement of the DSNJU003 (change log inventory) utility

A value of -NONE indicates one of the following situations:

- A generic LU name is not configured.
- The DDF record in the BSDS already has a value for IPNAME. Therefore, DDF did not activate its Systems Network Architecture (SNA) or Advanced Program to Program Communication (APPC) support.

If a generic LU name is configured and enabled, the *genericlu* field contains the fully qualified generic LU name, which consists of a network ID, a period delimiter, and the generic LU name. The network ID is determined from VTAM after DDF is started. A network ID of NONE indicates that the DDF is not yet started, or initialization with VTAM failed.

The location name, LU name, and generic LU name for DDF are also included in message DSNL004I when DDF is started.

Related information:

DB2 LOCATION NAME field (DB2 Installation and Migration)

DB2 NETWORK LUNAME field (DB2 Installation and Migration)

DB2 GENERIC LUNAME field (DB2 Installation and Migration)

Syntax and options of the DSNJU003 control statement (DB2 Utilities)

"DSNL004I" on page 307

DSNL080I

DSNL084I: TCP/IP port information:

The location information is followed by information about the TCP/IP port that is associated with DDF:

`DSNL084I TCPPORT=`*tcp-port* `SECPORT=`*secure-port* `RESPORT=`*resync-port* `IPNAME=`*ip-name*

tcp-port The TCP/IP port number for the SQL listener, as recorded in the BSDS. This port number can be specified by one of the following methods:

- The DRDA PORT field on the DSNTIP5 installation panel
- The PORT option in the DDF statement of the DSNJU003 (change log inventory) utility

A value of 0 (zero) indicates that an SQL listener port was not specified, and DDF is not configured to use TCP/IP network protocols.

A value of -NONE indicates that DB2 does not accept connections on this port. One possible reason that DB2 does not accept connections is that the MAXDBAT subsystem parameter is set to 0 (zero). The current MAXDBAT value is listed in the DSNL090I message, which is included as part of the DISPLAY DDF output when the DETAIL option is specified.

secure-port
> The TCP/IP port number for the secure SQL listener, as recorded in the BSDS. This port number can be specified by one of the following methods:
>
> - The SECURE PORT field on the DSNTIP5 installation panel
> - The SECPORT option in the DDF statement of the DSNJU003 (change log inventory) utility
>
> A value of 0 (zero) indicates both of the following conditions:
>
> - A secure SQL listener port was not specified.
> - DDF is not configured to accept secure inbound connections by using the Secure Socket Layer (SSL) protocol through this port number.
>
> If a secure port is not specified, remote connections can still use the TCP/IP port for the SQL listener (*tcp-port*) and use the SSL protocol on that port. However, DDF does not validate whether the connection uses SSL protocol.
>
> A value of -NONE indicates that DB2 does not accept connections on this port. One possible reason that DB2 does not accept connections is that the MAXDBAT subsystem parameter is set to 0 (zero). The current MAXDBAT value is listed in the DSNL090I message, which is included as part of the DISPLAY DDF output when the DETAIL option is specified.

resync-port
> The TCP/IP port number for the two-phase commit resynchronization (resync) listener, as recorded in the BSDS. This port number can be specified by one of the following methods:
>
> - The RESYNC PORT field on the DSNTIP5 installation panel
> - The RESPORT option in the DDF statement of the DSNJU003 (change log inventory) utility
>
> A value of 0 (zero) indicates that a resynchronization listener port was not specified, and DDF is not configured to use TCP/IP network protocols.

ip-name The IPNAME value, as recorded in the BSDS. For a description of the IPNAME value, see the DSNJU003 syntax information.

This value is unique to this DDF.

If this DB2 subsystem is a member of a data sharing group, the IPNAME value is unique to that data sharing group. If all the members of a data sharing group activate only their TCP/IP communications support, all members of that group must be defined with the same IPNAME value.

This TCP/IP port information for DDF is also included in message DSNL004I, which is displayed when DDF is started.

Related information:
> DRDA PORT field (DB2 Installation and Migration)

SECURE PORT field (DB2 Installation and Migration)

RESYNC PORT field (DB2 Installation and Migration)

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

"DSNL004I" on page 307

Syntax and options of the DSNJU003 control statement (DB2 Utilities)

DSNL085I: IP address:

The TCP/IP port information is followed by the IP address that is associated with DDF:

DSNL085I IPADDR=*ip-address*

ip-address
> The IP address of DDF.
>
> If the subsystem is a member of a data-sharing group, the IP address is a sysplex-wide IP address. DB2 expects this sysplex-wide IP address to be configured as a distributed dynamic virtual IP address (distributed DVIPA). A distributed DVIPA distributes connections across the group.
>
> A value of -NONE indicates that one or more of the following conditions is true:
> - DDF is not configured to use TCP/IP network protocols. See the *tcp-port* value in message DSNL084I.
> - DDF is not started.
> - TCP/IP initialization failed.
>
> If both IPv4 and IPv6 addresses are configured, message DSNL085I is displayed twice, once for each protocol type.
>
> If the subsystem is not a member of a data sharing group and the listener ports are defined, DDF obtains its IP address from one of the following places:
> - The IP address that is recorded in the BSDS
> - The information that is returned by a z/OS UNIX System Services getaddrinfo call
>
> If the subsystem is a member of a data sharing group and the listener ports are defined, DDF obtains the location's sysplex-wide IP address from one of the following places:
> - The IP address that is recorded in the BSDS
> - The IP address that is bound in the TCP/IP PORT definition statement for the DRDA SQL port
>
> If the IP address is recorded in the BSDS, DDF accepts connections on any address. Otherwise, DDF accepts connections on only the IP addresses that are displayed in messages DSNL085I and DSNL089I.
>
> **Related information:**
> > Communicating with data sharing groups (DB2 Data Sharing Planning and Administration)
> >
> > getaddrinfo (BPX1GAI, BPX4GAI) -- Get the IP address and information for a service name or location (z/OS UNIX System Services Programming: Assembler Callable Services Reference)
> >
> > Starting TCP/IP support (DB2 Installation and Migration)

DSNL086I: TCP/IP domain name:

The IP address is followed by the TCP/IP domain name that is associated with DDF:

DSNL086I *domain-type* DOMAIN=*domain-name*

domain-type
> The type of domain name that is displayed, which can be one of the following values:
>
> **SQL** The name that is used by TCP/IP to route requests to the location. The name is associated with the IP address that is displayed in message DSNL085I.
>
> **RESYNC**
> > The name that is used by TCP/IP to route requests to DDF. This information is displayed only if DDF is a member of a DB2 data sharing group. This name is associated with the IP address that is displayed in message DSNL089I.

DSNL080I

domain-name
> The domain name that is associated with the domain type. A value of -NONE indicates that TCP/IP is not configured or that TCP/IP initialization failed.

This domain information is also included in messages DSNL519I and DSNL004I.

Related information:
> "DSNL519I" on page 378
>
> "DSNL004I" on page 307
>
> Communicating with data sharing groups (DB2 Data Sharing Planning and Administration)

DSNL087I and DSNL088I: Aliases:

If aliases were defined for the DDF location by using the MODIFY DDF command or by using the DSNJU003 (change log inventory) utility, the output includes alias information:

```
DSNL087I ALIAS          PORT     SECPORT     STATUS
DSNL088I alias          port     secure-port status
```

alias   An alias name.

> You can manage static aliases by using the DSNJU003 (change log inventory) utility. You can manage dynamic aliases by using the MODIFY DDF command and specifying the ALIAS option.

port   The TCP/IP port that is associated with the alias. A value of blanks indicates that a port was not specified for the alias. However, the alias can still be accessed by using other active DRDA SQL ports.

> This value is ignored for statically defined location aliases in non-data-sharing environments.

secure-port
> The TCP/IP secure port that is associated with the alias. A value of blanks indicates that a secure port was not specified for the alias. However, the alias can still be accessed by using other active DRDA secure SQL ports.

> This value is ignored for statically defined location aliases in non-data-sharing environments.

status   The operational status of the alias, which can be one the following values:

> **STARTG**
>> The alias is starting. The MODIFY DDF command was issued with the ALIAS keyword to start the alias, and the start operation is in progress.
>
> **STARTD**
>> The alias is started or ready to start with the alias port. DB2 might be processing existing connections to the alias and is accepting new connection requests to the alias by using the alias port. Otherwise, DB2 starts accepting new connection requests to the alias by using the alias port when the following conditions become true:
>> - DDF is started.
>> - The MAXDBAT subsystem parameter is set to a value greater than 0.
>
> **STOPG**
>> The alias is stopping. The MODIFY DDF command with the ALIAS keyword was issued to stop an alias, and the stop operation is in progress.
>
> **STOPD**
>> The alias is stopped. DB2 does not accept new connection requests to the alias by using the alias port. However, DB2 can still accept requests to the alias by using the group SQL port if DDF is active and listening on it. This value applies to only those aliases that are defined and managed dynamically by the MODIFY DDF command with the ALIAS keyword.
>
> **CANCLG**
>> The alias is canceling. The MODIFY DDF command with the ALIAS keyword was issued to cancel an alias, and the cancel operation is in progress.
>
> **CANCLD**
>> The alias is canceled. DB2 canceled all existing database access threads (DBATs) that were processing connections to the alias. DB2 does not accept new connection requests to the alias by

using the alias port. However, DB2 can still accept requests to the alias by using the group SQL port if DDF is active and listening on it. This value applies to only those aliases that are defined and managed dynamically by the MODIFY DDF command with the ALIAS keyword.

STATIC

The alias is static. DB2 accepts new connection requests when DB2 is available to accept TCP/IP connection requests. This option applies to only those aliases that were created by the DSNJU003 utility. Such aliases cannot be defined or managed dynamically by the MODIFY DDF command.

Related information:

-MODIFY DDF (DB2) (DB2 Commands)

DSNJU003 (change log inventory) (DB2 Utilities)

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

DSNL089I: Member IP address:

If DDF is part of a data sharing group and all of the following conditions are true, the member IP address is displayed:

- DDF is configured to use TCP/IP network protocols. See the *tcp-port* value in message DSNL084I.
- DDF is started.
- TCP/IP initialization was successful.

```
DSNL089I MEMBER IPADDR= ip-address
```

ip-address

The IP address that is associated with the specific member of the group or with a specific location alias that represents a member or subset of members. This value is the IP address that is returned to a client in the server list for the purposes of workload balancing.

If both IPv4 and IPv6 addresses are configured, message DSNL089I is displayed twice, once for each protocol type.

DDF obtains its member IP address from one of the following places:

- The IP address that is specified on the TCP/IP PORT statement by using the BIND keyword
- The IP address that is recorded in the BSDS
- The information that is returned by a z/OS UNIX System Services getaddrinfo call

Related information:

Communicating with data sharing groups (DB2 Data Sharing Planning and Administration)

getaddrinfo (BPX1GAI, BPX4GAI) -- Get the IP address and information for a service name or location (z/OS UNIX System Services Programming: Assembler Callable Services Reference)

Starting TCP/IP support (DB2 Installation and Migration)

DSNL090I: Subsystem parameter information (CMTSTAT, CONDBAT, and MAXDBAT):

If the DETAIL option was specified on the DISPLAY DDF command, the output also includes subsystem parameter information that is associated with DDF:

```
DSNL090I DT=dt  CONDBAT=    condbat MDBAT=  mdbat
```

dt   The DDF thread value. This value corresponds to the CMTSTAT subsystem parameter. *dt* can have one of the following values:

A   DDF is configured with DDF THREADS ACTIVE.

I   DDF is configured with DDF THREADS INACTIVE.

condbat The maximum number of inbound connections. This value corresponds to the CONDBAT subsystem parameter.

mdbat   The maximum number of database access threads (DBATs). This value corresponds to the MAXDBAT subsystem parameter.

This value determines the maximum number of *active slots*, which means the maximum number of concurrent active DBATs that can potentially execute SQL.

DSNL080I

Related information:

DDF THREADS field (CMTSTAT subsystem parameter) (DB2 Installation and Migration)

MAX REMOTE CONNECTED field (CONDBAT subsystem parameter) (DB2 Installation and Migration)

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

DSNL091I: Subsystem parameter information (MAXCONQN and MAXCONQW):

If the DETAIL option was specified on the DISPLAY DDF command, the output also includes subsystem parameter information that is associated with DDF:

```
DSNL091I MCONQN=maxconqn-value MCONQW=maxconqw-value
```

maxconqn-value
> The depth of the queue of connections that are waiting for a DBAT to process the next request. When this value is reached, and the threshold that is specified by the MAXDBAT subsystem parameter is also reached, DB2 closes the oldest waiting client connection.
>
> This value corresponds to the MAXCONQN subsystem parameter.

maxconqw-value
> The maximum amount of time that a client connection waits for a thread to process its request. DB2 attempts to close any connection that waits longer than this value.
>
> This value corresponds to the MAXCONQW subsystem parameter.

Related information:

CONN QUEUE MAX DEPTH field (MAXCONQN susbsystem parameter) (DB2 Installation and Migration)

CONN QUEUE MAX WAIT field (MAXCONQW subsystem parameter) (DB2 Installation and Migration)

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

DSNL092I and DSNL093I: DBAT and connection information:

If the DETAIL option was specified on the DISPLAY DDF command, the output also includes the following information:

- The number of active, inactive, queued, and disconnected database access threads (DBATs)
- The number of queued and inactive connections

```
DSNL092I ADBAT=adbat QUEDBAT=quedbat INADBAT=inadbat CONQUED=conqued
DSNL093I DSCDBAT=dscdbat INACONN=inaconn
```

adbat   The current number of DBATs. This number includes DBATs that are actively associated with work for remote connections and disconnected DBATs.

> This value increases as new DBATs are created or when inactive DBATs become active. This value decreases as DBATs terminate or become inactive or as disconnected DBATs terminate. When this value reaches or exceeds the MAXDBAT value, newly attached connections or new work on existing inactive connections or threads might be queued. This queueing occurs if no disconnected DBATs are available to service the work.
>
> The MAXDBAT value is shown in message DSNL090I.

quedbat
> The cumulative number of the following new work items that had to wait for a DBAT to become available:
>
> - Newly attached connections
> - New work on inactive connections
> - New work on inactive DBATs
>
> This value is a cumulative counter that is incremented when the MAXDBAT limit is reached. (This value corresponds to the QDSTQDBT field in IFCID 1.) The *quedbat* counter is reset only when this DB2 subsystem is restarted.
>
> A non-zero value suggests that performance and throughput might be affected. If so, consider increasing the MAXDBAT value.

The MAXDBAT value is shown in message DSNL090I.

inadbat   The current number of inactive DBATs (which were previously called type 1 inactive threads). This value applies only if the *dt* value in message DSNL090I indicates that DDF INACTIVE support is enabled.

For more information about any DBATs that are included in this number, look at the output from the DISPLAY THREAD command with the TYPE(INACTIVE) option. The DISPLAY THREAD output begins with message DSNV401I.

conqued

The current number of connection requests that are queued and are waiting to be serviced. This value applies only if the *dt* value in message DSNL090I indicates that DDF INACTIVE support is enabled.

These connections can be newly attached connections or inactive connections for which a new request arrived from the client. The number of inactive connections is shown in message DSNL093I.

These requests cannot be processed when one of the following conditions occur:

- The maximum number of DBATs is reached, and no disconnected DBATs exist to process the request.

  The maximum number of threads is reached when the *adbat* value is equal to or greater than MAXDBAT value. (The MAXDBAT value is shown in message DSNL090I.) When this situation occurs, processing continues for these queued connections when another active DBAT releases its slot either by terminating or by moving to the inactive state.

- DDF is suspended or suspending. See the *status* value in message DSNL081I. Processing continues when DDF is started.

dscdbat   The current number of disconnected DBATs. This value applies only if the *dt* value in message DSNL090I indicates that DDF INACTIVE support is enabled.

This value indicates the number of DBAT pool threads. DBAT pool threads are the database access threads that are available to service queued connections requests. The DBATs that are included in this number are also accounted for in the *adbat* value.

Over time, disconnected DBATs can hold allocation duration locks. An *allocation duration lock* is a lock that persists across commit boundaries. This type of lock requires the connection to be deallocated or the thread to be enabled if it is inactive. You can release these allocation duration locks by terminating these DBATs. You can terminate disconnected DBATs by using one of the following methods:

- Issue the STOP DDF command with the MODE(SUSPEND) option. Stopping DDF with MODE(SUSPEND) terminates these disconnected DBATs without affecting current or new connections. However, activity on these current or new connections is not processed until DDF is started.

- Change the value of the POOLINAC subsystem parameter. This parameter controls the length of time after which unused disconnected DBATs are to be automatically terminated.

inaconn

The current number of inactive connections (which were previously called type 2 connections). This value applies only if the *dt* value in message DSNL090I indicates that DDF INACTIVE support is enabled.

This value represents connections with clients where the client last ended a unit of work either by a commit or rollback operation. This operation caused DB2 to change the state of the connection to inactive, which means that the DBAT is separated from the connection. The connection becomes inactive while the DBAT is made available to service new client connections or to service new requests on existing client connections.

This value is not included in the *inadbat* value. When a new request arrives from the client system on the connection, the connection is then moved to the *conqued* value.

For more information about any connections that are included in this number, look at the output from the DISPLAY THREAD command with the TYPE(INACTIVE) option. The DISPLAY THREAD output begins with message DSNV401I.

Related information:

POOL THREAD TIMEOUT field (POOLINAC subsystem parameter) (DB2 Installation and Migration)

-DISPLAY THREAD (DB2) (DB2 Commands)

-STOP DDF (DB2) (DB2 Commands)

"DSNV401I" on page 942

DSNL080I

DSNL094I: Closed client connections:

The output also includes information about connections that were closed if both of the following conditions are true:
- The DETAIL option was specified on the DISPLAY DDF command
- The DB2 subsystem is a member of a data sharing group

```
DSNL094I WLMHEALTH=health-value CLSDCONQN=number-closed CLSDCONQW=wait-closed
```

health-value
> The health indicator value that was last reported to WLM for the DB2 subsystem. This value is a number from 0 to 100.

number-closed
> The number of client connections that were closed because the value of the MAXCONQN subsystem parameter was reached since DDF was last started.

wait-closed
> The number of client connections that were closed because the value of the MAXCONQW subsystem parameter was reached since DDF was last started.

Related information:

> IWM4HLTH – Setting Server Health Indicator (MVS Programming: Workload Management Services)

> CONN QUEUE MAX DEPTH field (MAXCONQN susbsystem parameter) (DB2 Installation and Migration)

> CONN QUEUE MAX WAIT field (MAXCONQW subsystem parameter) (DB2 Installation and Migration)

DSNL096I: Detailed information for a specific alias:

If the ALIAS and DETAIL options were specified on the DISPLAY DDF command, the output also includes detailed information about the specific location alias:

```
DSNL096I  ADBAT=active-dbats CONQUED=concurrent-queued TCONS=total-connections
```

active-dbats
> The number of active DBATs that are currently processing requests on behalf of the specified alias.

concurrent-queued
> The number of connection requests that are currently queued and waiting to be serviced on behalf of the specified alias.

total-connections
> The total number of remote connections that are currently associated with the alias.

DSNL100I, DSNL101I, DSNL102I: Server list information:

The output also includes server list information if the following conditions are true:
- The DETAIL option was specified on the DISPLAY DDF command
- DDF is part of a data sharing group
- DDF is configured to use TCP/IP (If the value of *tcp-port* in message DSNL084I is NONE, DDF is not configured to use TCP/IP.)

The following server list entry information is displayed for each DDF location or alias that registered to WLM as part of the data sharing group:

```
DSNL100I  LOCATION SERVER LIST:
DSNL101I  WT     IPADDR        IPADDR
DSNL102I  weight ipv4-address ipv6-address
```

Message DSNL102I is repeated for each server list entry.

The server list information is provided to remote clients that are balancing the sysplex workload. The server list information might represent a single member of the data sharing group or a location alias that represents either a single member or a subset of members. This information influences which DDF is sent future work. DDF responds with an updated server list when DDF generates the response to a connection request or a connection reuse request.

weight   The WLM weight for the DDF location or location alias that is identified by the IP addresses. If no value is
           displayed, WLM indicates that DDF has no capacity and the entry is not included in the server list that is
           provided to the remote clients.

ipv4-address
           The IPv4 address that is used to identify a DDF location or location alias that is registered to WLM. The
           address is in colon-hexadecimal format. If no IPv4 address is configured, this IP address is not displayed.

ipv6-address
           The IPv6 address that is used to identify a DDF location or location alias that is registered to WLM. The
           address is in colon-hexadecimal format. If no IPv6 address is configured, this IP address is not displayed.

Related information:
    Communicating with data sharing groups (DB2 Data Sharing Planning and Administration)

DSNL105I and DSNL106I: DDF options:

The output also includes a list of the DDF option values:

```
DSNL105I  CURRENT DDF OPTIONS ARE:
DSNL106I  option-name = option-value
```

One instance of message DSNL106I is displayed for each option.

option-name
           The DDF option.

option-value
           The value of the DDF option.

DSNL099I: End of output:

The DISPLAY DDF output ends with the following message:

```
DSNL099I  csect  DISPLAY DDF REPORT COMPLETE
```

csect-name
           The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Related reference:

▶ -DISPLAY DDF (DB2) (DB2 Commands)

DSNL081I   **STATUS=***status*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message
"DSNL080I" on page 326.

DSNL082I   **LOCATION LUNAME GENERICLU**

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message
"DSNL080I" on page 326.

DSNL083I   *location luname genericlu*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message
"DSNL080I" on page 326.

DSNL084I   **TCPPORT=***tcp-port* **SECPORT=***secure-port* **RESPORT=***resync-port* **IPNAME=***ip-name*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message
"DSNL080I" on page 326.

DSNL085I   **IPADDR=** *ip-address*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL086I   *domain-type* **DOMAIN=***domain-name*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL087I   **ALIAS PORT SECPORT STATUS**

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL088I   *alias port secure-port status*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL089I   **MEMBER IPADDR=** *ip-address*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL090I   **DT=***dt* **CONDBAT=***condbat* **MDBAT=***mdbat*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL091I   **MCONQN=***maxconqn-value* **MCONQW=***maxconqw-value*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL092I   **ADBAT=***adbat* **QUEDBAT=***quedbat* **INADBAT=***inadbat* **CONQUED=***conqued*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL093I   **DSCDBAT=***dscdbat* **INACONN=***inaconn*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL094I   **WLMHEALTH=***health-value* **CLSDCONQN=***number-closed* **CLSDCONQW=***wait-closed*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL096I   **ADBAT=***active-dbats* **CONQUED=***concurrent-queued* **TCONS=***total-connections*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL099I   *csect* **DISPLAY DDF REPORT COMPLETE**

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL100I   **LOCATION SERVER LIST:**

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL101I   **WT IPADDR IPADDR**

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL102I   *weight ipv4-address ipv6-address*

Explanation: This message is a section of output from the DISPLAY DDF command, which is part of message "DSNL080I" on page 326.

DSNL105I   **CURRENT DDF OPTIONS ARE:**

Explanation: This message introduces a list of DDF options in response to a DISPLAY DDF command or at DDF startup.

The DISPLAY DDF output is described in message "DSNL080I" on page 326.

DSNL106I   *option-name = option-value*

Explanation: This message displays DDF options in response to a DISPLAY DDF command or at DDF startup.

The DISPLAY DDF output is described in message "DSNL080I" on page 326.

DSNL199I   **NO ALIAS FOUND**

Explanation: The DISPLAY DDF command was issued with the ALIAS keyword, but the specified alias was not found.

System action: Processing continues.

User response: Issue the DISPLAY DDF command to verify that the specified alias exists and reissue the DISPLAY DDF ALIAS command with the correct alias name.

Related reference:

 -DISPLAY DDF (DB2) (DB2 Commands)

DSNL200I   **DISPLAY LOCATION REPORT FOLLOWS-**

Explanation: This message displays output from the DISPLAY LOCATION command.
- Output heading
- No locations found
- Truncated output
- Detail output
- Output conclusion

Output heading: The output heading is followed by information about connections with remote locations:

LOCATION
    The identifier for the remote location.
- For TCP/IP requester connections, the location is the IP address concatenated with the port number of the remote location. For TCP/IP server connections, the location is the IP address of the remote location.

- For SNA connections, the location is the LU name of the remote location.

PRDID An alphanumeric string that identifies the product. The format is *pppvvrrm*:

    ppp   A 3-character code that identifies the product:

AQT	IBM DB2 Analytics Accelerator for z/OS
ARI	DB2 for VSE & VM
DSN	DB2 for z/OS
JCC	IBM Data Server Driver for JDBC and SQLJ
QSQ	DB2 for i
SQL	DB2 for Linux, UNIX, and Windows

    vv   A 2-digit number that identifies the product version number, such as 09 or 10.

    rr   A 2-digit number that identifies the product release level, such as 01 or 05.

    m   A 1-digit number that identifies the product modification level, such as 0 or 1.

T The connection type with the remote location. Expected values are:

    R   A requester connection from the local subsystem that is accessing a remote system.

    S   A server connection from a remote system that is accessing the local subsystem.

ATT and CONNS
> The type and number of connections with the partner location. When the ATT column is blank, the value of CONNS is the total number of connections with the partner location. The ATT column is blank unless the DETAIL option is specified.
>
> The meanings of the ATT values are:

    AES   The connection uses AES encryption.

    CBF   The connection is a secondary connection for an application that uses package-based continuous block fetch.

    IPS   The connection uses Internet Protocol security (IPSec).

    TLS   The connection uses a secure sockets layer (SSL) that is provided by Application Transparent-Transport Layer Security (AT-TLS).

    TRS   The connection is a trusted connection.

    WLB   The connection is from a client that is using Sysplex workload balancing.

    XA   The connection is part of a distributed transaction.

No locations found: When the DISPLAY LOCATION command is unable to find any locations that are associated with the specified list of locations, a message line indicates that no locations were found.

If all locations were specified, there is no distributed activity. If a location list was specified, reissue the DISPLAY LOCATION command with the LOCATION(*) parameter to display information for all locations.

Truncated output: If the output was generated in response to a command from an MVS console and more than 254 response messages were generated, the output is truncated. A message line indicates this truncation.

You can reissue the DISPLAY LOCATION command and specify a specific location or list of locations to see complete output for that context.

Detail output: When a DISPLAY LOCATION command is issued with the DETAIL option, the output might contain additional lines for each remote location.

The output might contain one or more lines that provide information about the number of connections from or to remote systems that have certain attributes:

LOCATION
> The identifier for the remote location.

- For TCP/IP requester connections, the location is the IP address concatenated with the port number of the remote location. For TCP/IP server connections, the location is the IP address of the remote location.
- For SNA connections, the location is the LU name of the remote location.

PRDID An alphanumeric string that identifies the product. The format is *pppvvrrm*:

    ppp   A 3-character code that identifies the product:

        AQT   IBM DB2 Analytics Accelerator for z/OS

        ARI   DB2 for VSE & VM

        DSN   DB2 for z/OS

        JCC   IBM Data Server Driver for JDBC and SQLJ

        QSQ   DB2 for i

        SQL   DB2 for Linux, UNIX, and Windows

    vv   A 2-digit number that identifies the product version number, such as 09 or 10.

    rr   A 2-digit number that identifies the product release level, such as 01 or 05.

    m   A 1-digit number that identifies the product modification level, such as 0 or 1.

T   The connection type with the remote location. Expected values are:

    R   A requester connection from the local subsystem that is accessing a remote system.

    S   A server connection from a remote system that is accessing the local subsystem.

ATT and CONNS
The number of connections that are associated with the remote location and that have the attribute that is identified by the ATT column. Because more than one attribute might apply to the same connection, the total number of connections is not the sum of the numbers for the different attributes. The expected values for ATT are:

    AES   Using security credentials with AES encryption.

    IPS   Using IPSec.

    TLS   SSL using Application Transparent-Transport Layer Security (AT-TLS).

    TRS   Using trusted context.

    WLB   Using workload balancing connections.

    XA   Using XA transaction manager processing.

The output might also include lines that contain detailed information about the connections that are associated with the DB2 system thread. A line that is identified by the partial message number L203 provides descriptive headings. The lines that follow are identified by the partial message number L204.

SYSTASK
Describes the DB2 system threads. The expected values are:

SYSPLX-*x*
Indicates SNA-related SYSPLEX processing where DB2 systems periodically create connections to a remote data sharing group to obtain WLM balancing information that is related to all members of the group.

The expected values of *x* are:

    O   The local DB2 system is contacting a remote member of a data sharing group to request WLM balancing information.

    I   The local DB2 system is receiving a connection that requests WLM balancing information.

RESYNC
A connection that is used to resynchronize a unit of work that encountered a previous thread or communication error. One line is displayed for each thread that needs resynchronization with the remote site.

DSNL203I

SESSID
The identifier of the session:
• For SNA network connections, the VTAM-defined session instance identifier of the session on which the
conversation is executing. If the session identifier is not applicable, this field contains zeros.

You can use the VTAM DISPLAY NET,ID=*db2-luname*,SCOPE=ACT command to obtain the full session ID.
The DISPLAY NET command lists all sessions (SID) for the DB2 logical unit. Scan the DISPLAY NET
output for the complete SID.

• For Internet Protocol network connections, a string in the format *local:remote*, where:

local   Specifies the local DB2 system's TCP/IP port number.

remote   Specifies the remote partner's TCP/IP port number.

A   The connection activity:

N   The connection is active in the network.

W   The connection is suspended in DB2 and is waiting for VTAM or TCP/IP notification that the
function is complete.

blank   No specific activity condition exists.

ST   The status of the connection:

A   The connection is in allocation.

C   Session limits are being negotiated (CNOS) with the SNA partner before connection allocation.

D   The connection is in deallocation.

R   Receiving.

S   Sending.

blank   No specific status condition exists.

TIME The timestamp of the last message that is sent or received on the connection, in the format *yydddhhmmssth*,
where

yy   The last two digits of the year.

ddd   The number of the day in the year. For example, February 1 is represented as 032.

hhmmssth
The time of day in 24-hour format, to hundredths of the second.

Output conclusion: The following line identifies the end of the output of the DISPLAY LOCATION command:
`DISPLAY LOCATION REPORT COMPLETE`

System action: Processing continues.

User response: No action is required.

Related tasks:

➡ Displaying information about connections with other locations (DB2 Administration Guide)

Related reference:

➡ -DISPLAY LOCATION (DB2) (DB2 Commands)

DSNL203I   **L203-SYSTASK SESSID A ST TIME**

Explanation: This message forms the heading for the listing of DB2 system threads that communicate with other
locations. This message is issued when a detail display is specified on the DISPLAY LOCATION command. For
descriptions of the message contents, see "DSNL200I" on page 337.

The message number and subsystem recognition character are not included in this message. However, L203 is
included in the message text to identify the message.

System action: Processing continues.

Related reference:

⮕ -DISPLAY LOCATION (DB2) (DB2 Commands)

Related information:

"DSNL200I" on page 337

DSNL204I   **L204-**_system-task session-ID activity status time_

Explanation: This message follows the DSNL203I message for a remote location when a detail display is specified on the DISPLAY LOCATION command. This message is displayed for each conversation owned by a DB2 system thread that communicates with the site. For descriptions of the message contents, see "DSNL200I" on page 337.

The message number and subsystem recognition character are not included in this message. However, L204 is included in the message text to identify the message.

System action: Processing continues.

Related reference:

⮕ -DISPLAY LOCATION (DB2) (DB2 Commands)

Related information:

"DSNL200I" on page 337

DSNL206I   _csect-name named-command_ **COMMAND INVALID,** _reason_

Explanation: The specified command could not be processed.

csect-name
         The name of the control section that issued the message.

named-command
         The command that could not be processed.

reason   The reason that the command could not be processed.

System action: Processing continues.

DSNL300I   _csect-name_ **MODIFY DDF REPORT FOLLOWS:**

Explanation: Processing of the MODIFY DDF command has begun. Additional messages about the command follow this message. Message DSNL301I is issued when command processing is complete.

csect-name
         The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -MODIFY DDF (DB2) (DB2 Commands)

DSNL301I   _csect-name_ **MODIFY DDF REPORT COMPLETE**

Explanation: Processing of the MODIFY DDF command is complete.

csect-name
         The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Related reference:

⬀ -MODIFY DDF (DB2) (DB2 Commands)

DSNL302I   *option-name* **IS SET TO** *option-value*

Explanation: A MODIFY DDF command has changed an option.

option-name
>   The name of the DDF processing option.

option-value
>   The new value of the named processing option. When *option-name* is PKGREL, possible values are:

>   **BNDOPT**
>   >   Packages are allocated based on the RELEASE bind option value.

>   >   If a database access thread (DBAT) has processed RELEASE(DEALLOCATE) packages at some point, then the thread will remain active with its connection when a unit of work finishes. If the DBAT was to have been pooled and the connection made inactive, an accounting record is made and the current WLM enclave is deleted.

>   >   Access threads that are not actively processing any units of work might continue to hold package allocation locks while waiting for the next request.

>   **COMMIT**
>   >   Packages are always allocated under RELEASE(COMMIT) rules regardless of their package RELEASE bind option value.

>   >   When an access thread has completed a unit of work, and the DBAT can be pooled, DDF pools the DBAT, makes an accounting record, deletes the current WLM enclave, and inactivates the connection. Pooled access threads release all package allocation locks.

>   >   If the CMTSTAT subsystem parameter is set to ACTIVE, then the message will always indicate that PKGREL IS SET TO COMMIT, regardless of the value specified on the command.

System action: Processing continues.

User response: No action is required.

Related reference:

⬀ -MODIFY DDF (DB2) (DB2 Commands)

DSNL303I   **THE MODIFY DDF COMMAND FAILED WITH RC=**odes*return-code* **and RSN=**odes*reason-code*

Explanation: A BSDS error was encountered.

return-code
>   DB2 return code from the BSDS function.

reason-code
>   DB2 reason code from the BSDS function.

System action: The command is not processed.

User response: Analyze the return code and reason code to determine what type of error occurred. Refer to messages DSNJ107I and DSNJ108I for more information.

DSNL304I   **THE ALIAS** *alias-name* **ALREADY EXISTS.**

Explanation: The MODIFY DDF command was issued to add an ALIAS but the requested alias name already exists and the alias cannot be added.

alias-name
>   The name of the location alias.

System action: The command is not processed.

User response: Specify an alias that does not already exist. You can use the output from DISPLAY DDF command to see the list of aliases that are currently defined.

DSNL305I   **THE ALIAS** *alias-name* **DOES NOT EXIST.**

Explanation: The MODIFY DDF command was issued against an alias that does not exist.

alias-name
      The name defines the location alias.

System action: The command is not processed.

User response: Specify an alias that already exists. You can use the output from the DISPLAY DDF command to see the list of aliases that are currently defined.

DSNL306I   **THE MAXIMUM NUMBER OF SUPPORTED ALIASES HAS BEEN EXCEEDED.**

Explanation: The modify DDF command was issued to add a new alias, but the new alias would exceed the maximum number of aliases that can be defined. You can define as many as 40 aliases by using the MODIFY DDF command.

System action: The command is not processed.

User response: Delete unneeded aliases before adding additional aliases. You can use the output from the DISPLAY DDF command to see the list of currently defined aliases.

DSNL307I   *invalid-value* **IS NOT A VALID PARAMETER FOR** *keyword***.**

Explanation: An invalid parameter value is specified for the MODIFY DDF command.

invalid-value
      The invalid value that was specified.

keyword The option keyword for which the invalid value was specified.

      IPv4   *invalid-value* is not in the dotted decimal format.

      IPv6   *invalid-value* is not in the colon hexadecimal format.

      PORT   *invalid-value* is not a decimal number between 1 and 65535, including 65535.

      SECPORT
            invalid-value is not a decimal number between 1 and 65535, including 65535.

      ALIAS *invalid-value* is not a valid alias name. The alias name must meet all of the following requirements:
            • Contains a maximum of 16 characters
            • Contains only letters (excluding alphabetic extenders), numbers, or the underscore character.
            • Begins with a letter.

System action: The command is not processed.

User response: Reissue the command and specify a value in the valid format for the keyword that you specified.

DSNL308I   **NO** *address-port* **IS DEFINED FOR THE ALIAS** *alias-name***.**

Explanation: The MODIFY DDF command was issued to delete an IP address or port , but an IP address or port is not associated with the ALIAS.

address-port
      One of the following values:

      IPV4 ADDRESS
            No IPv4 address is defined for the alias.

      IPV6 ADDRESS
            No IPv6 address is defined for the alias.

      PORT   No port is defined for the alias.

> **SECPORT**
>> No secure port is defined for the alias.

alias-name
> The name of the location alias.

System action: The command is not processed.

User response: Use the output from the DISPLAY DDF command with the ALIAS keyword to see the list of IP addresses, ports, and secure ports that are associated with each alias.

DSNL309I   MEMBER-SPECIFIC IP ADDRESSES ARE NOT DEFINED.

Explanation: DDF is stopped and its TCP/IP configuration is unknown or DDF is started but configured to receive only from a specific IP address.

To issue a MODIFY DDF command with the ALIAS keyword and the IPV4 or IPV6 option, DDF must be configured to receive from any IP address. To configure DDF to receive from any IP address, the DDF communications record that is defined in the BSDS must be configured with the DDF IPV4 or IPV6 option. The MODIFY DDF command can be used to add or modify alias IP addresses only when an IP address is defined by the DSNJU003 utility and not by the PORT statement of the TCP/IP profile.

System action: The command is not processed.

User response: Use the output from DSNJU004 utility (the DDF communication record) to verify that no IP address was specified in the BSDS. Specify at least one IP address by using the DSNJU003 utility.

DSNL310I   THE ALIAS *keyword keyword-value* IS ALREADY IN USE.

Explanation: The port number specified for the PORT or SECPORT keyword in the MODIFY DDF command is already in use for the location, or for a different alias.

keyword PORT or SECPORT

keyword-value
> The value specified for the PORT or SECPORT keyword.

System action: The command is not processed.

User response: Use the output from the DISPLAY DDF command to see the ports that are defined for each alias and location name.

DSNL311I   THE ALIAS *alias-name* CANNOT BE STARTED.

Explanation: The specified alias cannot be started because it is not stopped or canceled.

alias-name
> The name of the location alias.

System action: The command is not processed.

User response: Use the output from DISPLAY DDF command to check the status of each alias. Only stopped or canceled aliases can be started.

DSNL312I   THE ALIAS *alias-name* CANNOT BE STOPPED OR CANCELED.

Explanation: The MODIFY DDF command was issued with the ALIAS parameter to stop or cancel an alias but the alias is not started.

alias-name
> The name of the location alias.

System action: The command is not processed.

User response: Use the output from DISPLAY DDF command to check the status of each alias. Only started aliases can be stopped, and only started aliases can be canceled.

DSNL313I   **THE ALIAS** *alias-name* **CANNOT BE MODIFIED OR DELETED.**

Explanation: The MODIFY DDF command was issued to modify an attribute of an alias that is already started. The alias must be stopped or canceled before it can be modified or deleted.

alias-name
> The name that specifies the location alias.

System action: The command is not processed.

User response: Stop or cancel the specified alias, and then modify the alias attribute, or delete the alias. Use the output from DISPLAY DDF command (DSNL088I) to see the status of each alias.

DSNL314I   **THE ALIAS** *alias-name* **IS** *state***.**

Explanation: The MODIFY DDF command was issued with the ALIAS keyword to start, stop, or cancel an alias, and the specified operation was successful.

alias-name
> The name of the location alias.

state   Whether the alias was STARTED, STOPPED, or CANCELED.

System action: The system action depends on the action that was successful:

STARTED
> DB2 begins accepting new connections for the alias.

STOPPED
> DB2 stops accepting new connections to the alias. All data base access threads processing connections to the specified alias have finished their processing.

CANCELED
> DB2 stops accepting new connections to the alias and cancels all existing connections.

User response: No action required.

DSNL315I   **A DYNAMICALLY DEFINED ALIAS WITH NAME** *alias-name* **ALREADY EXISTS.**

Explanation: An alias that was defined by the DSNJU003 utility has the same name as an existing alias that was already defined by the MODIFY DDF commands. The new alias that was defined by the DSNJU003 utility is ignored because the alias that was defined by the MODIFY DDF command takes precedence.

System action: The existing alias that was defined by the MODIFY DDF command is used and new the statically defined alias is ignored.

User response: You can define location aliases statically by using the using DSNJU003 utility or dynamically by using the MODIFY DDF command. Dynamic aliases are preferable because you can define and modify them without stopping and restarting DB2 or DDF. When a static and dynamic alias each have the same name, only the dynamic alias is used. If you want to use the statically defined alias instead, use the MODIFY DDF command to delete the dynamic location alias that has the matching name. You can use the output from DISPLAY DDF command to see the list of aliases that are currently defined. The alias status is 'STATIC' for statically defined aliases.

DSNL400E   **INDOUBT THREAD RESOLUTION WITH COORDINATOR** *location* **SHOWS HEURISTIC**
                DAMAGE FOR *luwid=token* **- REMOTE COMMIT VERSUS LOCAL ABORT. INFORMATION**
                RECORDED IN TRACE RECORD WITH IFCID=207 AND IFCID SEQUENCE NUMBER=*seq-no1*
                AND IN TRACE RECORD WITH IFCID=203 AND IFCID SEQUENCE NUMBER=*seq-no2*

Explanation: DB2 has participant responsibility in the logical unit of work identified by 'luwid=token'. Resolution with the coordinator identified by 'location' shows heuristic damage occurred. The DB2 thread representing the logical unit of work was previously resolved using the RECOVER INDOUBT command. The action taken was to abort the thread. Subsequently, communication was reestablished with the coordinator and resolution shows that the coordinator committed the thread.

• The application was told that commit succeeded.

• Any resources updated at the coordinator were committed.

DSNL401E

- Updated resources at this DB2 were rolled back.

location
> The field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no1
> The IFCID sequence number associated with the 207 trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF. This trace record is written when the heuristic damage condition was detected and contains information about the damage condition.

seq-no2
> The IFCID sequence number associated with the 203 trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 203 was not active or it was active with a destination of the resident trace table only. IFCID 203 is a member of statistics class 4, which has a default destination of SMF. This trace record was written when the heuristic decision to commit or abort was made and contains information about the decision, including the names of any downstream participants.

System action: Alert A004 is generated. A trace record whose IFCID=207 is written. Indoubt resolution with the coordinator completes.

System programmer response: Determine the destination of statistics class 4 and locate and print the trace record with SEQUENCE NUMBER='seq-no1'. Call the database administrator.

Database Administrator Action: Correct the damage. This is not an easy task. Since the time of the heuristic action, the data that was aborted with the RECOVER INDOUBT command might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The statistics trace class 4 contains an IFCID 207 trace record. This record contains the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message.
- The DSN1LOGP utility provides a summary report that identifies the table spaces modified by the logical unit of work and any downstream participant locations involved in the logical unit of work.
- DISPLAY THREAD TYPE(INDOUBT) displays the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message. This information continues to appear in the DISPLAY THREAD report until it is purged with the RESET INDOUBT command.

Contact your IBM Support Center for further assistance.

DSNL401E   **INDOUBT THREAD RESOLUTION WITH COORDINATOR** *location* **SHOWS HEURISTIC DAMAGE FOR** *luwid=token* **- REMOTE ABORT VERSUS LOCAL COMMIT. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=207 AND IFCID SEQUENCE NUMBER=***seq-no1* **AND IN TRACE RECORD WITH IFCID=203 AND IFCID SEQUENCE NUMBER=***seq-no2*

Explanation: DB2 has participant responsibility in the logical unit of work identified by 'luwid=token'. Resolution with the coordinator identified by 'location' shows heuristic damage occurred. The DB2 thread representing the logical unit of work was previously resolved with the RECOVER INDOUBT command. The action taken was to commit the thread. Subsequently, communication was reestablished with the coordinator and resolution shows that the coordinator aborted the thread.

- The application was told that commit failed.
- Any resources updated at the coordinator were rolled back.
- Updated resources at this DB2 were committed.

location
> The field contains one of the following:
> - The requester's relational database name (location name)

- The requester's VTAM LU name (<luname>)
- The requester's IP address

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no1
> The IFCID sequence number associated with the 207 trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF. This trace record is written when the heuristic damage condition was detected and contains information about the damage condition.

seq-no2
> The IFCID sequence number associated with the 203 trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 203 was not active or it was active with a destination of the resident trace table only. IFCID 203 is a member of statistics class 4, which has a default destination of SMF. This trace record was written when the heuristic decision to commit or abort was made and contains information about the decision, including the names of any downstream participants.

System action: Alert A004 is generated. A trace record whose IFCID=207 is written. Indoubt resolution with the coordinator completes.

System programmer response: Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no1'. Call the database administrator.

Database Administrator Action: Correct the damage. This is not an easy task. Since the time of the heuristic action, the data that was committed with the RECOVER INDOUBT command might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The statistics trace class 4 contains an IFCID 207 trace record. This record contains the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message.
- The DSN1LOGP utility provides a summary report that identifies the table spaces modified by the logical unit of work, and any downstream participant locations involved in the logical unit of work.
- DISPLAY THREAD TYPE(INDOUBT) displays the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message. This information continues to appear in the DISPLAY THREAD report until it is purged with the RESET INDOUBT command.

Contact your IBM Support Center for further assistance.

DSNL402I   **INDOUBT THREAD RESOLUTION WITH PARTICIPANT** *location* **SHOWS HEURISTIC DAMAGE FOR** *luwid=token* **- REMOTE COMMIT VERSUS LOCAL ABORT. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=207 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 has coordinator responsibility for the logical unit of work identified by 'luwid=token'. Resolution with a participant shows heuristic damage occurred. The participant identified by 'location' manually resolved the indoubt thread identified by 'luwid=token'. The action taken was to commit the thread. This is inconsistent with the abort decision of the coordinator.

- The application was told that commit failed.
- Any resources updated at the participant were committed.
- Updated resources at this DB2 were rolled back.

location
> The field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in

hexadecimal format. If the sequence number is zero, then IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF.

If the location specified refers to a DB2 Linux, UNIX, and Windows system that is running at the Version 9.1 level or later, and that server location has enabled access to non-DB2 databases through the Federated Database or Information Integrator capability, then any updates made to that downstream non-DB2 database will be processed under the control of the DB2 Version 9.1 for Linux, UNIX, and Windows unprotected update capability. This permits changes to be made to this downstream non-DB2 database, even when the changes were being requested through applications running under two-phase commit or protected update control, such as CICS and IMS. During phase one of commit processing, the DB2 V9.1 for Linux, UNIX, and Windows server will have actually committed the changes to the downstream non-DB2 database by the time it replies to this DB2 that it is prepared to accept a commit decision from. However, if during phase one commit negotiation with other downstream database servers and other coordinated resource managers from CICS or IMS, the decision is made to abort the unit of work, this will result in heuristic damage to the data at the non-DB2 database downstream of the DB2 V9.1 for Linux, UNIX, and Windows server. No attempts will be made to resync with the identified remote location server, and a DB2 indoubt thread will not be left in DB2 to reflect this issue.

System action: Alert A004 is generated. A trace record whose IFCID=207 is written. Indoubt resolution with the participant completes.

System programmer response: Call the database administrator.

Database Administrator Action: Inform the database administrator at 'location' that heuristic damage occurred for the logical unit of work with LUWID='luwid'. DB2 was the coordinator for the logical unit of work and made the decision to abort the database updates made by 'luwid'. At 'location' a heuristic decision was made to COMMIT the updates made by 'luwid'.

Information about the heuristic damage continues to appear in the DISPLAY THREAD TYPE(INDOUBT) report until it is purged with the RESET INDOUBT command.

DSNL403I **INDOUBT THREAD RESOLUTION WITH PARTICIPANT** *location* **SHOWS HEURISTIC DAMAGE FOR** *luwid=token* **- REMOTE ABORT VERSUS LOCAL COMMIT. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=207 AND IFCID SEQUENCE NUMBER=**seq-no

Explanation: DB2 has coordinator responsibility for the logical unit of work identified by 'luwid=token'. Resolution with a participant shows heuristic damage occurred. The participant identified by 'location' manually resolved the indoubt thread identified by 'luwid=token'. The action taken was to abort the thread. This is inconsistent with the commit decision of the coordinator.

- The application was told that commit succeeded.
- Any resources updated at the participant were rolled back.
- Updated resources at this DB2 were committed.

location
> This field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A004 is generated. A trace record with IFCID 207 is written. Indoubt resolution with the participant completes.

System programmer response: Call the database administrator.

Database Administrator Action: Inform the database administrator at 'location' that heuristic damage occurred for the logical unit of work with LUWID='luwid'. DB2 was the coordinator for the logical unit of work and made the

decision to COMMIT the database updates made by 'luwid'. At 'location' a heuristic decision was made to back out (ABORT) the updates made by 'luwid'.

Information about the heuristic damage continues to appear in the DISPLAY THREAD TYPE(INDOUBT) report until it is purged with the RESET INDOUBT command.

DSNL404E   **PROTOCOL ERROR DURING INDOUBT THREAD RESOLUTION WITH COORDINATOR** *location* **LUWID=***luwid=token* **INFORMATION RECORDED IN TRACE RECORD WITH IFCID=206 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 has participant responsibility for the logical unit of work identified by 'luwid=token'. The DB2 thread associated with the logical unit of work is indoubt. A protocol error occurred during indoubt resolution with the coordinator identified by 'location'.

location
> Either the relational database name (location-name) or the logical unit name (<luname>) of the coordinator.

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 206 was not active or it was active with a destination of the resident trace table only. IFCID 206 is a member of statistics class 4, which has a default destination of SMF.

System action: The indoubt thread remains indoubt. Alert A003 is generated. A trace record whose IFCID=206 is written.

System programmer response: This is an APARable error, probably at the coordinator, but possibly at the DB2 participant. The DB2 documentation required is contained in the statistics class 4 trace record that was written. Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'.

Call the database administrator.

Database Administrator Action: DB2 does not attempt to automatically resolve the indoubt thread. The indoubt thread must be manually resolved. The commit or abort decision made at the coordinator must be determined so that the same decision can be made at this participant DB2.

Contact the database administrator at the coordinator location to determine whether the thread with LUWID=luwid committed or aborted.

If the coordinator system is another DB2, the following steps can be taken at the DB2 coordinator to determine the commit or abort decision.

1. If the coordinator DB2 system is running and a DB2 conditional restart was not performed since the error was detected at this (the participant) location, ask the database administrator at the DB2 coordinator location to issue one of the following DISPLAY THREAD TYPE(INDOUBT) commands:
   - A -DISPLAY THREAD TYPE(INDOUBT), using the LOCATION keyword and specifying the location name of the local DB2.
   - A -DISPLAY THREAD TYPE(INDOUBT) LUWID(luwid)

   If a thread with LUWID=luwid appears in the display, then the display also includes the commit or abort decision. If no thread with LUWID=luwid appears in the display, the thread was aborted.

2. If the DB2 coordinator is down, or if a conditional restart was performed there, ask the database administrator at the DB2 coordinator to run DSN1LOGP, specifying SUMMARY(ONLY) FILTER LUWID(luwid). The output from DSN1LOGP includes the commit or abort decision.

Use the RECOVER INDOUBT command at this (the participant) location to resolve the indoubt thread.

DSNL405I   **THREAD** *luwid=token* **PLACED IN INDOUBT STATE BECAUSE OF COMMUNICATION FAILURE WITH COORDINATOR** *location*. **INFORMATION RECORDED IN TRACE RECORD WITH IFCID=209 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: During execution of the two phase commit protocol with the coordinator identified by 'location', communication failed. Phase 1 of the protocol completed and the thread is prepared for either commit or abort.

DSNL406I

The thread is placed in the indoubt state and appears in a DISPLAY THREAD TYPE (INDOUBT) report. Periodic attempts are made to reestablish communication with the coordinator for automatic indoubt resolution. If successful, message DSNL407 or DSNL415 is written.

luwid=token
> The global logical unit of work id and the local token that identify the thread.

location
> This field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 209 was not active or it was active with a destination of the resident trace table only. IFCID 209 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A006 is generated. A trace record whose IFCID=209 is written.

Operator response: If message DSNL407 or DSNL415 for the LUWID specified in the DSNL405 message does not appear in a reasonable amount of time, call the system programmer.

System programmer response: Determine the cause of the communication failure and have the problem fixed. DB2 periodically attempts to reestablish communication. If the problem cannot be fixed in a reasonable amount of time, call the database administrator. Manual resolution of the indoubt thread might be necessary to release locked resources.

Database Administrator Action: Take the following actions to resolve this thread using manual means (that is, take heuristic action) if it is necessary to release locks held by this thread.
- Determine the proper commit or abort decision by contacting the database administrator at the coordinator 'location' shown in this message in order to determine whether the thread with LUWID=luwid committed or aborted.

  If the coordinator system is another DB2, the following steps can be taken at the DB2 coordinator to determine the commit or abort decision.

  1. If the coordinator DB2 system is running and a DB2 conditional restart was not performed since the error was detected at this (the participant) location, ask the database administrator at the DB2 coordinator location to issue one of the following DISPLAY THREAD TYPE(INDOUBT) commands:
     - A -DISPLAY THREAD TYPE(INDOUBT), using the LOCATION keyword and specifying the location name of the local DB2.
     - A -DISPLAY THREAD TYPE(INDOUBT) LUWID(luwid)

     If a thread with LUWID=luwid appears in the display, then the display also includes the commit or abort decision. If no thread with LUWID=luwid appears in the display, then the thread was aborted.

  2. If the DB2 coordinator is down or if a conditional restart was performed there, ask the database administrator at the DB2 coordinator to run DSN1LOGP, specifying SUMMARY(ONLY) FILTER LUWID(luwid). The output from DSN1LOGP includes the commit or abort decision.
- Use the RECOVER INDOUBT command to commit or abort the thread.

DSNL406I   **THREAD** *luwid=token* **AT PARTICIPANT** *location* **MAY BE INDOUBT BECAUSE OF A COMMUNICATION FAILURE. DB2 COMMITTED THE THREAD. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=209 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: During execution of the two phase commit protocol with the participant identified by 'location', communications failed. Phase 1 of the protocol completed with the participant, which is prepared to either commit or abort. The decision to commit the thread was made, but cannot be communicated to the participant at this time. The participant is indoubt.

DB2 becomes responsible for indoubt resolution with the participant. This responsibility appears in a DISPLAY THREAD TYPE (INDOUBT) report. Periodic attempts are made to reestablish communication with the participant for automatic indoubt resolution. If successful, message DSNL423 is written.

luwid=token
>   The global logical unit of work id and the local token that identify the thread.

location
>   This field contains one of the following:
>   - The requester's relational database name (location name)
>   - The requester's VTAM LU name (<luname>)
>   - The requester's IP address

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 209 was not active or it was active with a destination of the resident trace table only. IFCID 209 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A006 is generated. A trace record whose IFCID=209 is written.

Operator response: If message DSNL423 for the LUWID specified in message DSNL406 does not appear in a reasonable amount of time, call the system programmer.

System programmer response: Determine the cause of the communication failure and have the problem fixed. DB2 periodically attempts to reestablish communication. If the problem cannot be fixed in a reasonable amount of time, call the database administrator. The participant might need to manually resolve the indoubt state to release locked resources.

Database Administrator Action: If the unit of work with LUWID=luwid is indoubt at 'location', inform the database administrator there that the logical unit of work with LUWID=luwid was committed to assist in manual resolution of the indoubt thread at 'location'.

DSNL407I   AUTOMATIC RESOLUTION OF *luwid=token* WITH COORDINATOR *location* RESULTED IN COMMIT

Explanation: The identified indoubt logical unit of work was automatically resolved by communication with the coordinator location. After automatic resolution, this message is issued if message DSNL405 was previously issued to report the thread as indoubt. This message is not issued if message DSNL405 occurs in one restart of DDF or DB2, and automatic resolution occurs in a different restart of DDF or DB2.

luwid=token
>   The global logical unit of work id and the local token that identify the thread

location
>   This field contains one of the following:
>   - The requester's relational database name (location name)
>   - The requester's VTAM LU name (<luname>)
>   - The requester's IP address

System action: Processing continues normally.

DSNL408I   INDOUBT THREAD RESOLUTION WITH COORDINATOR *location* SHOWS HEURISTIC DAMAGE AT PARTICIPANT *location* FOR *luwid=token* - COORDINATOR COMMIT VERSUS PARTICIPANT ABORT. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=207 AND IFCID SEQUENCE NUMBER=*seq-no*

Explanation: DB2 has participant responsibility in the logical unit of work identified by 'luwid=token' for the coordinator at COORDINATOR 'location' and coordination responsibility in the same logical unit of work for the participant at PARTICIPANT 'location'. Heuristic damage occurred.

- The application was told that commit succeeded.
- Any resources updated at the coordinator and at this DB2 were committed.
- Updated resources at the participant were rolled back.

location
>   This field contains one of the following:
>   - The requester's relational database name (location name)

- The requester's VTAM LU name (<luname>)
- The requester's IP address

luwid=token

> The global logical unit of work id and the local token that identify the thread.

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A004 is generated. A trace record whose IFCID=207 is written. Indoubt resolution with the coordinator completes.

System programmer response: Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'. Call the database administrator.

Database Administrator Action: Inform the database administrator at the coordinator 'location' and at the participant 'location' of the problem.

Correcting the damage at the participant is not an easy task. Since the time of the heuristic action, the data that was aborted using manual means might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. The following information that is available at DB2 might be helpful in correcting the problem:

- The statistics trace class 4 contains an IFCID 207 trace record. This record contains the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message.
- The DSN1LOGP utility provides a summary report that identifies the table spaces modified by the logical unit of work at the local DB2 as well as any downstream participant locations involved in the logical unit of work. This information, while not directly related to the damage at the participant, might be useful in determining the scope of the problem.
- DISPLAY THREAD TYPE(INDOUBT) displays the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message. This information continues to appear in the DISPLAY THREAD report until it is purged with the RESET INDOUBT command.

Contact your IBM Support Center for further assistance.

DSNL409I **INDOUBT THREAD RESOLUTION WITH COORDINATOR** *location* **SHOWS HEURISTIC DAMAGE AT PARTICIPANT** *location* **FOR** *luwid=token* **- COORDINATOR ABORT VERSUS PARTICIPANT COMMIT. INFORMATION RECORDED IN TRACE RECORD WITH IFCID 207 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 has participant responsibility in the logical unit of work identified by 'luwid=token' for the coordinator at COORDINATOR 'location' and coordination responsibility in the same logical unit of work for the participant at PARTICIPANT 'location'. Heuristic damage occurred.

- The application was told that commit failed.
- Any resources updated at the coordinator and at this DB2 were rolled back.
- Updated resources at the participant were committed.

location

> This field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

luwid=token

> The global logical unit of work id and the local token that identify the thread.

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 207 was not active or it was active with a destination of the resident trace table only. IFCID 207 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A004 is generated. A trace record whose IFCID=207 is written. Indoubt resolution with the coordinator completes.

System programmer response: Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'. Call the database administrator.

Database Administrator Action: Inform the database administrator at the coordinator 'location' and at the participant 'location' of the problem.

Correcting the damage at the participant is not an easy task. Since the time of the heuristic action, the data that was committed using manual means might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. This information is available at DB2 and might be helpful in correcting the problem.

- The statistics trace class 4 contains an IFCID 207 trace record. This record contains the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message.

- The DSN1LOGP utility provides a summary report that identifies the table spaces modified by the logical unit of work at the local DB2 and any downstream participant locations involved in the logical unit of work. This information, while not directly related to the damage at the participant, might be useful in determining the scope of the problem.

- DISPLAY THREAD TYPE(INDOUBT) displays the RBA of the first DB2 log record written for the logical unit of work whose LUWID is included in this message. This information continues to appear in the DISPLAY THREAD report until it is purged with the RESET INDOUBT command.

Contact your IBM support center for further assistance.

DSNL410I   WARM START CONNECTION BY PARTNER *location* REJECTED. PARTNER CHANGED AT LEAST 1 OF - OUR LOG NAME *oldourname(newourname)*, THEIR LOG NAME *oldtheirname(newtheirname)*, PA_SUPPORT *oldpa(newpa)*, FLAG_BYTE_SENT *oldfb(newfb)*, CCLUNAME_SENT *oldccls(newccls)*, AND INDOUBT THREADS REQUIRE RESOLUTION. SEE IFCID=205 TRACE RECORD WITH IFCID SEQUENCE NUMBER=*seq-no*.

Explanation: An attempt to make a warm start connection with a partner was rejected because the partner specified a different set of sync point parameters than the ones that were in use when communications were lost. DB2 has knowledge of indoubt threads that involve the partner as either the coordinator or a participant. This error might be a recoverable error if the partner can restart with the original sync point parameters. If this is not possible, then the partner must perform a cold start connection with DB2.

location
> The 'location' with which the connection failed: either the relational database name (location-name) or the logical unit name (<luname>) of the partner.

oldourname
> The log name for this DB2 subsystem. This value is null if the partner did not initiate the log name exchange.

newourname
> The log name for this DB2 subsystem, as remembered by our partner and sent to us in the attempted warm start connection. This value is null if the partner did not initiate the log name exchange.

oldtheirname
> The log name used by the partner during the last connection, as remembered by DB2.

newtheirname
> The log name used by the partner in the attempted warm start connection.

oldpa   The sync point protocol used during the last connection, as remembered by the DB2 (PA - Presumed Abort, PN - Presumed Nothing).

newpa   The sync point protocol used during the last connection, as remembered by the partner (PA - Presumed Abort, PN - Presumed Nothing).

oldfb   The use of the PS header byte 2 during the last connection, as remembered by DB2 (F - used as flag byte, N - not used as flag byte).

newfb   The use of the PS header byte 2 during the last connection, as remembered by the partner (F - used as flag byte, N - not used as flag byte).

DSNL411E

oldccls Whether the luname of the conversation correlator was exchanged as part of the sync point protocol during the last connection, as remembered by DB2. (E - was exchanged, N - was not exchanged).

newccls

Whether the luname of the conversation correlator was exchanged as part of the sync point protocol during the last connection, as remembered by the partner (E - was exchanged, N - was not exchanged).

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 0205 was not active or it was active with a destination of the resident trace table only. IFCID 0205 is a member of statistics class 4, which has a default destination of SMF.

System action: The connection with the partner is rejected. Alert A002 is generated. DB2 retains indoubt knowledge.

System programmer response: Call the database administrator.

Database Administrator Action: Contact the database administrator at the partner 'location' and determine if it is possible for the partner to perform a warm start with same sync point parameters as ours ('oldourname', 'oldtheirname', 'oldpa', 'oldfb', 'oldccls'). If this is possible, the next attempt to connect will succeed.

If this cannot be done, then there are two other possible solutions:

- Have the partner 'location' perform a cold start connection with this DB2 subsystem.
- Use the RESET INDOUBT command with the FORCE option in conjunction with the LOCATION parameter to cause the next connection with the partner to be a cold start connection.

DSNL411E **COLD START CONNECTION BY COORDINATOR** *location* **ACCEPTED. INDOUBT THREADS NEED MANUAL RESOLUTION. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=204 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 has participant responsibility for indoubt threads. The coordinator informed DB2 that it performed a cold start operation and lost all knowledge of indoubt threads. The indoubt threads at this DB2 must be manually resolved with the RECOVER INDOUBT command.

location

This field contains one of the following:

- The requester's relational database name (location name)
- The requester's VTAM LU name (<luname>)
- The requester's IP address

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 204 was not active or it was active with a destination of the resident trace table only. IFCID 204 is a member of statistics class 4, which has a default destination of SMF.

System action: The connection with the partner is accepted. Alert A001 is generated. A trace record whose IFCID=204 is written.

System programmer response: Call the database administrator.

Database Administrator Action: DB2 is a participant with one or more indoubt threads where the coordinator is 'location'. The DBMS at 'location' performed a cold start. The DB2 participant assumes that the coordinator recovery log was lost or damaged and indoubt thread resolution cannot be achieved. There is probably inconsistent data at the coordinator. A list of indoubt threads involving the coordinator 'location' can be displayed using the DISPLAY THREAD command for indoubt type threads by including the LOCATION('location') parameter in the command.

The heuristic decision (that is, to commit or abort the logical unit of work) should be coordinated with any other participants. The existence of other participants might not be easy to determine. The information might be available in the coordinators recovery log even though the coordinator performed a cold start. If the coordinator is a DB2, then it might be possible to execute the DSN1LOGP utility to produce a summary report. If this is possible, the following information might be produced:

- The LUWID of the thread.
- The NID used by IMS or CICS to identify the thread.
- The name of all participants.

- The name of an upstream coordinator if the coordinator is also a participant. This could happen when an application requester accesses a DB2 AS using application-directed access, which in turn accesses another DB2 server with system-directed access. (The local DB2 might be the third system in this scenario.)

At this DB2 participant, all indoubt threads coordinated by the cold starting location must be resolved with the RECOVER INDOUBT command. Locked data remains unavailable until resolution. Threads that were already resolved with this command are discarded. Threads not yet resolved are discarded after resolution with the command.

The commit or abort decision provided using the RECOVER INDOUBT command for a logical unit of work are propagated to all downstream participants, if any.

DSNL412I **PROTOCOL ERROR DETECTED IN SYNCPOINT REQUEST/RESPONSE FROM COORDINATOR** *location* **LUWID=***luwid=token***. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=208 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 is a participant in the logical unit of work identified by 'luwid=token'. A protocol error occurred during the SNA sync point exchange with the coordinator identified by 'location'. The protocol error fits into one of the following categories:

- An unknown SNA PS header was received.
- An SNA PS header was received in the wrong sequence.
- An invalid LU6.2 verb sequence was used to deliver the SNA PS header.
- A DEALLOCATE TYPE(ABEND_SVC) was received from the coordinator, indicating that the coordinator detected a protocol violation in the PS headers transmitted by DB2.

location
> Either the relational database name (location-name) or the logical unit name (<luname>) of the coordinator.

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 208 was not active or it was active with a destination of the resident trace table only. IFCID 208 is a member of statistics class 4, which has a default destination of SMF.

If the protocol error was detected before the commit decision, the unit of work at the DB2 system might be indoubt. DB2 does not automatically resolve such an indoubt thread.

If the protocol error was detected after the commit decision, the unit of work either completed commit or abort processing.

System action: Alert A005 is generated. A trace record whose IFCID=208 is written.

System programmer response: This is an APARable error, probably at the coordinator, but possibly at DB2. The DB2 documentation required is contained in the statistics class 4 trace record. Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'.

The database administrator might need to manually resolve the indoubt thread.

Database Administrator Action: Determine if the 'luwid' is indoubt by using the DISPLAY THREAD command for type indoubt threads and including the LUWID('token') parameter in the command. If the thread is indoubt, it must be manually resolved using the RECOVER INDOUBT command. The commit or abort decision made at the coordinator must be determined so that the same decision can be made at DB2.

Contact the database administrator at the coordinator location to determine whether the thread with LUWID=luwid committed or aborted.

If the coordinator system is another DB2, then the following steps can be taken at the DB2 coordinator to determine the commit or abort decision.

1. If the coordinator DB2 system is running and a DB2 conditional restart was not performed since the error was detected at this (the participant) location, ask the database administrator at the DB2 coordinator location to issue one of the following DISPLAY THREAD TYPE(INDOUBT) commands:

- A -DISPLAY THREAD TYPE(INDOUBT), using the LOCATION keyword and specifying the location name of the local DB2.
- A -DISPLAY THREAD TYPE(INDOUBT) LUWID(luwid)

If a thread with LUWID=luwid appears in the display, then the display also includes the commit or abort decision. If no thread with LUWID=luwid appears in the display, then the thread was aborted.

2. If the DB2 coordinator is down or if a conditional restart was performed there, ask the database administrator at the DB2 coordinator to run DSN1LOGP, specifying SUMMARY(ONLY) FILTER LUWID(luwid). The output from DSN1LOGP includes the commit or abort decision.

DSNL413I   **PROTOCOL ERROR DETECTED IN SYNCPOINT REQUEST/RESPONSE FROM PARTICIPANT** *location* **LUWID=***luwid=token***. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=208 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 is the coordinator of the logical unit of work identified by 'luwid=token'. A protocol error occurred during the SNA sync point exchange with the participant identified by 'location'. The protocol error fits into one of the following categories:

- An unknown SNA PS header was received.
- An SNA PS header was received in the wrong sequence.
- An invalid LU6.2 verb sequence was used to deliver the SNA PS header.
- A DEALLOCATE TYPE(ABEND_SVC) was received from the participant, indicating that the participant detected a protocol violation in the PS headers transmitted by DB2.

location
> The relational database name (location-name) of the participant.

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 208 was not active or it was active with a destination of the resident trace table only. IFCID 208 is a member of statistics class 4, which has a default destination of SMF.

If the protocol error is detected before the commit decision, the unit of work is backed out.

If the protocol error is detected after the commit decision, the outcome of the unit of work is unchanged. In this case, the remote participant might be indoubt. DB2 does not automatically resolve the indoubt thread with the partner 'location' after a protocol error is detected.

System action: Alert A005 is generated. A trace record whose IFCID=208 is written.

System programmer response: This is an APARable error, probably at the participant, but possibly at the DB2 coordinator. The DB2 documentation required is contained in the statistics class 4 trace record. Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'.

Call the database administrator. The participant might need to manually resolve the indoubt thread.

Database Administrator Action: The DB2 coordinator does not know the status of the thread at the participant. The DB2 coordinator assumes that it is indoubt and maintains knowledge of the indoubt participant until it is purged with the RESET INDOUBT command. If the thread is indoubt at the participant, it must be resolved using manual (heuristic) means. The status of the thread at the local DB2 (the coordinator) can be determined with the DISPLAY THREAD command for type indoubt threads by including the LUWID('token') parameter in the command.

If the 'participant' location is also a DB2, the RECOVER INDOUBT command can be used at the participant to resolve the indoubt thread.

DSNL414E   **PROTOCOL ERROR DURING INDOUBT THREAD RESOLUTION WITH PARTICIPANT** *location* **LUWID=***luwid=token* **INFORMATION RECORDED IN TRACE RECORD WITH IFCID=206 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 has coordinator responsibility for the logical unit of work identified by 'luwid=token'. The DB2 thread associated with the logical unit of work is either committed or aborted. A protocol error occurred during

indoubt resolution with the participant identified by 'location'.

The thread at the participant remains indoubt. DB2 rejects further attempts to automatically resolve the indoubt thread.

location
> The relational database name (location-name) of the participant.

luwid=token
> The global logical unit of work id and the local token that identify the thread.

seq-no The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, IFCID 206 was not active or it was active with a destination of the resident trace table only. IFCID 206 is a member of statistics class 4, which has a default destination of SMF.

System action: Alert A003 is generated. A trace record whose IFCID=206 is written.

System programmer response: This is an APARable error, probably at the participant, but possibly at the DB2 coordinator. The DB2 documentation required is contained in the statistics class 4 trace record. Determine the destination of statistics class 4. Locate and print the trace record with SEQUENCE NUMBER='seq-no'.

Call the database administrator. The participant might need to manually resolve the indoubt thread.

Database Administrator Action: The DB2 coordinator maintains knowledge of the indoubt participant until it is purged with the RESET INDOUBT command. If the thread is indoubt at the participant, it must be manually (heuristically) resolved. The status of the thread at the local DB2 (the coordinator) can be determined with the DISPLAY THREAD command for type indoubt threads by including the LUWID('token') parameter in the command.

If the participant is another DB2, the RECOVER INDOUBT command can be used to resolve the indoubt thread.

DSNL415I   **AUTOMATIC RESOLUTION OF** *luwid=token* **WITH COORDINATOR** *location* **RESULTED IN ABORT**

Explanation: The identified indoubt logical unit of work was automatically resolved by communication with the coordinator location. After automatic resolution, this message is issued if message DSNL405 was previously issued to report the thread as indoubt. This message is not issued if message DSNL405 occurs in one restart of DDF or DB2, and automatic resolution occurs in a different restart of DDF or DB2.

luwid=token
> The global logical unit of work id and the local token that identify the thread

location
> This field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

System action: Processing continues normally.

DSNL416I   **COLD START CONNECTION REJECTED BY PARTNER** *location*

Explanation: DB2 attempted to make a cold-start connection with a partner location. The partner rejected this attempted connection. DB2 never rejects cold-start connections by partners.

location
> Either the relational database name (location-name) or the logical unit name (<luname>) of the partner.

System action: The connection was not made.

System programmer response: Call the database administrator.

Database Administrator Action: DB2 is not able to connect the partner 'location' until the partner 'location' allows a cold-start connection with DB2. Contact the database administrator at the partner 'location'.

Contact your IBM Support Center for further assistance.

DSNL417E   *csect-name* **COLD START DETECTED AT COORDINATOR LOCATION** *location* **FOR INDOUBT THREAD WITH** *luwid=token*. **MANUAL RESOLUTION IS REQUIRED. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=204 and IFC SEQUENCE NUMBER=***seq-no*.

Explanation: DB2 has resync responsibility for the indoubt thread identified. The partner location informed DB2 that it had performed a cold start operation and lost all knowledge of this indoubt thread. This indoubt thread must be manually resolved with the RECOVER INDOUBT command, followed by the RESET INDOUBT command.

location
> This field contains one of the following:
> - The requester's relational database name (location name)
> - The requester's VTAM LU name (<luname>)
> - The requester's IP address

luwid=token
> The logical unit of work id and the local token that identify the thread.

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 204 was not active or it was active with a destination of the resident trace table only. IFCID 204 is a member of statistics class 4, which has a default destination of SMF and contains information about a partner's cold start.

System action: Alert A001 is generated. A trace record whose IFCID=204 is written.

System programmer response: Call the database administrator.

Database Administrator Action: DB2 is a participant with one or more indoubt threads where the coordinator is 'location'. The DBMS at 'location' performed a cold start. The DB2 participant assumes that the coordinator recovery log was lost or damaged and indoubt thread resolution cannot be achieved. There may be inconsistent data at the coordinator. A list of indoubt threads involving the coordinator 'location' can be displayed using the DISPLAY THREAD command for indoubt type threads by including the LOCATION('location') parameter in the command.

The heuristic decision (that is, to commit or abort the logical unit of work) should be coordinated with any other participants. The existence of other participants might not be easy to determine. The information might be available in the coordinators recovery log even though the coordinator performed a cold start. If the coordinator is a DB2, then it might be possible to execute the DSN1LOGP utility to produce a summary report. If this is possible, the following information might be produced:

- The LUWID of the thread.
- The NID used by IMS or CICS to identify the thread.
- The name of all participants.
- The name of an upstream coordinator if the coordinator is also a participant. This could happen when an application requester accesses a DB2 AS using application-directed access, which in turn accesses another DB2 server with system-directed access. (The local DB2 might be the third system in this scenario.)

At this DB2 participant, all indoubt threads coordinated by the cold starting location must be resolved with the RECOVER INDOUBT command. Locked data remains unavailable until resolution. Threads that were already resolved with this command are discarded. Threads not yet resolved are discarded after resolution with the command.

The commit or abort decision provided using the RECOVER INDOUBT command for a logical unit of work are propagated to all downstream participants, if any.

DSNL418E   *csect-name* **COLD START DETECTED AT PARTICIPANT LOCATION** *location* **FOR INDOUBT THREAD WITH** *luwid=token*. **POSSIBLE DAMAGE. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=204 and IFC SEQUENCE NUMBER=***seq-no*.

Explanation: DB2 has coordinator responsibility for the indoubt thread identified. The partner location informed DB2 that it had performed a cold start operation and lost all knowledge of this indoubt thread. There might be damage at the participant.

location
> This field contains one of the following:

- The requester's relational database name (location name)
- The requester's VTAM LU name (<luname>)
- The requester's IP address

luwid=token
　　The global logical unit of work id and the local token that identify the thread.

seq-no　The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 204 was not active or it was active with a destination of the resident trace table only. IFCID 204 is a member of statistics class 4, which has a default destination of SMF.

System action:　Alert A001 is generated. A trace record whose IFCID=204 is written.

System programmer response:　Call the database administrator.

Database Administrator Action: DB2 is the coordinator with indoubt thread resolution responsibility for one or more indoubt logical units of work at 'location'. The DBMS at 'location' performed a cold start. DB2 assumes that the participant recovery log was lost or damaged and indoubt thread resolution cannot be achieved. There is probably inconsistent data at the participant. Minimally, the participant might not completely reflect the final outcome of the threads that were indoubt at the time the failure occurred. A list of indoubt threads involving the partner 'location' can be displayed using the DISPLAY THREAD command for type indoubt threads by including the LOCATION('location) parameter in the command.

Information about the indoubt participant continues to appear in the DISPLAY THREAD TYPE(INDOUBT) report until it is purged with the RESET INDOUBT command.

DSNL419I　　PARTNER *location* HAS RESOLVED WARM START STATUS AND CONNECTION IS ACCEPTED

Explanation:　A successful warm start connection was made with a partner. A previous attempt to make a warm start connection with the partner was rejected with a DSNL410I message because the partner specified an incorrect log name.

location
　　Either the relational database name (location-name) or the logical unit name (<luname>) of the partner

System action:　The connection was successful. Normal processing continues.

DSNL420I　　COLD START CONNECTION BY PARTICIPANT *location* ACCEPTED. POSSIBLE DAMAGE. INFORMATION RECORDED IN TRACE RECORD WITH IFCID=204 AND IFCID SEQUENCE NUMBER=*seq-no*

Explanation:　DB2 has coordinator responsibility for indoubt threads at a participant and just connected with the participant, which lost all knowledge of indoubt threads because of a previous cold start. There might be damage at the participant.

location
　　This field contains one of the following:
- The requester's relational database name (location name)
- The requester's VTAM LU name (<luname>)
- The requester's IP address

seq-no　The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 204 was not active or it was active with a destination of the resident trace table only. IFCID 204 is a member of statistics class 4, which has a default destination of SMF.

System action:　The connection with the partner is accepted. Alert A001 is generated. A trace record whose IFCID=204 is written.

System programmer response:　Call the database administrator.

Database Administrator Action: DB2 is the coordinator with indoubt thread resolution responsibility for one or more indoubt logical units of work at 'location'. The DBMS at 'location' performed a cold start. DB2 assumes that the participant recovery log was lost or damaged and indoubt thread resolution cannot be achieved. There is probably

inconsistent data at the participant. Minimally, the participant might not completely reflect the final outcome of the threads that were indoubt at the time the failure occurred. A list of indoubt threads involving the partner 'location' can be displayed using the DISPLAY THREAD command for type indoubt threads by including the LOCATION('location) parameter in the command.

Information about the indoubt participant continues to appear in the DISPLAY THREAD TYPE(INDOUBT) report until it is purged with the RESET INDOUBT command.

DSNL421I   SNA XLN PROTOCOL VIOLATION BY PARTNER *location*. **INFORMATION RECORDED IN TRACE RECORD WITH IFCID=0236 AND IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 detected a protocol violation in the SNA Exchange Log Names (XLN) exchange with the partner at the specified location.

location
> Either the relational database name (location-name) or the logical unit name (<luname>) of the partner that sent the invalid XLN message.

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in hexadecimal format. If the sequence number is zero, then IFCID 0236 was not active or it was active with a destination of the resident trace table only. IFCID 0236 is a member of statistics class 4, which has a default destination of SMF.

System action: The attempt to connect with the remote site fails. DB2 writes a trace record with IFCID 0236.

Operator response: Call the system programmer.

System programmer response: Contact the system programmer for the remote site. The invalid XLN message is recorded in the IFCID 0236 trace record. The system logic error that causes the invalid XLN message must be corrected at the remote site.

DSNL422I   WARM START CONNECTION BY PARTNER *location* **REJECTED BECAUSE THE PARTNER REMEMBERS OUR LOG NAME INCORRECTLY. OUR LOG NAME IS** *name1* **AND THE PARTNER REMEMBERS IT AS** *name2*

Explanation: An attempt to make a warm start connection with a partner was rejected because the partner specified our log name as name2. Our log name is name1, which is the LUNAME of the local DB2. This error might be a recoverable error if the partner can restart with our log name as name1. If this is not possible, then the partner must perform a cold start connection with DB2.

location
> The 'location' with which the connection failed: either the relational database name (location-name) or the logical unit name (<luname>) of the partner.

name1   Our log name, as remembered by us.

name2   Our log name, as remembered by our partner and sent to us in the attempted warm start connection.

System action: The connection with the partner is rejected. DB2 retains indoubt knowledge.

System programmer response: Call the database administrator.

Database Administrator Action: Contact the database administrator at the partner 'location' and determine if it is possible for the partner to perform a warm start with our log name specified as the LUNAME of this DB2. If this is possible, the next attempt to connect will succeed.

If this cannot be done, then there are two other possible solutions.
- Have the partner 'location' perform a cold start connection with this DB2 subsystem.
- Use the RESET INDOUBT command with the FORCE option in conjunction with the LOCATION parameter to cause the next connection with the partner to be a cold start connection.

DSNL423I   **AUTOMATIC RESOLUTION OF** *luwid=token* **WITH PARTICIPANT** *location* **RESULTED IN**
             COMMIT

Explanation: The identified indoubt logical unit of work was automatically resolved through communication with
the participant location. After automatic resolution, this message is issued if message DSNL406 was previously issued
reporting the thread as indoubt. This message is not issued if message DSNL406 occurs in one restart of DDF or DB2,
and automatic resolution occurs in a different restart of DDF or DB2.

luwid=token
             The global logical unit of work id and the local token that identify the thread.

location
             This field contains one of the following:
             - The requester's relational database name (location name)
             - The requester's VTAM LU name (<luname>)
             - The requester's IP address

System action: Processing continues normally.

DSNL424I   **AUTOMATIC RESOLUTION OF** *luwid=token* **WITH PARTICIPANT** *location* **RESULTED IN ABORT**

Explanation: The identified indoubt logical unit of work was automatically resolved by communication with the
participant location. After automatic resolution, this message is issued if message DSNL425 was previously issued
reporting the thread as being indoubt. This message is not issued if message DSNL425 occurs in one restart of DDF
or DB2, and automatic resolution occurs in a different restart of DDF or DB2.

luwid=token
             The global logical unit of work id and the local token that identify the thread

location
             This field contains one of the following:
             - The requester's relational database name (location name)
             - The requester's VTAM LU name (<luname>)
             - The requester's IP address

System action: Processing continues normally.

DSNL425I   **THREAD** *luwid=token* **AT PARTICIPANT** *location* **MAY BE INDOUBT BECAUSE OF A**
             COMMUNICATION FAILURE. DB2 ABORTED THE THREAD. INFORMATION RECORDED IN
             TRACE RECORD WITH IFCID=209 AND IFCID SEQUENCE NUMBER=*seq-no*

Explanation: During execution of the two phase commit protocol with the participant identified by 'location',
communications failed. Phase 1 of the protocol completed with the participant and it is prepared for either commit or
abort. The decision to abort the thread was made, but cannot be communicated to the participant at this time. The
participant is indoubt.

DB2 becomes responsible for indoubt resolution with the participant. This responsibility appears in a DISPLAY
THREAD TYPE (INDOUBT) report. Periodic attempts are made to reestablish communication with the participant for
automatic indoubt resolution. If successful, message DSNL424 is written.

luwid=token
             The global logical unit of work id and the local token that identify the thread.

location
             This field contains one of the following:
             - The requester's relational database name (location name)
             - The requester's VTAM LU name (<luname>)
             - The requester's IP address

seq-no   The IFCID sequence number associated with the trace record. The sequence number is printed in
             hexadecimal format. If the sequence number is zero, IFCID 209 was not active or it was active with a
             destination of the resident trace table only. IFCID 209 is a member of statistics class 4, which has a default
             destination of SMF.

System action: Alert A006 is generated. A trace record whose IFCID=209 is written.

Operator response: If message DSNL424 for the LUWID specified in message DSNL406 does not appear in a reasonable amount of time, call the system programmer.

System programmer response: Determine the cause of the communication failure and have the problem fixed. DB2 periodically attempts to reestablish communication. If the problem cannot be fixed in a reasonable amount of time, call the database administrator. The participant might need to resolve the indoubt state manually in order to release locked resources.

Database Administrator Action: If the unit of work with LUWID=luwid is indoubt at 'location', inform the database administrator there that the logical unit of work with LUWID=luwid was aborted to assist in manual resolution of the indoubt thread at 'location'.

DSNL429I   INCONSISTENT DDF LOG DATA, THE FIRST CONNECTION WITH EACH REMOTE LOCATION WILL BE A COLD START CONNECTION

Explanation: DB2 detected an inconsistency in the DB2 log during restart operations. Since the log is inconsistent and cannot be trusted, DB2 performs a cold start connection with each partner LU the next time a connection with that LU is attempted. If DB2 has indoubt resolution responsibility either as a coordinator or as a participant, then this information is lost as indicated by messages DSNL438 and/or DSNL439, which might follow this message. This means that:

- All information that DB2 maintained about having coordinator responsibility for remote participants was purged. These remote participants need manual resolution for any indoubt threads for which DB2 was the coordinator. A DSNL439I message is displayed for each indoubt remote participant.

- All information that DB2 maintained about having participant responsibility for remote coordinators was purged. A DSNL438I message is displayed for each database access thread that has a remote coordinator. If a database access thread remains indoubt at the end of restart, then the thread must be manually resolved. A -DISPLAY THREAD (*) TYPE(INDOUBT) LOCATION (*) command provides a report that lists the indoubt threads that need manual resolution.

System action: Processing continues normally.

System programmer response: Call the database administrator.

Database Administrator Action: See the DSNL438I and DSNL439I messages to determine what action to take.

DSNL432I   DDF IS COORDINATOR FOR *nnnn* THREADS WITH *mmmm* INDOUBT REMOTE PARTNERS

Explanation: This message is issued when DDF is started or stopped, if DDF determines that this DB2 has coordinator responsibility for one or more indoubt threads with one or more partners. This is a summary status message.

nnnn   The number of indoubt threads for which DB2 is the coordinator

mmmm
        The number of remote partners where the threads are indoubt

System action: Processing continues normally.

Operator response: If the message was issued as a result of DDF starting, no action is necessary. If the message was issued as a result of DDF stopping, contact the database administrator.

Database Administrator Action: If the message was issued as a result of stopping DDF, then DB2 maintained all information necessary to perform automatic resynchronization with partner locations and resolve all indoubt units of work. Issue the START DDF command to initiate this process.

If the message was issued as a result of starting DDF, then DDF started automatic resynchronization.

DSNL433I   DDF IS PARTICIPANT FOR *nnnn* INDOUBT THREADS WITH *mmmm* REMOTE PARTNERS

Explanation: This message is issued when DDF is started or stopped, if DDF determines that this DB2 has participant responsibility for one or more indoubt threads with one or more partners. This is a summary status message.

nnnn   The number of indoubt threads for which DB2 is the participant

mmmm
        The number of remote partners that have coordinator responsibility

System action: Processing continues normally.

Operator response: If the message was issued as a result of DDF starting, no action is necessary. If the message was issued as a result of DDF stopping, contact the database administrator.

Database Administrator Action: If the message was issued as a result of stopping DDF, then DB2 maintained all information necessary to perform automatic resynchronization with partner locations and resolve all indoubt units of work. Issue the START DDF command to initiate this process.

If you do not want to perform automatic resynchronization, you can still resolve the indoubt units of work for which DB2 is a participant and which hold database resources locked. Use the DISPLAY THREAD command for type indoubt threads to produce a display that lists these threads. Then use the RECOVER INDOUBT command to either commit or abort these units of work.

To avoid data inconsistency, coordinate this manual resolution with the database administrators at the locations that have coordinator responsibility for each indoubt unit of work. If any of the indoubt units of work at DB2 also have coordinator responsibility for participant locations, then you should notify the database administrators at the participant locations of the commit or abort decision so they can make the same decision.

DSNL434E   DDF NOT STARTED - INCORRECT VTAM RELEASE LEVEL

Explanation: The local VTAM release level is older than the minimum release level required by DB2. When DB2 issues the VTAM OPEN macro, the VTAM release level is returned to DB2. DB2 examines the VTAM release level to determine whether the local VTAM system provides the function required by DB2.

System action: The attempt to START DDF fails.

Operator response: Call the system programmer.

System programmer response: Install a release of VTAM that is compatible with the DB2 system. The DB2 Program Directory identifies the required VTAM release level.

DSNL438I   RESYNC STATUS PURGED FOR COORDINATOR *location*. **LUWID=***luwid*, **URID=***urid*, **STATUS=***s*
        INFORMATION RECORDED IN TRACE RECORD WITH IFCID=235 and IFCID SEQUENCE
        NUMBER=*seq-no*

Explanation: The information related to resynchronization with the remote coordinator location for the logical unit of work identified by 'luwid', which is displayed in this message, is being purged by DB2. Automatic indoubt resolution responsibility for this distributed logical unit of work can no longer be performed.

This situation can arise in one of two ways;
- A DB2 conditional restart (other than a DB2 cold start; that is, a DB2 conditional restart with STARTRBA=ENDRBA) is being performed.
- An inconsistency in the DB2 log was detected during DB2 restart processing (indicated by a DSNL429I message).

location
        This field contains one of the following:
        - The requester's relational database name (location name)
        - The requester's VTAM LU name (<luname>)
        - The requester's IP address

luwid   The global logical unit of work id that identifies the logical unit of work.

urid   The unit of recovery identifier. This is the RBA on the DB2 recovery log of the log record that marks the beginning of the logical unit of work. This is the earliest RBA required to process the logical unit of work during restart.

s   Restart status of the logical unit of work. When DB2 came down, the logical unit of work was in one of these states:

        A   INABORT: The logical unit of work was in the "must-complete" phase of abort, but abort has not yet completed.

C	INCOMMIT: The logical unit of work was in the "must-complete" phase of commit, but commit has not yet completed.
D	INDOUBT: The logical unit of work completed the first phase of commit, but DB2 has not yet received the commit or abort decision from the coordinator.
F	INFLIGHT: The logical unit of work has not yet completed the first phase of commit and should be backed out.
U	UNKNOWN: The status of the logical unit of work is not known to DB2 because of conditional restart.

seq-no The IFCID sequence number associated with the trace record. The trace record is written to statistics class 4, which is started to SMF as a default during DB2 installation. Statistics class 4 must be started to a permanent destination (SMF or GTF) for the trace record to be accessible. The sequence number is printed in hexadecimal.

System action: Processing continues normally.

System programmer response: Call the database administrator.

Database Administrator Action: The resynchronization information that was purged by DB2 is recorded in this message. To determine the proper commit or abort decision made by the coordinator at 'location', use the status code as follows:

A	The correct decision is to abort the logical unit of work.
C	The correct decision is to commit the logical unit of work.
D	Contact the database administrator at 'location' to determine the correct decision.
F	The correct decision is to abort the logical unit of work.
U	Contact the database administrator at 'location' to determine the correct decision.

DSNL439I **RESYNC STATUS PURGED FOR PARTICIPANT** *location*. **LUWID=***luwid*, **URID=***urid*, **STATUS=***s* **INFORMATION RECORDED IN TRACE RECORD WITH IFCID=235 and IFCID SEQUENCE NUMBER=***seq-no*

Explanation: DB2 is purging the information related to resynchronization with the remote participant location for the logical unit of work identified by 'luwid', which is displayed in this message. Automatic indoubt resolution responsibility for this distributed logical unit of work can no longer be performed.

This situation can arise in one of two ways:

- A DB2 conditional restart (other than a DB2 cold start; that is, a DB2 conditional restart with STARTRBA=ENDRBA) is being performed.
- An inconsistency in the DB2 log was detected during DB2 restart processing (indicated by a DSNL429I message).

location
This field contains one of the following:
- The requester's relational database name (location name)
- The requester's VTAM LU name (<luname>)
- The requester's IP address

luwid The global logical unit of work id that identifies the logical unit of work.

urid The unit of recovery identifier. This is the RBA on the DB2 recovery log of the log record that marks the beginning of the logical unit of work. This is the earliest RBA required to process the logical unit of work during restart.

s Restart status of the logical unit of work. When DB2 came down, the logical unit of work was in one of these states:

A	INABORT: The logical unit of work was in the "must-complete" phase of abort, but abort has not yet completed.
C	INCOMMIT: The logical unit of work was in the "must-complete" phase of commit, but commit has not yet completed.

D   INDOUBT: The logical unit of work completed the first phase of commit, but DB2 has not yet received the commit or abort decision from the coordinator.

F   INFLIGHT: The logical unit of work has not yet completed the first phase of commit and should be backed out.

U   UNKNOWN: The status of the logical unit of work is not known to DB2 because of conditional restart.

seq-no   The IFCID sequence number associated with the trace record. The trace record is written to statistics class 4, which is started to SMF as a default during DB2 installation. Statistics class 4 must be started to a permanent destination (SMF or GTF) for the trace record to be accessible. The sequence number is printed in hexadecimal.

System action: Processing continues normally.

System programmer response: Call the database administrator.

Database Administrator Action: The resynchronization information that was purged by DB2 is recorded in this message. When communicating the proper commit or abort decision to the database administrator at 'location', use the status code as follows:

A   Abort the logical unit of work.

C   Commit the logical unit of work.

D   See the DSNL438I message produced for the same 'luwid' and 'urid' to determine the coordinator 'location'. Contact the database administrator at 'location' to determine the correct decision.

F   Abort the logical unit of work.

U   If there is a DSNL438I message for the same 'luwid' and 'urid', proceed as described above under 'D'. Otherwise, there might be information available on the DB2 log that a DSN1LOGP summary report can provide. If this is not successful, contact the IBM Support Center.

DSNL440I   *csect-name* **RESET INDOUBT COMMAND SPECIFICATION IN ERROR**

Explanation: A RESET INDOUBT command was issued, but the specification of keywords or keyword parameter values was in error.

System action: No indoubt information was purged. System processing continues normally.

User response: One of the following errors was made in the RESET INDOUBT command specification:
- RESET INDOUBT was issued without any other keywords specified. At least one of the following keywords must also be specified: LUNAME, IPADDR, LOCATION, or LUWID.
- RESET INDOUBT was issued with both the LUWID and LOCATION keywords specified, but a list of LOCATION keyword parameter values was specified. When both the LUWID and LOCATION keywords are specified only one LOCATION keyword parameter value is allowed.
- RESET INDOUBT was issued with at least one LOCATION keyword parameter value that began with a '<', but did not end with a '>', or the length of the bracketed value was not 1 to 8 characters. When the LOCATION keyword parameter value is specified in the <luname> format, the luname value must be bracketed with '<' and '>' and be 1 to 8 characters long.
- RESET INDOUBT was issued with an invalid IPADDR keyword parameter. Verify that the specified IPADDRs are in the proper form, that there is a double period delimiter (..) between the IPADDR and port, and that the port number is valid.

If needed, issue the DISPLAY THREAD TYPE(INDOUBT) command to display information (including LOCATION, LUWID, and LUNAME values) about all indoubt threads. Correct the RESET INDOUBT command specification, and reissue the command.

DSNL441I   **KEYWORD 'LUWID' PARAMETER VALUE** *parmvalue* **FORMAT IN ERROR**

Explanation: A RESET INDOUBT command with the LUWID parameter was issued, but the specified parameter value (parmvalue) has an invalid format for an LUWID. There are two correct formats for the LUWID parameter. The first correct format is called a token, and is a 1 to 5 digit decimal number. The second correct format is 'netid.luname.luwunique.comseq', where:

netid   A 1 to 8 character network ID

luname
> A 1 to 8 character luname

luwunique
> A 12 character hexadecimal number

comseq
> A 4 character hexadecimal number

All fields in the second correct format must be separated by a period (.) delimiter.

System action: Indoubt information about the specified logical unit of work is not purged from the system.

User response: To determine the correct LUWID value for the unit of work indoubt information that you want to purge, issue the DISPLAY THREAD TYPE(INDOUBT) command to display all units of work for which indoubt information exists. Then reissue the RESET INDOUBT command, specifying the correct LUWID parameter value.

DSNL442I   **QUALIFYING INDOUBT INFORMATION FOR LOCATION** *location* **NOT FOUND**

Explanation: A -RESET INDOUBT LOCATION('location') command was issued, but DB2 has no indoubt information about the specified location.

location
> The value specified after the LOCATION keyword in the RESET INDOUBT command.

System action: No indoubt information was purged. System processing continues normally.

User response: Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to list the locations for which DB2 has indoubt information. After determining the correct location, reissue the command.

DSNL443I   **QUALIFYING INDOUBT INFORMATION FOR LUWID=***value* **NOT FOUND**

Explanation: A -RESET INDOUBT LUWID('value') command was issued where 'value' is either an luwid or the local token that represents an luwid. Either DB2 has no indoubt information about the specified luwid or token, or DB2 has indoubt information that cannot be purged using the -RESET INDOUBT command with the LUWID option.

value   The global logical unit of work id or the local token that was specified for the LUWID keyword in the RESET INDOUBT command.

System action: No indoubt information was purged. System processing continues normally.

User response: Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to list the indoubt information known to DB2. If the -DISPLAY THREAD output does not include information about the specified luwid, then that is the reason the message was issued. If the -DISPLAY THREAD output does include information about the specified luwid, this information cannot be purged using the -RESET INDOUBT command with the LUWID option. Refer to the description of each message produced by the -DISPLAY THREAD output that relates to the specified luwid in order to determine how to proceed. After determining the correct luwid or token, reissue the command.

DSNL444I   **QUALIFYING INDOUBT INFORMATION FOR LOCATION** *location* **HAS BEEN PURGED**

Explanation: A -RESET INDOUBT LOCATION('location') command issued to purge indoubt information about the named location completed successfully and the indoubt information was purged.

location
> Either the relational database name (location-name) or the logical unit name (<luname>) that was specified after the LOCATION keyword on the RESET INDOUBT command.

System action: The specified indoubt information was purged. System processing continues normally.

DSNL445I   QUALIFYING INDOUBT INFORMATION FOR LUWID *value* HAS BEEN PURGED

Explanation: A -RESET INDOUBT LUWID('value') command, which was issued to purge indoubt information about the specified luwid, completed successfully. The indoubt information was purged.

value   The value specified for the LUWID keyword in the -RESET INDOUBT LUWID(value) command that is being processed.

System action: System processing continues normally.

DSNL446I   COLD START WILL BE PERFORMED WITH *nettype netvalue*

Explanation: A -RESET INDOUBT LUNAME('luname') FORCE or a -RESET INDOUBT IPADDR('ipaddr') FORCE command issued to force a cold start completed successfully. The next connection with 'luname' or 'ipaddr' is a cold start.

nettype
>
> Identifies the type of network used to access the remote site. There are two possible values:
>
> **LU --**   indicates that VTAM is used to connect to the remote site.
>
> **IPADDR --**
> > indicates that TCP/IP is used to connect to the remote site.

netvalue
>
> The network address of the remote location. *netvalue* can be blank.
> - If *nettype* is LU, *netvalue* specifies the VTAM LU name of the remote location.
> - If *nettype* is IPADDR, *netvalue* specifies the IP address of the remote location.

System action: System processing continues normally.

DSNL447I   CANNOT DELETE ENTRY *value* BECAUSE THERE IS AN INDOUBT THREAD TO BE RESOLVED FIRST

Explanation: A -RESET INDOUBT command was issued with LUNAME(*value*), IPADDR(*value*), LOCATION(*value*), or LUWID(*value*). The *value* is an LU name, an IP address, a location, or a LUWID (or its token), that identifies a logical unit of work for which DB2 has participant indoubt resolution responsibility, coordinator indoubt resolution responsibility, or both.

System action: Resynchronization responsibility information was not purged. System processing continues normally.

User response: Use the information in this section to determine the correct action to take. Then issue the proper command to perform that action.

The two most probable causes for the problem are that the -RESET INDOUBT command was issued with the wrong parameters, or that an incorrect 'value' was specified. Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to display the indoubt information known to DB2.

1. If the intent was to purge participant or coordinator indoubt resolution responsibility, then examine the -DISPLAY THREAD output (DSNV401) to determine the location name, or names, for which this responsibility is to be purged. Then use the -RESET INDOUBT LUNAME(*luname*) command with the FORCE option or the -RESET INDOUBT IPADDR(*ip-address*) command with the FORCE option to purge this responsibility.

   To determine the luname or IP address and resync port for an upstream coordinator that can be used with the FORCE option, look at the coordinator information after the DSNV406I message in the -DISPLAY THREAD output. There are two tokens associated with the coordinator information that are delimited by a dash (-) for SNA connections, and by a double period (..) for TCP/IP connections.

   - The first token is the partner location name or "<luname>" for an SNA connection, or the IP address for the TCP/IP connection.
   - The second token is the partner LU name for an SNA connection, or the resync port number for a TCP/IP connection.

   To determine the luname or IP address and resync port for downstream participants that can be used with the FORCE option, look at the location names after the V444 message and in the V450 message in the DISPLAY THREAD output (DSNV401). There are two or three tokens associated with the participant location.

- The first token is the partner location name.
- The second token is the partner LU name for an SNA connection, or the IP address for a TCP/IP connection. The second token is delimited from the first token by a dash (-).
- The third token is the partner resync port number. The third token appears only if the second token is an IP address, and is delimited from the second token by a double period (..).

Remember that the FORCE option causes DB2 to purge all knowledge of indoubt resolution responsibility involving the LUNAME or IPADDR values specified in the command.

Use caution when using the FORCE parameter to purge participant indoubt information when no errors are precluding automatic resolution. Normally, after the use of the RECOVER INDOUBT command, automatic resolution with the coordinator determines if heuristic damage occurred. This detection is lost if RESET INDOUBT is used before automatic resolution with the coordinator can be achieved.

Use caution when using the FORCE parameter to purge coordinator indoubt information when no errors are precluding automatic resolution. Once the information is purged, any participant that is indoubt is forced to use a heuristic decision process to resolve the indoubt logical unit of work.

2. Examine the -DISPLAY THREAD output. If the value specified in the command (and displayed in the DSNL447I message) is incorrect, determine the correct value and reissue the command using the correct value.

Related reference:

▶ -RESET INDOUBT (DB2) (DB2 Commands)

▶ -RECOVER INDOUBT (DB2) (DB2 Commands)

▶ -DISPLAY THREAD (DB2) (DB2 Commands)

Related information:

"DSNV401I" on page 942

"DSNL447I" on page 367

DSNL448I   **UNABLE TO FORCE COLD START WITH** *nettype netvalue*, **SESSION STILL EXISTS**

Explanation: A -RESET INDOUBT LUNAME('luname') FORCE or a -RESET INDOUBT IPADDR('ipaddr') FORCE command was issued to force a cold start, but network connections still exist between DB2 and the partner 'luname' or 'ipaddr'. The information related to this partner was not purged, so a cold start is not performed.

nettype
> Identifies the type of network used to access the remote site. There are two possible values:

> **LU --**   indicates that VTAM is used to connect to the remote site.

> **IPADDR --**
>> indicates that TCP/IP is used to connect to the remote site.

netvalue
> The network address of the remote location. *netvalue* can be blank.
> - If *nettype* is LU, *netvalue* specifies the VTAM LU name of the remote location.
> - If *nettype* is IPADDR, *netvalue* specifies the IP address of the remote location.

System action: System processing continues normally.

User response: For SNA, you can use the VTAM command DISPLAY NET to determine the existence of sessions between two logical units. Terminate the sessions by using the VTAM command VARY NET,TERM to allow the RESET INDOUBT to occur. For TCP/IP, terminate all connections to the IP address.Determine the correct luname or ipaddr and reissue the command.

DSNL449I   **DB2 HAS NO INFORMATION RELATED TO LUNAME** *luname*

Explanation: A -RESET INDOUBT LUNAME('luname') FORCE command was issued but DB2 has no knowledge of the specified 'luname'.

luname
The logical unit name, as defined to VTAM, of the coordinator or participant

System action: No information was purged. System processing continues normally.

User response: Determine the correct luname as follows:

- Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to determine locations for which DB2 has indoubt responsibility.
- Query the SYSIBM.LOCATIONS table to correlate a location name with an luname. The LINKNAME column contains the luname associated with the location name in the LOCATION column.

Then reissue the command, specifying the correct luname.

DSNL450I DDF THREAD *luwid=token* IS BETWEEN PHASE 1 AND PHASE 2 OF COMMIT. THREAD IS NOW AN INDOUBT THREAD AS A RESULT OF CANCEL REQUEST.

Explanation: The identified DDF thread was in a prepared state waiting for the commit or backout decision of the commit coordinator. The thread cannot be canceled in this state. However, the thread was removed from the active state and placed into the indoubt state. The conversation with the coordinator was deallocated. The thread is now indoubt and appears in the display thread report for indoubt threads. The resources held by the thread continue to be held. Automatic resolution of the indoubt state is attempted periodically. Resources held by the thread are released after indoubt resolution occurs.

luwid=token
The global logical unit of work id and the local token that identify the thread

System action: Processing continues normally.

Operator response: If the purpose of canceling this thread was to free the resources held by the thread, call the database administrator.

Database Administrator Action: To release the resources held by the indoubt thread before automatic resolution with the coordinator, a manual commit or abort decision can be made with the RECOVER INDOUBT command. First, issue the DISPLAY THREAD command for an indoubt thread report, which includes the subject thread. From this report, determine the name of the coordinator. Contact the coordinator system and find out whether a commit or abort decision was made.

DSNL451I QUALIFYING INDOUBT INFORMATION FOR LUNAME *luname* NOT FOUND

Explanation: A -RESET INDOUBT LUNAME('luname') command was issued, but DB2 has no indoubt information about the specified luname.

luname
The logical unit name, as defined to VTAM, of the coordinator or participant

System action: No indoubt information was purged. System processing continues normally.

User response: Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to list the locations for which DB2 has indoubt information. Entries associated with a particular LUNAME have a location name in this form: '<luname>'. Use 'luname' (without the enclosing brackets) as the input string to RECOVER INDOUBT.

After determining the correct luname, reissue the command.

DSNL452I QUALIFYING INDOUBT INFORMATION FOR LUNAME *luname* HAS BEEN PURGED

Explanation: A -RESET INDOUBT LUNAME('luname') command, which was issued to purge indoubt information about the specified luname completed successfully. The indoubt information was purged.

luname
The luname specified for the LUNAME keyword in the -RESET INDOUBT LUNAME(luname) command being processed.

System action: The specified indoubt information was purged. System processing continues normally.

DSNL453I   *csect* **QUALIFYING INDOUBT INFORMATION FOR IPADDR** *ipaddr* **NOT FOUND**

Explanation: A -RESET INDOUBT IPADDR('ipaddr') command was issued, but DB2 has no indoubt information about the specified IP address.

csect   The CSECT name of the DB2 module that issued the message

ipaddr   The IP address of the coordinator or participant.

System action: No indoubt information was purged. System processing continues normally.

User response: Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to list the locations for which DB2 has indoubt information. Entries associated with TCP/IP have a string with this format in the LOCATION field: "nnn.nnn.nnn.nnn".

After determining the correct *ipaddr*, reissue the command.

DSNL454I   *csect* **QUALIFYING INDOUBT INFORMATION FOR IP ADDRESS** *ipaddr* **HAS BEEN PURGED**

Explanation: A -RESET INDOUBT IPADDR('ipaddr') command, which was issued to purge indoubt information about the specified IP address completed successfully. The indoubt information was purged.

csect   The CSECT name of the DB2 module that issued the message

ipaddr   The IP address specified for the IPADDR keyword in the -RESET INDOUBT IPADDR(ipaddr) command being processed

System action: The specified indoubt information was purged. System processing continues normally.

DSNL455I   *csect* **DB2 HAS NO INFORMATION RELATED TO IP ADDRESS** *ipaddr*

Explanation: A -RESET INDOUBT IPADDR('ipaddr') FORCE command was issued but DB2 has no knowledge of the specified 'ipaddr'.

csect   The CSECT name of the DB2 module that issued the message.

ipaddr   The IP address of the coordinator or participant.

System action: No information was purged. System processing continues normally.

User response: Determine the correct luname as follows:

- Use the -DISPLAY THREAD(*) TYPE(INDOUBT) LOCATION(*) command to determine locations for which DB2 has indoubt responsibility. Entries associated with TCP/IP locations specify "nnn.nnn.nnn.nnn" in the LOCATION field.

Reissue the command; specify the correct ipaddr.

DSNL500I   **CONVERSATION FAILED TO LOCATION=***locname* **RESOURCE=***luname.modename***.** *convid.sessid* **RTNCD=***global vtam primary return code* **FDBK2=***global vtam secondary return code* **RCPRI=***appc-specific primary return code* **RCSEC=***appc-specific secondary return code* **SENSE=***sna sense code*

Explanation: An advanced program-to-program communications (APPC) conversation failed, or could not be allocated, because a VTAM return code indicates a conversation failure. VTAM return codes are defined in *z/OS Communications Server SNA Programmer's LU 6.2 Reference*.

SNA sense codes that start with 00 represent DB2 reason codes.

The variables that are not return codes are described as follows:

Variable
     Description

locname   The location name associated with the remote site.

luname   The VTAM LU name associated with the remote location.

modename
     The logon mode used by DB2 to allocate the conversation.

convid   The conversation identifier given DB2 by VTAM when the conversation was allocated.

sessid   The session identifier given DB2 by VTAM when the conversation was allocated.

System action: Communication on the conversation is discontinued.

Operator response: Notify the system programmer.

System programmer response: Take the actions that are specified for the SNA sense code and DB2 reason code values.

Related concepts:

▱➔ DB2 reason codes (DB2 Codes)

Related reference:

▱➔ SNA sense codes (DB2 Codes)

DSNL501I   **CNOS PROCESSING FAILED FOR LU** *luname* **AND MODE** *modename* **RTNCD=***global vtam return code* **FDBK2=***global vtam secondary return code* **RCPRI=***appc-specific primary return code* **RCSEC=***appc-specific secondary return code* **SENSE=***vtam sense code*

Explanation: A failure occurred while attempting to negotiate limits with the specified location for a LU and mode. VTAM return codes are described in *z/OS Communications Server SNA Programmer's LU 6.2 Reference*. VTAM sense codes that start with 00 represent DB2 reason codes.

The variables are described as follows:

luname   The VTAM LU name associated with the remote location.

modename
         The logon mode used by DB2 to allocate the conversation.

System action: All communication to the remote location fails until the limits can be negotiated with the remote location.

Operator response: Notify the system programmer.

System programmer response: Refer to *z/OS Communications Server SNA Programmer's LU 6.2 Reference* and the DB2 reason code for failure analysis.

Related concepts:

▱➔ DB2 reason codes (DB2 Codes)

DSNL502I   **SYSTEM CONVERSATION FAILED TO LOCATION** *locname* **RESOURCE=***luname.modename***.** *convid.sessid* **RTNCD=***global vtam primary return code* **FDBK2=***global vtam secondary return code* **RCPRI=***appc-specific primary return code* **RCSEC=***appc-specific secondary return code* **SENSE=***sna sense code*

Explanation: A system conversation failed, or could not be allocated, because VTAM returned a nonzero return code.

SNA sense codes that start with 00 represent DB2 reason codes.

locname   The location name associated with the remote site.

luname   The VTAM LU name associated with the remote location.

modename
         The logon mode used by DB2 to allocate the conversation.

convid   The conversation identifier given DB2 by VTAM when the conversation was allocated.

sessid   The session identifier given DB2 by VTAM when the conversation was allocated.

System action: Communication on the conversation is not allowed.

Operator response: Notify the system programmer.

System programmer response: Take the actions that are specified for the SNA sense code and DB2 reason code values.

Related concepts:

⮕ DB2 reason codes (DB2 Codes)

Related reference:

⮕ SNA sense codes (DB2 Codes)

Related information:

⮕ Return Codes (z/OS Communications Server: SNA Programmer's LU 6.2 Guide)

DSNL504I   *csect-name* **RESOURCE UNAVAILABLE REASON=***reason* **TYPE=***type* **LOCATION=***location*
          NAME=*name*

Explanation: The requested function requires a resource that is currently unavailable. The reason code, type, location, and name identifies the resource.

Operator response: Note the reason, type codes, location, and name.

Problem determination: The reason for the resource's unavailability is identified by the reason code and the resource type code.

Related concepts:

⮕ DB2 reason codes (DB2 Codes)

Related information:

Appendix C, "Resource types," on page 1153

DSNL510I   *csect-name* **CONVLIMIT NEGOTIATED FROM** *x* **TO** *y* **WITH LOCATION** *locname* **FOR LU** *luname*
          AND MODE *modename*

Explanation: The maximum conversation limit is changed for a mode with a remote subsystem. The limit might change when a CNOS request is received from or sent to a remote subsystem, and the limit is determined by VTAM.

x   The most recent value of CONVLIMIT before the last CNOS request was processed by advanced
          program-to-program communications (APPC)/VTAM for the specified mode name to the specified location.

y   The current value of CONVLIMIT for the specified mode name to the specified location.

locname
          The location name associated with the remote site that sent the CNOS request to the local DB2 subsystem, or
          received the CNOS request from the local DB2 subsystem.

luname
          The VTAM LU name associated with the remote location that had its CONVLIMIT changed.

modename
          The logon mode for the specified location that had its CONVLIMIT changed.

System action: Conversation requests are queued when the maximum conversation limit is reached for a mode to remote subsystem.

DSNL511I   *csect-name* **TCP/IP CONVERSATION FAILED TO LOCATION** *location-name* **IPADDR=***ip-address*
          PORT=*port* **SOCKET=***socket* **RETURN CODE=***return-code* **REASON CODE=***reason-code*

Explanation: A TCP/IP socket call failed with the error condition identified by the return code and reason code.

csect-name
          The name of the control section that issued the message.

location-name
          The location name associated with the remote site.

ip-address
          The IP address of the remote site.

port   The TCP/IP port used by the remote site.

socket   The TCP/IP socket call that returned the error.

return-code
>   The error condition reported by TCP/IP, as a decimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

reason-code
>   Further detail associated with the TCP/IP return code, as a hexadecimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

The following values together indicate a zero-length receive, which is considered a closed condition:
* SOCKET=RECV
* RETURN CODE=0
* REASON CODE=00000000

System action: Communication on the conversation is discontinued.

Operator response: Notify the system programmer.

System programmer response: Use the information provided in the message to correct the problem.

User response: Notify the system programmer.

Problem determination: Determine the meaning of the return and reason codes issued by z/OS UNIX System Services or z/OS Communications Server.

Related reference:

➡ z/OS UNIX System Services: Messages and Codes

➡ z/OS Communications Server: IP and SNA Codes

DSNL512I   *csect-name* **TCP/IP** *socket* **FAILED WITH RETURN CODE=***return-code* **AND REASON CODE=***reason-code*

Explanation: A TCP/IP socket call used to establish communication between DB2 and TCP/IP failed.

csect-name
>   The name of the control section that issued the message.

socket   The name of the TCP/IP socket call that failed.

return-code
>   The return code reported by TCP/IP, as a decimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

reason-code
>   The reason code reported by TCP/IP, as a hexadecimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

System action: DB2 will periodically retry the socket call to establish TCP/IP communication.

Operator response: Notify the system programmer.

System programmer response: The response is based on the TCP/IP socket call that failed.

socket=**GETHOSTNAME**
socket=**GETNAMEINFO(***ip-address***)**
socket=**GETADDRINFO(***host-name***)**
>   The GetHostName, GetNameInfo, or GetAddrInfo call has failed.
>
>   During DDF initialization, DB2 first determines its host name using a GetHostName socket call. DB2 must then determine its IP address. If an IP address is specified in the BSDS using the DSNJU003 utility, or on the PORT statement in the TCP/IP profile, then DB2 uses this IP address. Otherwise, DB2 executes a GetAddrInfo(*host-name*) socket call, using the *host-name* that was returned on the prior GetHostName socket call, which returns the first IP address listed in the name server or local host table.
>
>   **Important:** The host name or IP address returned could depend on the selected TCP/IP stack, in the event that there are multiple TCP/IP stacks. DB2 requires z/OS UNIX System Services sockets, and it is z/OS

UNIX System Services that determines the TCP/IP stack to use. z/OS UNIX System Services provides a way for users to select a specific TCP/IP stack by using the BPXTCAFF utility.
DB2 then issues a GetNameInfo socket call to determine the domain name associated with the derived IP address.

The z/OS Resolver handles the GetNameInfo socket call because DB2 uses the z/OS UNIX System Services Asynchronous Sockets interface. This resolver uses the resolver configuration information available to DDF. Until this information for the selected TCP/IP is correctly specified for DDF, the z/OS Resolver will be unable to determine the domain name for DDF. You can find the search order for the "resolver configuration files" for a z/OS application, such as DDF, in the IP Configuration manual for the appropriate level of the Communication Server for your z/OS operating system.

socket=**GETHOSTNAME**
> DB2 DDF initialization invoked a TCP/IP GetHostName socket call in order to determine the host name associated to TCP/IP, and hence the host name associated with the DB2 subsystem. However, the call failed and DB2 is not able to determine the host name. DB2 retries the socket call periodically.

socket=**GETNAMEINFO(***ip-address***)**
> DB2 invoked a TCP/IP GetNameInfo socket call to determine the domain name for the specified IP address. However, the call failed and DB2 is not able to determine the domain name. DB2 retries the socket call periodically.

socket=**GETADDRINFO(***host-name***)**
> DB2 invoked a TCP/IP GetAddrInfo socket call to determine the IP address associated with the specified host name. However, the call failed and DB2 is not able to determine the IP address. DB2 retries the socket call periodically.
>
> If you want a BSDS-specified IP address, correct the BSDS. This requires you to stop and restart DB2. Otherwise, verify that the provided host name is correct and take the appropriate action:
>
> - If the host name is incorrect, correct the host name in the appropriate resolver configuration file, refresh TCP/IP, and then stop and start DDF so that the new host name is refreshed for DDF initialization.
> - If the host name is correct, then TCP/IP is unable to resolve the host name to an IP address. You must make changes in order to allow for proper host name to IP address resolution. If the changes have been made to the name server or local host table, refresh the resolver.

socket=**POLL**
> The online ZPARM was changed using the SET SYSPARM command, changing the MAXDBAT from a positive value to zero. In this case, the message is informational, indicating that DB2 terminated its TCP/IP listening capability for SQL related work because DB2 can no longer accept SQL related connections from remote partners when MAXDBAT is set to zero.

socket=**SHUTDOWN_REG (USS_UNAVAILABLE)**
> The z/OS UNIX System Services (USS/OMVS) environment is not available. DDF start processing continues and this message is periodically repeated until the Unix System Services environment becomes available or DDF is stopped. You can use the z/OS "F OMVS, RESTART" command to start the z/OS UNIX System Services environment.

socket=**SETRLIMIT and** *reason code*=**1148033C**
> You should define DB2 as a superuser. For instructions on how to enable DDF as a superuser, see the information about enabling DDF for UNIX System Services.

socket=**TCPKEEPALIVE**
> DB2 issues this socket call as a result of the TCP/IP KeepAlive value specified in the DB2 parameters module (usually DSNZPARM). You may specify this value by the TCP/IP KEEPALIVE value in the DSNTIP5 installation panel. Another way to specify the value is through the TCPKPALV value in the DSN6FAC macro, used to build the DB2 parameters module.
>
> In this situation, processing continues, and DB2 does not retry the socket call. Verify that the TCP/IP stack being used by DB2 contains the required companion support for this function. If not, modify the DB2 parameter module to prevent DB2 from attempting to override the TCP/IP stack value.

socket=**BSDS_ERROR (ADDRESS_ON_PORT_STMT)**
> An IP address is specified in both the BSDS and the PORT statement of the TCP/ IP profile. An address

should be specified in only one place. That is, DB2 detected that IP addresses have been specified in the BSDS (through the IPV4, GRPIPV4, IPV6, and GRPIPV6 keywords of the DSNJU003 Change Log Inventory utility) and also in the TCP/IP profile on the PORT statement for the port value to be used by this subsystem. Only one of the specifications can now have IP addresses. The DB2 TCP/IP environment will not initialize, and the error is periodically repeated until the error is corrected.

One of the IP address specifications must be removed from the BSDS or from the TCP/IP profile. However, the BSDS specification should be retained because the recommendation is to use BSDS specifications to define the IP addresses that are to be associated to DB2, and, because removing the BSDS specifications would require DB2 to be stopped. In this case, to allow the initialization of the DB2 TCP/IP environment to continue. The TCP/IP profile PORT statement can be changed to remove the IP addresses and then the changed specifications can be dynamically brought on-line, which favors the BSDS specifications.

socket=**BSDS_ERROR (IPv6_REQUIRES_DUALMODE_STACK)**

A member or subsystem, or possibly a group, IPv6 address is specified in the BSDS without TCP/IP dual mode stack enabled. DB2 ignores all IPv6 addresses defined in the BSDS and DDF start-processing continues normally.

You should either enable a dual mode stack or delete all of the BSDS-defined IPv6 addresses and use only IPv4 addresses.

socket=**BINDSPECIFIC NOT SUPPORT WITH SECURE PORT**

A secure port was specified in the BSDS but an IP address is simultaneously defined on the PORT statement in the TCP/IP profile (PROFILE.TCPIP).

DDF supports secure ports only when it is listening on the INADDR_ANY.

socket=**DVIPA ACTIVATION(***ip-address***)**

The specified IP address cannot be activated as a VIPARANGE DVIPA. DB2 does not know whether the specified IP address was intended to be used as a DVIPA. If it was intended to be used as a DVIPA, correct the error and recycle DDF to activate it. However, if the IP address was not intended to be used as a DVIPA, ignore this message.

User response: Notify the system programmer.

Problem determination: Determine the meaning of the return and reason codes issued by z/OS UNIX System Services or z/OS Communications Server.

Related tasks:

z/OS UNIX System Services Planning: Requesting transport affinity (BPXTCAFF information)

Enabling DDF for UNIX System Services (DB2 Installation and Migration)

Related reference:

z/OS UNIX System Services: Messages and Codes

z/OS Communications Server: IP and SNA Codes

DSNL513I   *csect* **TCP/IP GETHOSTBYNAME FAILED WITH ERRNO=***errno* **FOR DOMAIN** *domain*

Explanation: DB2 encountered a TCP/IP gethostbyname socket call error when attempting to convert a TCP/IP domain name into an IP address.

Variable
Description

csect   The CSECT name of the DB2 module that issued the message.

errno   The error condition reported by TCP/IP.

domain
The TCP/IP domain name that DB2 wants to convert into an IP address. DB2 obtains the *domain* value from the IPADDR column of the SYSIBM.IPNAMES communications database table or from the trusted context ADDRESS attribute defined in the SYSIBM.SYSCTXTRUSTATTRS catalog table.

System action: If the row in SYSIBM.IPNAMES containing the specified *domain* in the IPADDR column is invalid, then

- If the error occurs during DDF startup, the row in SYSIBM.IPNAMES is ignored.
- If the error occurs during an SQL statement, the SQL statement fails and the location associated with the SYSIBM.IPNAMES row is not available for TCP/IP communication.

If the trusted context ADDRESS attribute containing the specified *domain* defined in the SYSIBM.SYSCTXTRUSTATTRS is invalid, then

- If the error occurs during implicit trusted connection processing, then the error is ignored and the connection is established without any additional privileges.
- If the error occurs during explicit trusted connection processing, then the connection processing fails.

Operator response: Notify the system programmer.

System programmer response: Correct the IPADDR column value in SYSIBM.IPNAMES or register the *domain* with TCP/IP. Correct the trusted context ADDRESS attribute value using the SQL ALTER TRUSTED CONTEXT statement. register the *domain* with TCP/IP.

User response: Notify the system programmer.

Problem determination: See System Programmer Response.

Related reference:

➡ SYSIBM.IPNAMES table (DB2 SQL)

Related information:

Socket return codes (ERRNOs) (z/OS Communications Server: IP and SNA Codes)

DSNL514I   *csect-name* **TCP/IP SERVICE** *serv* **IS NOT DEFINED**

Explanation: The TCP/IP getservbyname socket call did not find a matching definition for *serv*.

Variable
     Description

serv   The TCP/IP service name that DB2 wants to convert into a TCP/IP port number. DB2 obtains the service name from the PORT column of the SYSIBM.LOCATIONS communication database table.

System action: The row in SYSIBM.LOCATIONS containing the specified service name in the PORT column is invalid. The associated LOCATION cannot be used for TCP/IP communication. If the error occurs during an SQL statement, the SQL statement fails.

Operator response: Notify the system programmer.

System programmer response: If the value specified by *serv* is incorrect, correct the PORT column value in SYSIBM.LOCATIONS.

If the value specified by *serv* is correct, register the service name with TCP/IP.

TCP/IP service names are case sensitive, so make sure that the PORT column in SYSIBM.LOCATIONS is an exact match for the service name defined to TCP/IP.

User response: Notify the system programmer.

Problem determination: See System Programmer Response.

DSNL515I   *csect-name* **TCP/IP BIND FAILED FOR PORT** *port* **WITH RETURN CODE=***return-code* **AND REASON CODE=***reason-code*

Explanation: DB2 encountered an error on the TCP/IP bind socket call while attempting to register a TCP/IP port number with the TCP/IP instance.

csect-name
     The name of the control section that issued the message.

port   The TCP/IP port number that DB2 was unable to register with the TCP/IP instance.

return-code
     The return code reported by TCP/IP, as a decimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

reason-code
> The reason code reported by TCP/IP, as a hexadecimal number. This code is issued by z/OS UNIX System Services or z/OS Communications Server.

System action: The request fails. DB2 will continue to try to activate the TCP/IP port number periodically.

Operator response: Notify the system programmer.

System programmer response: If this error occurs for the TCP/IP port that is used for incoming DRDA requests, the error is probably caused by DB2 restarting on a CEC that already has another DB2 subsystem using the DRDA port number.

If this error occurs for a TCP/IP port that is used for incoming DRDA requests to an ALIAS, the error is probably caused by some of the reasons that an error would occur for the normal DRDA request port. Check to ensure that the port value was also not previously reserved for another application in the system's TCP/IP profile.

Define the TCP/IP port number used by DB2 to the TCP/IP instance, or use the change log inventory utility to correct the TCP/IP port number used by DB2.

User response: Notify the system programmer.

Problem determination: Determine the meaning of the return and reason codes issued by z/OS UNIX System Services or z/OS Communications Server.

Related reference:

➨ z/OS UNIX System Services: Messages and Codes

➨ z/OS Communications Server: IP and SNA Codes

DSNL516I   *csect-name* **LANGUAGE ENVIRONMENT SERVICES NOT AVAILABLE. CEEPIPI ROUTINE CANNOT BE LOADED**

Explanation: DB2 must load the Language Environment interface module (CEEPIPI) to process functions requiring Language Environment services. These Language Environment services are required for TCP/IP socket calls, encryption and decryption, and so on. If the CEEPIPI module is not included in an accessible data set such as STEPLIB data sets or the LPA library, DDF is unable to use the Language Environment.

System action: DDF startup continues, but DDF users cannot use DB2 functions that require Language Environment services.

System programmer response: Add the Language Environment library to the DDF STEPLIB concatenation or to the LPA library. The Language Environment library must be APF authorized when it appears in the DDF STEPLIB concatenation.

Problem determination: Collect the following diagnostic item listed in

• Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

.

DSNL517I   *csect-name* **CEEPIPI** *func* **SERVICE FAILED WITH RETURN CODE=***return-code*

Explanation: DB2 uses the Language Environment CEEPIPI service to issue certain TCP/IP socket calls. The Language Environment return code *return-code* was not expected for the indicated Language Environment function call.

func   The function requested by DB2. Possible values are:

> **add_entry(module-name)**
> > Add an entry to the Language Environment PIPI table dynamically. Module-name is the name of the module that failed to be added to the PIPI table.
>
> **call_sub**
> > Run a program that issues a TCP/IP socket call using an existing Language Environment run time environment.
>
> **del_sub**
> > Delete an entry from the Language Environment PIPI table dynamically.

init_sub
> Create Language Environment sub run time environment.

term   Terminate the Language Environment run time environment.

return-code
> The return code received from CEEPIPI.

System action: The attempt to issue the Language Environment request fails.

If this message is issued during DDF startup, DDF startup continues.

If this message is issued during an SQL statement, a -904 SQLCODE is returned to the calling application.

System programmer response: See Problem Determination.

Problem determination: Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

If the function requested by DB2 was *add-entry(module-name)*, an incorrect link edit of the module could create unresolved references. To determine if link edit problems exist, run the z/OS AMBLIST utility with LISTLOAD and OUTPUT=XREF.

Related reference:

▷ z/OS Communications Server: IP and SNA Codes

Related information:

▷ -904 (DB2 Codes)

DSNL518I   *csect-name* **LE/370 ENCLAVE FAILED WITH RETURN CODE=***erc* **AND FEEDBACK=***fi*

Explanation: DB2 uses the Language Environment CEEPIPI call_sub service to execute certain TCP/IP socket calls. The CEEPIPI interface returned information indicating a failure occurred while executing the TCP/IP socket.

erc   The Language Environment enclave return code.

fi   Contains one or two message identifiers. The message identifiers are constructed from the Language Environment condition token.

> The message identifiers are constructed as follows:
> * Single message: Facility_Id combined with Msg_No
> * Two messages: Facility_Id combined with Class_Code and Facility_Id combined with Cause_Code

System action: The SQL statement fails with SQLCODE -904.

System programmer response: A description of the Language Environment enclave return code and the condition token are documented in *Language Environment for OS/390 & VM Programming Guide*.

Problem determination: Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* SYSABEND, SYSUDUMP, or SYSMDUMP output.
.

DSNL519I   *csect* **TCP/IP SERVICES AVAILABLE FOR DOMAIN** *domain* **AND PORT** *tcpport*

Explanation: DB2 is ready to accept connections on any IP address supported by the TCP/IP stack. This message is issued when re-initialization is successful. DB2 will not accept any connection request until this message is issued.

Variable
> **Description**

csect   The CSECT name of the DB2 module that issued the message.

domain
The TCP/IP domain name used by this DB2 subsystem to accept incoming connection requests.

tcpport The TCP/IP port number used by this DB2 subsystem to accept incoming connection requests.

System action: The DB2 TCP/IP port number is available to accept incoming connection requests on any IP address supported by the TCP/IP stack.

DSNL520I   *csect* **TCP/IP SERVICES AVAILABLE FOR RESPORT** *port*

Explanation: DB2 encountered a TCP/IP failure during DDF startup processing. The DSNL515I message identifies the cause of the failure. After the failure is detected, DDF periodically attempts to reinitialize TCP/IP to allow TCP/IP resynchronization to occur. This message is issued when reinitialization is successful.

Variable
    Description

csect The CSECT name of the DB2 module that issued the message.

port The TCP/IP port number used by this DB2 subsystem to accept incoming resynchronization connection requests.

System action: The DB2 TCP/IP port is available to accept incoming resynchronization connection requests.

DSNL521I   *csect-name* **INVALID DDF BSDS VALUES DETECTED (PORT=***port***,RESPORT=***rport* **SECPORT=***sport***).**

Explanation: DB2 encountered invalid values for the TCP/IP port numbers in the DDF record of the BSDS. Valid TCP/IP port numbers must obey the following rules:
- If PORT or SECPORT is non-zero, RESPORT must also be non-zero.
- If RESPORT is non-zero, PORT or SECPORT must also be non-zero.
- PORT, RESPORT, and SECPORT must not specify the same value.

Variable
    Description

port The TCP/IP port number used by this DB2 subsystem to accept incoming DRDA SQL connection requests.

rport The TCP/IP port number used by this DB2 subsystem to accept incoming two-phase commit resynchronization requests.

sport The TCP/IP port number used by this DB2 subsystem to accept incoming secure DRDA connection requests.

System action: DDF startup terminates abnormally.

System programmer response: Use the Change Log Inventory utility (DSNJU003) to correct the TCP/IP port numbers used by DB2.

DSNL523I   *csect* **TCP/IP SERVICES AVAILABLE FOR IP ADDRESS** *ipaddr* **AND PORT** *port*

Explanation: DB2 is ready to accept connections on the specified IP address and port.

Variable
    Description

csect The CSECT name of the DB2 module that issued the message.

ipaddr The IP address used by this DB2 subsystem to accept incoming connection requests.

port The port number used by this DB2 subsystem to accept incoming connection requests. It can be a group SQL port, a resync port or an alias port.

System action: DDF is available to accept incoming requests on the specified IP address and port. If the IP address is specified on the PORT statement of the TCPIP profile, DB2 accepts connections only on the specified IP address. If the IP address is specified by either the DSNJU003 utility or the MODIFY DDF command, DB2 accepts connections on any IP address active on the stack, including the specified IP address, regardless of whether a domain name is provided or not.

DSNL600I • DSNL700I

This message is issued each time that the listener is re-initialized to accept requests for the specified IP address and port.

System programmer response: If you want DB2 to accept connections on any IP address, including the specified IP address, then the IP address must be specified in the BSDS. Specifying a domain name is optional when the IP address is specified in the BSDS.

DSNL600I   **RESOURCE UNAVAILABLE CORRELATION ID=**_correlation-id_ **CONNECTION ID=**_connection-id_ **LUW ID=**_luw-id=token_ **REASON=**_reason-code_ **TYPE=**_resource-type_ **LOCATION=**_location-name_ **NAME=**_resource-name_

Explanation: This message is produced at the requesting site when a distributed request requires a resource that is not available.

The resource identified by 'resource-name' and 'resource-type' was not available at 'location-name' due to the indicated reason code.

correlation-id
> The correlation identifier of the thread associated with the requester of the resource. Correlation IDs beginning with '0nn' where 'nn' is a number from 01 to 28, identify system agents.

connection-id
> The connection identifier of the requester.

luw-id=token
> The logical unit of work ID of the thread followed by an '=' sign and a token which may be used in place of luw-id in any DB2 command that accepts luw-id as input.

Refer to message DSNT500I for an explanation of the resource type and resource name.

System action: The SQL statement cannot be executed.

Operator response: See Problem Determination.

Problem determination: The reason the resource is unavailable is identified by the reason code. Collect the information described for the reason code received.

Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

Related concepts:

DB2 reason codes (DB2 Codes)

Related information:
"DSNT500I" on page 555

DSNL700I   _csect-name_ **CDB RESOURCE UNAVAILABLE, DEFAULTS MAY BE USED REASON=**_reason_ **TYPE=**_type_ **NAME=**_name_

Explanation: This message indicates that a requested function required the CDB which is currently unavailable. Defaults can be used to allocate conversations or set conversation limits with VTAM and TCP/IP partners.

System action: If the SYSIBM.MODESELECT table is unavailable, VTAM defaults are used to allocate conversations. If the the SYSIBM.LUMODES table is unavailable, VTAM defaults can be used for conversation limits. If the SYSIBM.LOCATIONS, SYSIBM.IPNAMES, SYSIBM.LULIST, or SYSIBM.LUNAMES tables are unavailable, conversations can not be allocated to remote LOCATIONs.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Note the REASON code, the TYPE code, and the NAME. See Problem Determination section for an explanation of the collected data.

Problem determination: The reason the resource is unavailable is identified by a reason code. Use the table defined for message DSNT500I to find the TYPE of resource identified in the message.

Related concepts:

▶ SQL codes (DB2 Codes)

▶ Communications database for the requester (Managing Security)

▶ Communications database for the server (Managing Security)

Related information:

"DSNT500I" on page 555

▶ Recovering from communications database failure (DB2 Administration Guide)

DSNL701I   **CDB OBJECT** *object-name* **DOES NOT EXIST**

Explanation: The communications database (CDB) does not contain *object-name*. The incorrect *object-name* can be the database name, the table space name, or the table name.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine what caused the failure, define the object as required, and issue the -START DDF command.

Problem determination: Print the definition of the communications database (CDB).

DSNL702I   **CDB TABLE** *table-name* **DOES NOT HAVE PROPER COLUMN DEFINITIONS, COLUMN** *column-name* **IS MISSING OR INCORRECTLY DEFINED**

Explanation: The communications database (CDB) needs a definition for 'column-name' in 'tablename'. The current definition is missing or incorrect.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine what caused the failure, define the column as required, and issue the -START DDF command.

Problem determination: Print the definition of the communications database (CDB).

DSNL703I   **CDB TABLE** *table-name* **DOES NOT HAVE PROPER INDEX DEFINITIONS, INDEX** *index-name* **IS MISSING OR INCORRECTLY DEFINED**

Explanation: The communications database (CDB) requires an index name for *table-name*. The current definition is missing or incorrect.

System action: The distributed data facility (DDF) terminates.

Operator response: Notify the system programmer.

System programmer response: Define the index as required and issue the -START DDF command.

Problem determination: Print the definition of the communications database (CDB).

Related tasks:

▶ Populating the communications database (DB2 Installation and Migration)

Related reference:

▶ -START DDF (DB2) (DB2 Commands)

Related information:

▶ Recovering from a problem with a communications database that is incorrectly defined (DB2 Administration Guide)

DSNL704I   CDB OBJECT *object-name* DOES NOT HAVE PROPER REFERENTIAL CONSTRAINT
DEFINITIONS

Explanation: The communications database (CDB) requires referential constraint definitions for *object-name*. The current definition is missing or incorrect.

System action: Termination of the distributed data facility (DDF) is initiated.

Operator response: Notify the system programmer.

System programmer response: Determine what caused the failure, define the missing referential constraint, and issue the -START DDF command.

Problem determination: Print the definition of the communications database (CDB).

Related concepts:

➦ The role of the communications database (CDB) (DB2 Installation and Migration)

Related tasks:

➦ Populating the communications database (DB2 Installation and Migration)

➦ Populating the communications database for use with TCP/IP (DB2 Installation and Migration)

Related reference:

➦ -START DDF (DB2) (DB2 Commands)

DSNL705I   CDB IS UNAVAILABLE OR INCORRECTLY DEFINED

Explanation: During start-up of the distributed data facility, the required CDB was unavailable or incorrectly defined.

System action: Termination of the DDF is initiated.

Operator response: Notify the system programmer.

System programmer response: Refer to the previous message that specifies the cause of the CDB failure (DSNL700I, DSNL701, DSNL702I, DSNL703I, or DSNL704I). Determine what caused the failure, fix the error condition, and issue the -START DDF command.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

.

DSNL706I   *csect-name* AN ERROR HAS OCCURRED IN DDF EXIT ROUTINE *routine*

Explanation: An error occurred following invocation of a DDF exit routine by DB2.

The variable "routine" is the name of the exit routine, DSNLHLIx, invoked by DB2.

System action: The system abends the failing database-access agent.

DSNL800I   *csect-name* EXECUTION FAILED BECAUSE THE BIND DEPLOY FUNCTION IS NOT SUPPORTED
BY THE SERVER WHICH CAUSED TERMINATION OF CONNECTION: LOCATION
connection-location, PRODUCT ID *product-identifier*

Explanation: The current BIND command failed because the command was routed to a server that does not support the bind deploy function.

csect-name
        The name of the control section that issued the message.

connection-location
        The name of the server.

product-identifier
> An alphanumeric string that identifies the product. The format is *pppvvrrm*:

> *ppp*   A 3-character code that identifies the product:

>> **AQT**   IBM DB2 Analytics Accelerator for z/OS

>> **ARI**   DB2 for VSE & VM

>> **DSN**   DB2 for z/OS

>> **JCC**   IBM Data Server Driver for JDBC and SQLJ

>> **QSQ**   DB2 for i

>> **SQL**   DB2 for Linux, UNIX, and Windows

> *vv*   A 2-digit number that identifies the product version number, such as 09 or 10.

> *rr*   A 2-digit number that identifies the product release level, such as 01 or 05.

> *m*   A 1-digit number that identifies the product modification level, such as 0 or 1.

System action: The BIND command fails. The package is not deployed.

Programmer response: Ensure that the database server is DB2 for z/OS Version 9 new-function mode or greater to support bind deploy function.

Chapter 10. DSNM messages

Message identifiers that begin with DSNM communicate information about the IMS attachment facility.

All of the DSNM messages are sent to the IMS master terminal operator (MTO) and to the appropriate MVS console.

DSNM001I   IMS/TM xxxx CONNECTED TO SUBSYSTEM yyyy

Explanation: This is an information message only. The control region of the IMS/TM system specified in the message as 'xxxx' has successfully connected to the DB2 subsystem specified in the message as 'yyyy'. The message is sent to the IMS/TM master terminal operator.

System action: IMS/TM transactions can access DB2 resources.

DSNM002I   IMS/TM xxxx DISCONNECTED FROM SUBSYSTEM yyyy RC=rc

Explanation: The IMS/TM system named 'xxxx' control region has disconnected from the DB2 subsystem named 'yyyy' for the RC, reason code, of 'rc'. The message is sent to the IMS/TM master terminal operator. The reason code values are:

Value   Description

A   IMS/TM is terminating normally (for example, /CKPT FREEZE command).

B   IMS/TM is terminating abnormally.

C   DB2 is terminating in a quiesce fashion (for example, DB2 stop quiesce command).

D   DB2 is terminating abnormally.

E   The connection between IMS/TM and the DB2 subsystem is terminating; however, both subsystems remain operational.

System action: The connection between IMS/TM and DB2 terminated. IMS/TM transactions can no longer access DB2 resources.

Operator response: Notify the system programmer.

System programmer response: Reason code A, C, or E makes this an information message. Reason code D indicates that a serious problem occurred in the DB2 subsystem, and that the SYS1.LOGREC should be examined. For reason code B or D, determine the failing subcomponent.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: SYS1.LOGREC and/or dump information may be available for reason codes B and D.

DSNM003I   IMS/TM *ims/tm-name* FAILED TO CONNECT TO SUBSYSTEM *subsystem-name* RC=*return-code*

Explanation: The IMS/TM system (named *ims/tm-name*) control region did not connect to the DB2 subsystem (named *subsystem-name*). This message is sent to the IMS/TM master terminal operator. The *return-code* value indicates the reason for the failure.

Value   Description

00   The DB2 subsystem is not operational. However, a notify message has been queued and, when the DB2 subsystem is started, it will automatically connect to IMS/TM.

04   The connection attempt failed between IMS/TM and the DB2 subsystem.

System action: IMS/TM transactions cannot access DB2 resources.

Operator response: For return code 00, start the DB2 subsystem. For return code 04, notify the system programmer.

Problem determination: For return code 04, an IMS/TM attachment facility snap record (code X'5501FF') is written to the IMS/TM log to indicate the reason for the failure.

Related tasks:

▷ Controlling CICS connections (DB2 Administration Guide)

▷ Controlling IMS connections (DB2 Administration Guide)

Related information:

▷ Recovering DB2 when the CICS attachment facility cannot connect to DB2 (DB2 Administration Guide)

▷ Recovering from IMS-related failures (DB2 Administration Guide)

DSNM004I   RESOLVE INDOUBT ENTRY(S) ARE OUTSTANDING FOR SUBSYSTEM *subsystem-identifier*

Explanation: The *subsystem-identifier* subsystem has remaining indoubt entries after the IMS/TM subsystem has attempted to resolve the work units that are indoubt. This message is sent to the IMS/TM master terminal operator.

System action: The connection between the DB2 and IMS/TM subsystems remains active, and applications can access DB2 resources.

Operator response: Notify the system programmer.

System programmer response: Use the DISPLAY THREAD and RECOVER INDOUBT commands to display and resolve the work units that are indoubt.

Problem determination: Recovery information is available on the DB2 log and IMS/TM log to assist in the resolution of the work unit(s) indoubt. For each indoubt work unit, an IMS/TM attachment facility snap record (code X'5501FE') is written to the IMS/TM log.

Related tasks:

▷ Resolving indoubt units of recovery (DB2 Administration Guide)

Related reference:

▷ -DISPLAY THREAD (DB2) (DB2 Commands)

▷ -RECOVER INDOUBT (DB2) (DB2 Commands)

DSNM005I   IMS/TM RESOLVE INDOUBT PROTOCOL PROBLEM WITH SUBSYSTEM *subsystem-identifier*

Explanation: The *subsystem-identifier* subsystem has protocol incompatibilities with IMS/TM. The DB2 resolve indoubt process received an unknown or invalid recovery request (RRE) from IMS/TM. The request was rejected by DB2 and kept by IMS/TM. The message is sent to the IMS/TM master terminal operator.

System action: The connection between the DB2 and IMS/TM subsystems remains active, and applications can access DB2 resources.

Operator response: Notify the system programmer.

System programmer response: Call the support center after extracting from the IMS/TM log the IMS/TM attachment facility snap records (code X'5501FE') using the IMS/TM utility DFSERA10.

These records contain the complete request from IMS/TM to DB2, the request type is at offset X'01C', this is a four-character identifier which should contain any of the following words:

* 'SHOW'
* 'IDBT'
* 'COMM'
* 'ABRT'

If the IMS/TM recovery request (RRE) found in the log is an invalid one, (not 'COMM' or 'ABRT'), you may manually delete it using the IMS/TM/CHANGE SUBSYS RESET command after consulting with the IBM Software Support.

If the IMS/TM request is a valid one (abort or commit), you must find the action that IMS/TM performed for that RRE by locating the X'3730' (commit) or X'3801' (abort) record in the IMS/TM log and perform the same action under DB2 by using the -RECOVER INDOUBT command.

Problem determination: Recovery information is available on the DB2 log and IMS log to assist in the resolution of the problem. For each indoubt work unit, an IMS/TM attachment facility snap record (code X'5501FE') is written to the IMS/TM log.

Related tasks:

➡ Resolving indoubt units of recovery (DB2 Administration Guide)

Related reference:

➡ -RECOVER INDOUBT (DB2) (DB2 Commands)

DSNM100I   SUBSYSTEM xxxx NOT AVAILABLE

Explanation: The IMS/TM command facility which routes commands to the DB2 subsystem cannot perform its functions because the DB2 subsystem named 'xxxx' is considered not operational.

System action: The connection to the DB2 subsystem failed for the command. The command is rejected.

System programmer response: If the DB2 subsystem is operational, and if you suspect an error in DB2, you might need to report the problem.For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. The person entering the request may not be authorized to execute the command. If the DB2 subsystem is not operational, DB2 commands from IMS/TM cannot be processed.

Problem determination: A snap record (X'5501FF') is written to the IMS/TM log if a connection request failed.

DSNM104I   xxxx COMMAND FAILED

Explanation: The DB2 command specified by xxxx failed to complete.

System action: The command terminated without providing an output message. Commands will still be allowed by the IMS/TM attachment facility.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: If the command failed because of an abend, a snap record (X'5501FF') is written to the IMS/TM log and SYS1.LOGREC information is available.

Chapter 11. DSNP messages

Message identifiers that begin with DSNP communicate information about the data space manager.

The DSNP messages contain a connection ID, a correlation ID, and a logical-unit-of-work ID. Messages that have the same connection ID, correlation ID, and logical-unit-of-work ID relate to the same user. This makes it possible to scan a number of DSNP messages and group activities involving a given user.

DSNP001I *csect-name* **-** *data-set-name* **IS WITHIN** *n* **BYTES OF AVAILABLE SPACE. RC=**reason-code
CONNECTION-ID=connection-identifier**, CORRELATION-ID=**correlation-identifier**, LUW-ID=**logical-unit-of-work-id**=**token

Explanation: This message warns of an impending space shortage. The data space manager (DSM) extend function CSECT 'DSNPmmmm' detected the space shortage through the use of a 'hedge' quantity *n*. Reason code *reason-code* gives some information about why space could not be obtained.

There might also be some preceding DSM and VSAM messages that can help to determine why space cannot be obtained. The connection, correlation, and logical-unit-of-work IDs identify the user or application that encountered the problem. Correlation IDs beginning with '0*nn*', where *nn*is a number, identify system agents.

A LUW-ID of an asterisk (*) indicates that the thread does not have a LUW-ID. If the LUW-ID is not an asterisk (*), the value is a *logical-unit-of-work identifier* that is assigned to the thread followed by an equals (=) sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: If sufficient space is available for the current request, continue. Otherwise, message DSNP007I follows, and the current SQL request fails.

Operator response: DSM messages are directed to the database administrator (DBA) or system administrator. However, the DBA or system administrator can request that the operator issue an MVS WRITELOG operator command (for example, 'W A') to obtain MVS SYSLOG output. The SYSLOG sometimes contains media manager messages that might be helpful to the DBA or system administrator in determining why space could not be allocated.

System programmer response: Based on the reason code (RC) in the message, as well as information that might be available in access method services or VSAM messages preceding this one, try one of the following solutions:

- If you do not anticipate further extension of this table space or index, ignore the message.
- If the table space or index is backed by storage group-defined data sets and this data set is not at maximum extents, use the ALTER STOGROUP statement to add volumes to the storage group (up to 133 of them).
- If the table space or index is backed by 'user-defined' data sets, and not at maximum extents for the data set, issue an ALTER ADDVOLUMES access method services command to provide additional volumes (up to 133 of them). Or, if the table space is nonpartitioned and the high allocated RBA is greater than or equal to 2GB (2,147,483,648) for the indicated data set, define the data set required for the next sequential piece. (Actually, in this case, there should be an indication in console messages or the reason code that the data set required for the next piece was not found.)
- If the data set is at the maximum number of extents, recover or reorganize the table space or index after taking one of the following actions:
  - Enlarge the primary or secondary space allocation quantity of the table space or index.
  - If the DB2 subsystem is running on z/OS V1.7 or later, and the data set is not shared with any z/OS systems at an earlier level, convert the data set to SMS-managed with the Extent Constraint Removal option set to YES in the SMS data class. If you do this, the maximum number of extents is 7257.
- If all of the following conditions are true, the maximum number of extents is 7257:
  - The z/OS level is V1.7 or later

– The data set is SMS-managed

– The Extent Constraint Removal option is set to YES in the associated SMS data class

• If any of the conditions outlined in the previous bullet are not true, the maximum number of extents is:

– 251 to 255 for non-striped data sets

– 251 to 255 for each stripe for striped data sets

Problem determination: Examine the console messages, SYS1.LOGREC information, and SYSLOG to determine the reason for the condition. Try to resolve it by the methods described in the Operator Response section of this message. If you feel that this message indicates a problem in an IBM program, you should provide the following sources of information about the problem:

• Console sheet showing the message and related messages that might precede or follow it.

• System log printout (obtained via MVS WRITELOG operator command) showing related VSAM/media manager messages that might appear there (usually needed only if you have a message or reason code that indicates a media manager services problem occurred).

• Integrated catalog facility (ICF) catalog listing showing characteristics of the data set in question and of the ICF catalog itself.

• VTOC listing of packs associated with the request, showing space available for data sets, catalogs, VTOC index, and the VTOC itself, as well as data set extent information.

• Any associated dumps.

• SYS1.LOGREC listing showing entries at or near the time of the problem (unless these appear in an associated dump).

• Depending on the reason code, the results of certain SQL SELECT operations on DB2 catalog tables may be needed.

Related concepts:

➡ Primary space allocation (DB2 Administration Guide)

➡ Example of primary and secondary space allocation (DB2 Administration Guide)

Related tasks:

➡ Extending user-managed data sets (DB2 Administration Guide)

➡ Extending a data set (DB2 Administration Guide)

➡ Letting SMS manage your DB2 storage groups (DB2 Administration Guide)

DSNP002I   *csect* **- DEFINE FAILED FOR** *data-set-name* **RC=***reason-code* **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **LUW-ID=***logical-unit-of-work-id* **=***token*

Explanation: Data set definition failed (in CSECT 'DSNP*mmmm*') for the reason identified by RC. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem. The connection, correlation, and logical-unit-of-work IDs identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: The data space manager (DSM) terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to either the database administrator or the system administrator. Bring the message to their attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the reason code and preceding access method services messages, if any, and correct it, if possible. If an out-of-space condition exists, see the description of message DSNP001I for actions to consider.

Problem determination: See Problem Determination of message DSNP001I for this information.

Related concepts:

↪ DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP003I   *csect* **- RESET FAILED FOR** *data-set-name*. **RC=**rrrrrrrr **CONNECTION-ID=**xxxxxxxx,
CORRELATION-ID=yyyyyyyyyyyyy, **LUW-ID=**logical-unit-of-work-id=token

Explanation: Reset, a data space manager (DSM) function invoked by DB2 utilities, failed in CSECT 'DSNPmmmm'
for the reason identified by RC. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the
user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number,
identify system agents.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a
logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in
place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it
possible to scan a number of DSNP-prefixed messages and tie together activities involving a given user.

System action: DSM terminates its actions associated with the current SQL request. Higher level functions might
terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to
that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2
messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the reason code and preceding DB2, access
method services, or VSAM messages, if any, and correct it, if possible.

It might be necessary to delete and redefine the data set using access method services. Consider making a backup
copy before doing this. (Use DB2 utilities to make the copy; in general, VSAM utilities do not work on DB2 database
services data sets.)

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

↪ DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP004I   *csect* **- DELETE FAILED FOR** *data-set-name*. **RC=**reason-code **CONNECTION-ID=**connection-id,
CORRELATION-ID=correlation-id, **LUW-ID=**logical-unit-of-work-id=token

Explanation: Deletion failed in CSECT 'DSNPmmmm' for the reason identified by the RC=reason-code value. The
connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered
the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

The value of the LUWID is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a
token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts
logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUWID relate to the same user. This makes it
possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: DB2 terminates its actions associated with the current SQL request. DB2 might terminate the current
SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to
that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2
messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the reason code and preceding access method

services messages if any, and correct it, if possible. Check the data set and delete it by using access method services directly if reasonable to do so. Consider making a backup copy before doing this. (Use DB2 utilities to make the copy; in general, VSAM utilities do not work on DB2 data sets.)

Problem determination: See Problem Determination for message DSNP001I for this information.

Severity: 8 (error)

Related concepts:

DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP005I   *csect* **- COULD NOT REMOVE EXCESS VOLUMES FROM CANDIDATE LIST OF** *data-set-name* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: Volumes from a storage group were entered in the data set's candidate list and could not be removed by CSECT 'DSNPmmmm' after some data space manager (DSM) operation. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

This is not a major problem. However, the data set will not properly reflect subsequent deletions from the volume list in its storage group (via ALTER STOGROUP command in SQL), because the data set's integrated catalog facility (ICF) catalog entry still contains the volume serial numbers.

If needed, the storage group in question can be identified by issuing an SQL SELECT to get information about the table space or index that is identified by the third and fourth qualifiers (database name, table space name) in the data set name.

System action: Operation continues.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Look for associated DSM and access method services messages preceding this one to get additional information. Attempt an ALTER REMOVEVOLUMES access method services command, if appropriate, after issuing a LISTCAT to see which volumes are recorded as 'candidate' in the integrated catalog facility (ICF) catalog.

Problem determination: See Problem Determination for message DSNP001I for this information. (However, there should not be a need for VTOC or SYSLOG information for this case.)

DSNP006I   *csect* **- REQUIRED STOGROUP DEFINED DATA SET NOT AVAILABLE-** *data-set-name* **RC=***rrrrrrrr* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: An attempt to alter the shareoptions of this data set failed. The data set could not be accessed for the reason indicated by RC. The connection ID, correlation ID, and logical-unit-of-work ID (LUWID) identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

The value of the LUWID is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of the logical-unit-of-work-id in any DB2 command that accepts the logical-unit-of-work-id as input.

Messages having the same connection ID, correlation ID, and LUWID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: DB2 terminates its actions associated with the current SQL request. DB2 might terminate the current

SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: See Problem Determination for message DSNP001I for this information.

Severity: 8 (error)

Related concepts:

⏩ DB2 reason codes (DB2 Codes)

Related information:

"DSNP001I" on page 389

DSNP007I   *csect-name* **- EXTEND FAILED FOR** *data-set-name*. **RC=**reason-code **CONNECTION-ID=**connection-id, **CORRELATION-ID=**correlation-id, **LUW-ID=**logical-unit-of-work-id **=** *token*

Explanation: Extension failed in control section DSNP*mmmm* of the data space manager (DSM). The *data-set-name* value identifies the data set that is being extended. The *reason-code* value is the reason code, which indicates why the extend process failed. The *connection-id*, *correlation-id*, and *logical-unit-of-work-id* values identify the user or application that encountered the problem. Correlation identifiers begin with '0*nn*', where *nn* is a number that identifies system agents.

When the *logical-unit-of-work-id* value is '*' (asterisk) the thread does not have a LUWID. If the *logical-unit-of-work-id* value is not '*', the *logical-unit-of-work-id* value is a logical unit of work identifier (LUWID) that is assigned to the thread, followed by an '=' (equal sign) and a *token* value, which can be used in place of the LUWID in any DB2 command that accepts the LUWID value as input.

Messages that have the same *connection-id*, *correlation-id*, and *logical-unit-of-work-id* values relate to the same user. This situation makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: The DSM terminates its actions associated with this extend request. Higher level functions might terminate if they cannot continue without a successful extend. In this case, additional console messages and/or SQL codes are produced. In other cases, higher level functions might be able to resolve the space need without the extend, so they continue normally.

Operator response: See Operator Response for message DSNP001I for this information.

System programmer response: The following sources of information are available to help determine the cause of the failure:

- See the Programmer Response for message DSNP001I.
- Analyze the *reason-code* value in conjunction with the console messages.
- The reason code might apply to the data set identified in the message or to another data set that is needed for the extend process. If the reason code indicates a data set access failure, the console messages indicate the data set name. For example, if the RC indicates 'resource unavailable', console messages identify the resource that is unavailable. Most likely, the message is DSNT500I.
- Retrieve diagnostic information from the output of the `DISPLAY DB...RESTRICT` or `DISPLAY UTIL(*)...` commands.
- For information about finding the system diagnostic work area (SDWA), see DB2 Diagnosis Guide and Reference.

Problem determination: See Problem Determination for message DSNP001I.

DSNP008I

Related concepts:

Chapter 11, "DSNP messages," on page 389

➡ DB2 reason codes (DB2 Codes)

➡ Recovery of catalog and directory tables (DB2 Administration Guide)

Related tasks:

➡ Using disk space effectively (DB2 Performance)

➡ Improving the performance of sort processing (DB2 Performance)

➡ Recovering catalog and directory objects (DB2 Utilities)

➡ Displaying threads by LUWID (DB2 Administration Guide)

Related reference:

➡ Table spaces and indexes (DB2 SQL)

➡ -DISPLAY UTILITY (DB2) (DB2 Commands)

➡ -DISPLAY DATABASE (DB2) (DB2 Commands)

Related information:

"DSNP001I" on page 389

DSNP008I   *csect-name*- **REQUIRED USER-DEFINED DATA SET NOT AVAILABLE-***data-set-name* **RC=***reason-code* **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: One of the following actions applied against user-defined data sets failed in CSECT 'DSNP*mmmm*' of the data space manager (DSM):

- Create table space or index
- Extend table space or index
- Process the SQL ALTER DATABASE statement
- Process the REORG TABLESPACE utility with SHRLEVEL REFERENCE or CHANGE

Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

One or more of the data sets could not be accessed for the reason indicated by RC. The connection ID, correlation ID, and logical-unit-of-work ID (LUWID) identify the user or application that encountered the problem.

The value of the LUWID is alogical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages having the same connection ID, correlation ID, and LUWID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

If the name of the data set in the message is different from the name of the original data set (for example, '....A002' instead of '...A001'), a new data set might need to be created. A different name indicates that the original data set exceeded its limit.

System action: DB2 terminates its actions associated with the current SQL request or utility. DB2 might terminate the current SQL request or utility or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: The return code description might indicate that one of the following actions needs to be performed or it indicates the necessary steps:

- If the data set does not exist, create it before attempting the create table space/index or extend data set again.

- If the data set shareoptions are incorrect, modify the shareoptions with access method services (AMS).
- The appropriate AMS command is ALTER SHAREOPTIONS.

Problem determination: See Problem Determination for message DSNP001I for this information.

Severity: 8 (error)

Related concepts:

DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP009I   **THE FOLLOWING ERROR MESSAGES WERE RECEIVED FOR** *operation* **ON** *data-set-name*

DSNP010I   **END OF ERROR MESSAGES. CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*,
LUW-ID=*logical-unit-of-work-id=token*

Explanation: Messages DSNP009I and DSNP010I enclose a block of messages that are received by the data space manager (DSM) subcomponent of DB2 in response to an access method services (AMS) operation.

operation
> The access method services operation that was being performed.

data-set-name
> The data set on which the operation was being performed.

The prefix of the messages in the block is usually IDC (access method services messages). However, the prefix can also be IKJ (dynamic allocation) or other system subcomponent IDs. Refer to the appropriate MVS publication for descriptions of the enclosed messages.

The operation can be DEFINE CLUSTER, ALTER ADDVOLUMES, ALTER REMOVEVOLUMES, ALTER NEWNAME, or DELETE CLUSTER. One or more other DSM messages follow this message to explain what happened as a result of the failure of this access method services request. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: Operation continues.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Use this information to help resolve problems indicated by other DSM messages that follow this one.

Problem determination: See Problem Determination for message DSNP001I for this information.

DSNP011I   *csect-name* **- MEDIA MANAGER SERVICES ERROR FOR** *data-set-name*. **MMRC=***cccccccc*
DSMRC=*ffffffff* **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*,
LUW-ID=*logical-unit-of-work-id =token*

Explanation: The MMRC is the return code that DB2 received from media manager services. Refer to the appropriate MVS publication for a description of these codes. DSMRC is a DSM reason code that indicates which of the media manager services functions, update or extend, failed. The message is from CSECT 'DSNP*mmmm*' of the data space manager (DSM). The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the

user or application that encountered the problem. Correlation IDs beginning with '0*nn*', where *nn* is a number, identify system agents.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: The DSM terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: DSM messages are directed to either the database administrator (DBA) or the system administrator. However, they can request that the operator issue an MVS WRITELOG operator command (for example, 'W A') to obtain MVS SYSLOG output. The SYSLOG sometimes contains media manager messages that might be helpful to the DBA or system administrator in determining why space could not be allocated.

System programmer response: This message might be preceded by one or more IEC or ICY messages on the console or in SYSLOG. These messages help to determine what happened and what action to take.

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

DB2 reason codes (DB2 Codes)

Related information:

DSNP012I   *csect* **- ERROR IN ICF CATALOG LOCATE FUNCTION FOR** *data-set-name* **CTLGRC=***cc*
CTLGRSN=*zzzzrrss* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*,
LUW-ID=*logical-unit-of-work-id=token*

Explanation: CTLGRC and CTLGRSN are the return and reason codes that DB2 received from SVC 26 call (ICF catalog locate function). The reason code is in the same format as that returned to DB2 by the SVC 26 call.

If CTLGRC is 4 or 8, CTLGRSN is associated with access method services message IDC3009I. To determine the name of the CSECT that set CTLGRSN, convert *zzzz* from hexadecimal to EBCDIC, and append the resulting two characters to 'IGG0CL'. For example, suppose that *zzzz* is C6C6. The EBCDIC equivalent of X'C6C6' is 'FF', so the CSECT that set the reason code is IGG0CLFF.

rr is the hexadecimal equivalent of the reason code value that is documented in access method services message IDC3009I. *ss* is the hexadecimal equivalent of the return code value that is documented in access method services message IDC3009I (the return code follows the reason code.) Convert *rr* and *ss* to decimal. Then see the description of IDC3009I for explanations of the reason code and return code. For example, suppose that *rrss* is 0532. The decimal equivalent of X'32' is 50, and the decimal equivalent of X'05' is 5. Therefore, in IDC009I, you need to look up a return code of 50 and a reason code of 5.

If CTLGRC=AAAAAA08 and CTLGRSN=AAAAAA08, CTLGRSN is not associated with access method services message IDC3009I. AAAAAA08 indicates that data-set-name is not in the ICF catalog.

The connection ID, correlation ID, and LUW-ID identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

An LUW-ID with an asterisk (*) indicates that the thread does not have an LUW-ID. If the LUW-ID is not an asterisk (*), the value is a *logical-unit-of-work-id* that is assigned to the thread followed by an equals sign (=) and a token value, which can be used in place of *logical-unit-of-work-id* in any DB2 command that accepts *logical-unit-of-work-id* as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: DB2 terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

System programmer response: Correct the problem indicated by the catalog return and reason codes.

Problem determination: See Problem Determination for message DSNP001I for this information.

DSNP013I **INSUFFICIENT PRIMARY ALLOCATION SPACE TO FORMAT CONTROL PAGES IN DATA SET** *data-set-name* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id* =*token*

Explanation: Data space manager (DSM) discovered that the primary space quantity given in the data set definition was not large enough. Formatting of required control information in the first several pages was not possible. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: DSM terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: This should occur only for table spaces or indexes that are not storage group defined. (The DB2 interpreter forces the initial allocation to be large enough if a storage group is used.) Check both the page size specification (4KB or 32KB) and the number of control pages required (2 for table space, 3 for index space). Specify the primary allocation value again so that at least (pages multiplied by page size) bytes are allocated.

Problem determination: See Problem Determination for message DSNP001I for this information.

DSNP014I **IRLM MANUAL LOCK REQUEST FAILED FOR** *data-set-name*. **RC=***reason-code*,**SC=***subcode*, **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: DB2 requested a manual lock on a data set, but the lock could not be obtained for reasons specified in RC and SC. RC and SC are the return code and reason code from IRLM. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. DSNP001I or DSNP007I follow this message. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

If message DSNT376I was received before message DSNP014I, the lock request failed because of a timeout condition. Otherwise, refer to the IRLM return and reason codes for an explanation of the failure.

System action: The current SQL or the unit of recovery that issued the request might be terminated.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the IRLM return and reason code.

Problem determination: Provide the console sheet and a dump. A GTF trace might also be required. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. Refer to the explanation section of this message for additional problem determination information.

Related concepts:

⏏ IRLM return and reason codes (IRLM)

Related information:

DSNP015I *csect* **- IRLM MANUAL UNLOCK REQUEST FAILED FOR** *data-set-name*. **RC=***reason-code*,**SC=***sub-code*, **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: DB2 requested a manual unlock on a data set, but the unlock could not be performed for reasons specified in RC and SC. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: The current SQL or the unit of recovery that issued the request might be terminated.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the IRLM return and reason code.

Problem determination: Provide the console sheet and a dump. A GTF trace might also be required. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. See the explanation section of this message for additional problem determination information.

Related concepts:

⏏ IRLM return and reason codes (IRLM)

DSNP016I **CREATE FAILED FOR** *pageset-name*. **RC=***rrrrrrrr* **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id =token*

Explanation: Creation of a table space or index space failed in the data space manager (DSM) for the reason given in RC. The format of the *pageset-name* value is *dbname.tiname.instance*, where *dbname* is the database name and *tiname* is the table space or index space name.

This message identifies the function that was in progress at the time of a failure. The specific failure is usually identified by a preceding message. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: DSM terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by reason code and preceding messages, and correct it, if possible.

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP017I **RESET FAILED FOR** *pageset-name*. **RC=**reason-code **CONNECTION-ID=**connection-id, **CORRELATION-ID=**correlation-id, **LUW-ID=**logical-unit-of-work-id =token

Explanation: Reset failed in the data space manager (DSM) subcomponent of DB2 for a table space or index space for the reason that is indicated by the RC=*reason-code* value.) The format of the *pageset-name* value is *dbname.tiname*, where *dbname* is the database name and *tiname* is the table space or index space name.

This message identifies the function that was in progress at the time of a failure. The specific failure is usually identified by a preceding message. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: DSM terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the reason code and preceding messages, and correct it, if possible.

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related information:
"DSNP001I" on page 389

DSNP018I **DROP FAILED FOR** *pageset-name*. **RC=**reason-code **CONNECTION-ID=**connection-id, **CORRELATION-ID=**correlation-id, **LUW-ID=**logical-unit-of-work-id =token

Explanation: Drop failed in the data space manager (DSM) subcomponent of DB2 for a table space or index space for the reason given in RC. The format of the *pageset-name* value is *dbname.tiname*, where *dbname* is the database name and *tiname* is the table space or index space name. This message identifies the function that was in progress at the time of a failure. The specific failure is usually identified by a preceding message. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: DSM terminates its actions associated with the current SQL request. Higher level functions might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: Investigate the problem indicated by the reason code and preceding messages, and correct it, if possible.

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related information:

"DSNP001I" on page 389

DSNP019I   *csect-name* **- REALLOCATE FAILED FOR** *data-set-name*. **RC=***reason-code* **CONNECTION-ID=***connection-id*, **CORRELATION-ID=***correlation-id*, **LUW-ID=***logical-unit-of-work-id =token*

Explanation: After a successful ALTER REMOVE VOLUMES request, the data space manager (DSM) subcomponent of DB2 was unable to reallocate the data set for the reason indicated by the RC=*reason-code* value. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: Operation continues.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console.

System programmer response: The data set and corresponding page set piece or partition can usually continue to be used by the DB2 subsystem. However, they should be stopped and restarted at the earliest possible time in order to synchronize the allocation control blocks with the integrated catalog facility (ICF) catalog volume list.

Problem determination: See Problem Determination for message DSNP001I for this information.

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related tasks:

▶ Adding or removing volumes from a DB2 storage group (DB2 Administration Guide)

Related information:

"DSNP001I" on page 389

DSNP020I   *csect-name* **- USER DEFINED ATTRIBUTES INCORRECT FOR** *data-set-name*. **RC=***reason-code* **CONNECTION-ID=***connection-identifier*, **CORRELATION-ID=***correlation-identifier*, **LUW-ID=***logical-unit-of-work-id =token*

Explanation: The specified data set was access method services user-defined and does not have the required DB2 attributes. The connection ID, correlation ID, and LUWID values identify the user or application that encountered the problem.

The value of the LUWID is a logical unit of work identifier that is assigned to the thread, followed by an '=' sign and a token value, which can be used in place of 'logical-unit-of-work-id' in any DB2 command that accepts 'logical-unit-of-work-id' as input.

Messages having the same connection ID, correlation ID, and LUWID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: No action is taken with the named data set. An error code is returned and a dump is not taken.

Operator response: Notify the system programmer.

System programmer response: Delete the named data set and define it with the correct attributes.

Problem determination: Obtain an access method services LISTCAT for the named data set's attributes.

Severity: 8 (error)

Related tasks:

▶ Defining data sets (DB2 Administration Guide)

▶ Displaying threads by LUWID (DB2 Administration Guide)

Related reference:

▶ Data set naming conventions (DB2 Administration Guide)

DSNP021I   *csect-name* **EXTEND NOTIFY FAILED FOR** *data-set-name*, **SC=***sssssss*, **NOTESC=***nnnnnnnn*, **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: This is an internal error. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

An LUW-ID of '*' indicates that the thread does not have an LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of the logical-unit-of-work-id in any DB2 command that accepts the logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

If message DSNT376I was received before message DSNP021I, the notify request failed because of a timeout condition.

System action: The current SQL statement or the unit of recovery that issued the request might be terminated.

Problem determination: Obtain the console sheet and a dump. A GTF trace might also be required.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNP022I   *csect-name* **EXTEND DBET RECORDING FAILED FOR** *data-set-name*, **RC=***rrrrrrrr*, **SC=***sssssss*, **DBETSC=***nnnnnnnn*, **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: This is an internal error. The connection ID, correlation ID, and logical-unit-of-work ID (LUW-ID) identify the user or application that encountered the problem.

An LUW-ID of '*' indicates that the thread does not have an LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of the logical-unit-of-work-id in any DB2 command that accepts the logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: The current SQL statement or the unit of recovery that issued the request might be terminated.

Problem determination: Obtain the console sheet and a dump. A GTF trace might also be required.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNP026I   *csect-name* **DATASET IS NOT ALLOWED TO BE DELETED** *data-set-name*, **RC=***rrrrrrrr*, **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: During clean up of the shadow data set in the final step of online REORG, an error is detected due to an attempt to delete a regular table or index space data set. The connection ID, correlation ID, and logical unit of

Chapter 11. DSNP messages   **401**

work ID (LUWID) identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

The value of the LUWID is assigned to the thread followed by an '=' sign and a token value, that can be used in place of the LUWID in any DB2 command that accepts it as input.

Messages that have the same connection ID, correlation ID, and LUWID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: No action is taken with the named data set. An error code is returned and a dump is not taken.

Operator response: Notify the system programmer.

System programmer response: Determine the failing environment. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: This is an internal error in the online REORG of utility. For more information about this type of error, see message DSNP001I.

Related reference:

↪ REORG TABLESPACE (DB2 Utilities)

↪ REORG INDEX (DB2 Utilities)

Related information:
"DSNP001I" on page 389

DSNP027I *csect-name* **RENAME FAILED OLD DATASET=***data-set-name*, **NEW DATASET=***data-set-name*, **RC=***rrrrrrrr*, **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: An attempt to alter the data set name failed. The connection ID, correlation ID, and logical unit of work ID (LUWID) identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

The value of the LUWID is assigned to the thread followed by an '=' sign and a token value, that can be used in place of the LUWID in any DB2 command that accepts it as input.

Messages that have the same connection ID, correlation ID, and LUWID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: DB2 terminates its actions associated with the current SQL request. DB2 might terminate the current SQL request or the unit of recovery that issued the request.

Operator response: This message is directed to the database administrator or the system administrator. Bring it to that person's attention if it appears on the MVS operator's console. The installation should consider directing DB2 messages to a console other than the operator's console.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: See 'Problem Determination' for message DSNP001I for this information.

Severity: 8 (error)

DSNP028I *csect-name* **HSM RECALL FAILED FOR DSNAME = ***dsn* **HSMRC=***rrrrrrrr*,**HSMRSN=***ssssssss* **CONNECTION ID=***xxxxxxxx*, **CORRELATION ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: This message indicates that a required data set was migrated by HSM and an attempt to recall the data set failed, preventing DB2 from accessing the data set. The HSMRC and HSMRSN fields can be used to determine the HSM failure. See Problem Determination below. The connection ID, correlation ID, and 'logical-unit-of-work ID' (LUW-ID) identify the user or application that encountered the problem.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a 'logical-unit-of-work-id' that is assigned to the thread followed by an '=' sign and a token value, which can be used in

place of 'logical-unit-of-work-id' in any DB2 command that accepts 'logical-unit-of-work-id' as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

System action: DSM terminates its action with the associated data set. Higher level functions that issued the recall request might terminate.

Operator response: Notify the system programmer.

System programmer response: Investigate the reason for the HSM recall failure. Use the information in this message to help resolve this problem. Additional messages on the MVS operator's console might be associated with this message.

Problem determination: If the HSMRSN is 999, refer to the following list for the explanation:

HSMRC
> **Description**

4   HSM is not active.

8   No storage is available for MWE in CSA.

36   Input MWE is not within user storage.

40   Allocated amount of CSA for MWEs was exceeded.

100   There was an invalid function used.

For all other reason codes (HSMRSN), refer to the *z/OS DFSMShsm Managing Your Own Data* for an explanation of HSM user messages. The HSM user messages can be determined by appending the value in the HSMRC field of this message to ARC1100I (that is, HSMRC=12, HSM message is ARC1112I). The HSMRSN reason code is referenced in the ARC1100I messages.

DSNP030I   *csect-name* **THE CURRENT EXTENT REACHES THE LAST VOLUME FOR** *data-set-name*. **CONNECTION-ID=***xxxxxxxx*, **CORRELATION-ID=***yyyyyyyyyyyy*, **LUW-ID=***logical-unit-of-work-id=token*

Explanation: This message indicates that the last volume to which the data set *data-set-name* can extend has been reached. You cannot add more volumes if the current volume runs out of space.

The connection, correlation, and logical-unit-of-work IDs identify the user or application that encountered the problem. Correlation IDs beginning with '0nn', where 'nn' is a number, identify system agents.

A LUW-ID of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a *logical-unit-of-work-id* that is assigned to the thread followed by an '=' sign and a *token* value, which can be used in place of *logical-unit-of-work-id* in any DB2 command that accepts logical-unit-of-work-id as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. This makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: If sufficient space in current volume is available for the current request, DB2 continues. Otherwise, message DSNP007I is issued, and the current SQL request fails.

Operator response: For more information, issue an MVS WRITELOG operator command (for example, 'W A') to obtain MVS SYSLOG output. The SYSLOG sometimes contains VSAM or media manager messages that might be helpful in determining why space could not be allocated.

System programmer response: Depending on what information is available in the access method services or VSAM messages preceding this one, try one of the following:
- If you do not anticipate further extension of this table space or index, ignore the message.
- If the table space or index is backed by SMS-managed data sets, use LISTCAT to choose another SMS storage group that has more space available and recover or reorganize the table space or index into the new SMS storage group.
- If the table space or index is DB2-managed but is not SMS-managed, rearrange the DB2 storage group volume list by shifting volumes with free space or by adding new volumes. Use the SQL statements ALTER STOGROUP ADD VOLUMES and ALTER STOGROUP REMOVE VOLUMES and then recover or reorganize the table space or index.

DSNP031I

- If the table space or index is user-managed and not SMS-managed, delete and redefine the data set with more space and then recover or reorganize the table space or index.

Problem determination: Obtain an access method services LISTCAT for the named data set's VOLUME list for current extent space distribution of volumes.

See the Problem Determination section of message DSNP001I for more information.

Severity: 0 (informational)

DSNP031I   *csect-name* **- CURRENT SPACE HAS** *nn* **EXTENTS FOR** *data-set-name*. **THE MAXIMUM EXTENTS IS** *mm*. **CONNECTION-ID=** *connection-identifier*, **CORRELATION-ID=**-*correlation-identifier*, **LUW-ID=** *logical-unit-of-work-id=token*

Explanation: This message indicates that data set *data-set-name* has almost reached the maximum number of extents. The maximum number is indicated by *mm*. You cannot add more volumes if the current volume runs out of space.

The connection, correlation, and logical-unit-of-work IDs identify the user or application that encountered the problem. Correlation IDs beginning with *0nn* (where *nn* is a number) identifies system agents.

A LUW-ID of an asterisk (*) indicates that the thread does not have a LUW-ID. If the LUW-ID is not an asterisk (*), the value is a *logical-unit-of-work-id* that is assigned to the thread followed by an equals sign (=) and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts *logical-unit-of-work-id* as input.

Messages that have the same connection ID, correlation ID, and LUW-ID relate to the same user. Including all three pieces of information makes it possible to scan a number of DSNP-prefixed messages and group activities involving a given user.

System action: DB2 continues. If the table space or index space is not able to extend when needed, Message DSNP001I is issued and the current SQL request fails.

Operator response: For more information, issue an MVS WRITELOG operator command (for example, 'W A') to obtain MVS SYSLOG output. The SYSLOG sometimes contains VSAM or media manager messages that might be helpful in determining why space could not be allocated.

System programmer response: Based on the information that might be available in the access method services or VSAM messages preceding this one, try one of the following actions:

- If you do not anticipate further extension of this table space or index, ignore the message.
- If the table space or index is DB2-managed and has not yet reached the maximum number of extents, obtain an access method services LISTCAT for the named data set's to see how the extensions are distributed. Your options then include:
  - If there are multiple small extents, you can use the SQL statement ALTER TABLESPACE SECQTY to alter to a larger secondary space allocation quantity.
  - Otherwise, use the ALTER STOGROUP statement to add volumes to the storage group (up to 133 of them).
- If the table space or index is user-managed and has not yet reached the maximum number of extents, issue the access method services command ALTER ADDVOLUMES to provide additional volumes (up to 59 of them).
- If the data set is at the maximum number of extents, recover or reorganize the table space or index after taking one of the following actions:
  - Enlarge the primary or secondary space allocation quantity of the table space or index.
  - If the DB2 subsystem is running on z/OS V1.7 or later, and the data set is not shared with any z/OS systems at an earlier level, convert the data set to SMS-managed with the Extent Constraint Removal option set to YES in the SMS data class. If you do this, the maximum number of extents is 7257.

Problem determination: Obtain an access method services LISTCAT for the named data set's VOLUME list for the extent space distribution.

See the Problem Determination section of message DSNP001I for more information.

Severity: 0 (informational)

Related concepts:

Primary space allocation (DB2 Administration Guide)

Example of primary and secondary space allocation (DB2 Administration Guide)

Related tasks:

Extending user-managed data sets (DB2 Administration Guide)

Extending a data set (DB2 Administration Guide)

Letting SMS manage your DB2 storage groups (DB2 Administration Guide)

Related reference:

ALTER TABLESPACE (DB2 SQL)

ALTER STOGROUP (DB2 SQL)

Related information:

"DSNP001I" on page 389

Chapter 12. DSNQ messages

Message identifiers that begin with DSNQ communicate information about MQListener.

DSNQ001I   **MQListener was invoked with no major function on the command line.**

Explanation: The MQListener command line failed to specify a major function, such as help, run, add, remove, or show.

User response: Consult MQListener documentation and re-run MQListener with the function that was intended. For more information, run:

* db2mqln1/db2mqln2 help

Severity: 8 (Error)

DSNQ002I   **MQListener was invoked with an unknown command line parameter** *parameter-name***.**

Explanation: The MQListener command line contained an unknown parameter *parameter-name*.

User response: Consult MQListener documentation and rerun MQListener with the correct parameters. For more information, run

* db2mqln1/db2mqln2 help

Severity: 8 (Error)

DSNQ003I   **MQListener was invoked without the required command line parameter** *parameter name***.**

Explanation: The MQListener command line was missing the parameter *parameter name*, which was required.

User response: Consult MQListener documentation and re-run MQListener with the parameter that was required. For more information, run

* db2mqln1/db2mqln2 help

Severity: 8 (Error)

DSNQ004I   **MQListener was invoked with an illegal value for the** *parameter name* **command line parameter:** *parameter value***.**

Explanation: The MQListener command line contained an illegal value for the *parameter name* parameter. The illegal value was *parameter value*.

User response: Consult MQListener documentation and re-run MQListener with the correct parameters. For more information, run

* db2mqln1/db2mqln2 help

Severity: 8 (Error)

DSNQ010I   **MQListener encountered a database error accessing subsystem** *subsystem name* **during operation** *operation name***: sqlstate =** *sqlstate value* **(sqlcode =** *sqlcode value***).**

Explanation: MQListener encountered a DB2 database error accessing subsystem *subsystem name* while executing operation *operation name*. The resulting sqlstate was *sqlstate value* (sqlcode *sqlcode value*).

User response: Consult MQListener documentation and ensure that MQListener is properly installed in the specified DB2 subsystem and access to MQListener packages and configuration tables ahs been granted to the user running MQListener.

Severity: 8 (Error)

DSNQ011I **MQListener failed to make RRS connection to the subsystem** *subsystem name* **during operation** *operation name*: **reason code** = *reason code value* (**return code** = *return code value*).

Explanation: MQListener encountered a DB2 database error while attempting to connect to subsystem *subsystem name* during operation *operation name*. The resulting reason code was *reason code value* (return code *return code value*).

User response: Ensure that the DB2 subsystem *subsystem name* is active and running.

Severity: 8 (Error)

DSNQ012I **MQListener failed to terminate RRS connection from the subsystem** *subsystem name* **during operation** *operation name*: **reason code** = *reason code value* (**return code** = *return code value*).

Explanation: MQListener encountered a DB2 database error while attempting to disconnect from subsystem *subsystem name* during operation *operation name*. The resulting reason code was *reason code value* (return code *return code value*).

User response: Ensure that the DB2 subsystem *subsystem name* is active and running. Consult the DB2 documentation, regarding the exact meaning of the reason and return code.

Severity: 4 (Warning)

DSNQ020I **MQListener encountered a MQ error error executing operation** *operation name* **on object** *object name* **in queue manager** *queue manager name*: **reason code** = *reason code value*.

Explanation: MQListener encountered a MQ error error while executing operation *operation name* on object *object name* in queue manager *queue manager name*. The resulting reason code was *reason code value*.

User response: Ensure that MQSeries® is properly installed and configured, and accessible to the user running MQListener. Consult the MQSeries documentation for more information, particularly the Application Programming Reference for more information about the reason code.

Severity: 8 (Error)

DSNQ021I **MQListener failed to connect to MQ queue manager** *queue manager name*: **reason code** = *reason code value*.

Explanation: MQListener encountered a MQ error error while attempting to connect to MQ queue manager *queue manager name*. The resulting reason code was *reason code value*.

User response: Ensure that MQSeries is properly installed and configured, and accessible to the user running MQListener. Consult the MQSeries documentation for more information, particularly the Application Programming Reference for more information about the reason code.

Severity: 8 (Error)

DSNQ022I **MQListener failed to disconnect from MQ queue manager** *queue manager name*: **reason code** = *reason code value*.

Explanation: MQListener encountered a MQ error error while attempting to disconnect from MQ queue manager *queue manager name*. The resulting reason code was *reason code value*.

User response: Ensure that MQSeries is properly installed and running. Consult the MQSeries documentation for more information, particularly the Application Programming Reference for more information about the reason code.

Severity: 4 (Warning)

DSNQ030I **MQListener encountered a DB2 database error attempting to add a new task to the MQListener configuration** *configuration name* **in the subsystem** *subsystem name*. **The input queue specified in the task was** *input queue name* **and the specified queue manager was** *queue manager name*. **The resulting sqlstate was** *sqlstate value* (**sqlcode** = *sqlcode value*).

Explanation: MQListener encountered a DB2 database error while attempting to add a new task to the MQListener configuration *configuration name* in the subsystem *subsystem name*. The input queue specified in the task was *input*

queue name and the specified queue manager was *queue manager name*. The resulting sqlstate was *sqlstate value* (sqlcode = *sqlcode value*).

User response: Consult MQListener documentation and ensure that MQListener is properly installed in the specified subsystemand and access to MQListener packages and configuration tables has been granted to the user running MQListener. Ensure that the combination of configuration name, input queue, and queue manager is unique among the MQListener configuration tasks in the subsystem. Use MQListener "show" command to inspect the configuration.

Severity: 8 (Error)

DSNQ040I   **MQListener encountered a DB2 database error retrieving configuration** *configuration name* **from subsystem** *subsystem name* **during operation** *operation name*. **The resulting sqlstate was** *sqlstate value* **(sqlcode =** *sqlcode value* **).**

Explanation: MQListener encountered a DB2 database error while retrieving configuration *configuration name* from subsystem *subsystem name* during operation *operation name*. The resulting sqlstate was *sqlstate value* (sqlcode= *sqlcode value*).

User response: Consult MQListener documentation and ensure that MQListener is properly installed in the specified subsystem and access to MQListener packages and configuration tables has been granted to the user running MQListener.

Severity: 8 (Error)

DSNQ060I   **MQListener encountered a DB2 database error while removing a task from configuration** *configuration name* **in subsystem** *subsystem name*. **The input queue specified was** *input queue name* **and the queue manager specified was** *queue manager name*. **The resulting sqlstate was** *sqlstate value* **(sqlcode =** *sqlcode value* **).**

Explanation: MQListener encountered a DB2 database error while removing a task from configuration *configuration name* in subsystem *subsystem name*. The input queue that was specified was *input queue name*, and the queue manager that was specified was *queue manager name*. The resulting sqlstate was *sqlstate value* (sqlcode = *sqlcode value*).

User response: Consult MQListener documentation and ensure that MQListener is properly installed in the specified subsystem. Also, ensure that access to MQListener packages and configuration tables has been granted to the user who is running MQListener.

Severity: 8 (Error)

DSNQ061I   **MQListener attempted to remove an unknown task from configuration** *configuration name* **in subsystem** *subsystem name*. **The input queue specified was** *input queue name* **and the queue manager specified was** *queue manager name*.

Explanation: MQListener attempted to remove an unknown task from configuration *configuration name* in subsystem *subsystem name*. The input queue that was specified was *input queue name* and the queue manager that was specified was *queue manager name*.

User response: Ensure that specified parameters are correct, and re-run MQListener. Use MQListener "show" command to inspect the configuration.

Severity: 0 (Informational)

DSNQ070I   **MQListener has encountered a DB2 database error preparing to run the stored procedure** *schema name*. *procedure name* **in subsystem** *subsystem name*: **sqlstate =** *sqlstate value* **(sqlcode =** *sqlcode value* **).**

Explanation: MQListener has encountered a DB2 database error while preparing to run the stored procedure *schema name*. *procedure name* in subsystem *subsystem name*. The resulting sqlstate was *sqlstate value* (sqlcode = *sqlcode value*).

User response: Ensure that MQListener configuration is correctly specified and that a stored procedure of the specified schema and name exists with a correct signature and is accessible to the user running MQListener.

Severity: 8 (Error)

DSNQ071I   **MQListener encountered an unsupported datatype in the signature of the stored procedure** *schema name*. *procedure name* **in the subsystem** *subsystem name*: **datatype =** *datatype value*.

Explanation: MQListener encountered an unsupported datatype in the signature of the stored procedure *schema name*. *procedure name* in the subsystem *subsystem name*. The datatype was *datatype value*.

User response: Consult the MQListener documentation and ensure that the stored procedure has the correct datatype.

Severity: 8 (Error)

DSNQ072I   **MQListener was unable to start a thread running a task specified in the configuration** *configuration name* **by the input queue** *input queue name* **and queue manager** *queue manager name*. **The ECF error code was** *error code value*.

Explanation: MQListener was unable to start a thread running a task that was specified in the configuration *configuration name* by the input queue *input queue name* and queue manager *queue manager name*. The ECF error code was *error code value*.

User response: Ensure that MQListener configuration is correct, especially the -numInstances parameter, and that the MQListener process has sufficient system resources to execute the required number of simultaneous tasks.

Severity: 8 (Error)

DSNQ073I   **MQListener started a thread running the task specified in the configuration** *configuration name* **by the input queue** *input queue name* **and queue manager** *queue manager name*.

Explanation: MQListener started a thread running the task that was specified in the configuration *configuration name* by the input queue *input queue name* and queue manager *queue manager name*.

User response: None. This is the formal notification of a thread start, expected during startup or restart.

Severity: 0 (Informational)

DSNQ074I   **A thread running the task specified in the configuration** *configuration name* **by the input queue** *input queue name* **and queue manager** *queue manager name* **has exited.**

Explanation: A thread running the task specified in the configuration *configuration name* by the input queue *input queue name* and queue manager *queue manager name* has exited.

User response: None. This is the formal notification of a thread exit, expected during shutdown or restart.

Severity: 0 (Informational)

DSNQ075I   **MQListener encountered a DB2 database error while running the stored procedure** *schema name*. *procedure name* **in subsystem** *subsystem name* **during operation** *operation name*. **The resulting sqlstate was** *sqlstate value* **(** *sqlcode =* *sqlcode value* **).**

Explanation: MQListener encountered a DB2 database error while running the stored procedure *schema name*. *procedure name* in subsystem *subsystem name* during operation *operation name*. The resulting sqlstate was *sqlstate value* (sqlcode = *sqlcode value*).

User response: Consult MQListener documentation and ensure that MQListener is properly installed in the specified subsystem and access to MQListener packages and configuration tables has been granted to the user running MQListener. Ensure that the stored procedure executes correctly.

Severity: 8 (Error)

DSNQ080I   **MQListener received an unknown message on admin queue** *admin queue name* **in queue manager** *queue manager name*: *message*.

Explanation: MQListener received an unknown message on admin queue *admin queue name* in queue manager *queue manager name*: *message*.

User response: Ensure that MQSeries is correctly installed and configured, that MQListener is being run with the proper -adminQMgr and -adminQueue parameter values, and that the MQListener "admin" command is also using

the proper -adminQMgr and -adminQueue or -adminQueueList parameter values. If used, check that the -adminQueueList specifies the proper queue names. Check that the admin queue is not being used by another application.

Severity: 4 (Warning)

DSNQ081I **MQListener received a shutdown message on admin queue** *admin queue name*, **queue manager** *queue manager name*.

Explanation: MQListener received a shutdown message on admin queue *admin queue name*, queue manager *queue manager name*.

User response: None. This is a formal notification on receiving a shutdown message.

Severity: 0 (Informational)

DSNQ082I **MQListener received a restart message on admin queue** *admin queue name*, **queue manager** *queue manager name*.

Explanation: MQListener received a restart message on admin queue *admin queue name*, queue manager *queue manager name*.

User response: None. This is a formal notification on receiving a restart message.

Severity: 0 (Informational)

Chapter 13. DSNR messages

Message identifiers that begin with DSNR communicate information about the recovery manager and indicate the progress of DB2 through a restart process.

DSNR001I   RESTART INITIATED

Explanation: This message delimits the beginning of the restart process within startup. The phases of restart are about to begin. These phases are necessary to restore the operational environment to that which existed at the time of the previous termination and to perform any recovery actions that may be necessary to return DB2-managed resources to a consistent state.

DSNR002I   RESTART COMPLETED

Explanation: This message delimits the completion of the restart process within startup.

DSNR003I   RESTART...PRIOR CHECKPOINT RBA=*xxxxxxxxxxxx*

Explanation: The message indicates the first phase of the restart process is in progress and identifies (as 'xxxxxxxxxxxx') the log positioning RBA of the checkpoint from which the restart process will obtain its initial recovery information.

DSNR004I   RESTART...UR STATUS COUNTS IN COMMIT=*nnnn***, INDOUBT=***nnnn***, INFLIGHT=***nnnn***, IN ABORT=***nnnn***, POSTPONED ABORT=***nnnn*

Explanation: This message indicates the completion of the first phase of the restart process. The counts indicate the number of units of work whose execution state during a previous DB2 termination was such that (to ensure DB2 resource consistency) some recovery action must be performed during this restart process or (by the user) upon completion of the restart process. The counts of IN COMMIT, INDOUBT, INFLIGHT and IN ABORT may provide an indication of the time required to perform the remaining two phases of restart (forward and backward recovery). The counts of POSTPONED ABORT URs may provide an indication of the time required to perform -RECOVER POSTPONED processing.

The IN COMMIT count specifies the number that had started but not completed phase 2 of the commit process. These must undergo forward recovery to complete the commit process.

The INDOUBT count specifies the number that were interrupted between phase 1 and phase 2 of the commit process. These must undergo forward recovery to ensure resources modified by them are unavailable until their indoubt status is resolved.

The INFLIGHT count specifies the number that neither completed phase 1 of the commit process nor began the process of backing out. These must undergo backward recovery to restore resources modified by them to their previous consistent state.

The IN ABORT count specifies the number that were in the process of backing out. These must undergo backward recovery to restore resources modified by them to their previous consistent state.

The POSTPONED ABORT count specifies the number whose backout was postponed in an earlier restart process and not attended to prior to this restart process. These must undergo backward recovery after completion of restart, via the -RECOVER POSTPONED command, to restore resources modified by them to their previous consistent state.

DSNR005I   RESTART...COUNTS AFTER FORWARD RECOVERY IN COMMIT=*nnnn***, INDOUBT=***nnnn*

Explanation: The message indicates the completion of the forward recovery restart phase. The counts indicate the number of units of work whose recovery actions could not be completed during the phase. Typically, those in an IN COMMIT state remain because the recovery actions of some subcomponents have not been completed. Those units of work in an INDOUBT state will remain until connection is made with the subsystem that acts as their commit coordinator.

Operator response: No action is required unless the conditions persist beyond some installation-defined period of

time. Recovery action will be initiated when the resource is brought online. Indoubt resolution will be initiated as part of the process of reconnecting the subsystems.

DSNR006I **RESTART...COUNTS AFTER BACKWARD RECOVERY INFLIGHT=***nnnn***, IN ABORT=***nnnn***, POSTPONED ABORT=***nnnn*

Explanation: The message indicates the completion of the backward recovery restart phase. The counts indicate the number of units of work whose recovery actions could not be completed during the phase.

When INFLIGHT or IN ABORT URs do not complete backout during restart, they are converted to POSTPONED ABORT status, so the INFLIGHT and IN ABORT counts should be zero. The POSTPONED ABORT count indicates the number of URs for which backout work still exists (from this or a previous restart).

Backout work for the POSTPONED ABORT URs should be completed by the user (by using the RECOVER POSTPONED command) after DB2 restart has completed.

Operator response: No action is required unless the condition persists beyond some installation-defined period of time. Recovery action will be initiated when the resource collection is brought online.

DSNR007I **RESTART...STATUS TABLE**

Explanation: This message precedes a table showing the status of units of recovery (URs) after each restart phase. The message and the table will accompany the DSNR004I, DSNR005I, or DSNR006I message after each nested phase. At the end of the current status rebuild phase, it shows the status of any URs that require processing. At the end of the forward recovery and backoutphases, it shows the status of only those URs which needed processing but were not processed. The table helps to identify the URs that were active when DB2 came down, and to determine the log scope required to bring DB2 up.

The following list describes each column of the table:

T Connection type. The values could be:

    B Batch: a batch program or QMF user.

    I Internal: a system service being performed on some user's behalf.

    S Subsystem: work that originated from an attached subsystem, either CICS or IMS/VS.

CONID
    Connection identifier for related URs. Batch connections are not related to any other connection. Subsystem connections with the same identifier indicate URs that originated from the same subsystem.

CORRID
    Correlation identifier; unique within a subsystem connection, it relates a UR to a particular CICS or IMS/VS transaction.

AUTHORIZATION ID
    Authorization identifier of the user for whom this UR was created.

PLAN Name of the plan allocated to the UR.

S Restart status of the UR. When DB2 came down, the UR was in one of these situations:

    A INABORT: The UR was in the 'must-complete' phase of abort, and is yet to be completed.

    C INCOMMIT: The UR was in the 'must-complete' phase of commit, and is yet to be completed.

    D INDOUBT: The UR had completed the first phase of commit, but DB2 had not received the second phase instruction. The UR must be remembered so it can be resolved when the owning subsystem reattaches.

    F INFLIGHT: The UR had not completed the first phase of commit, and will be backed out.

    P POSTPONED ABORT: The UR has not yet completed backout. Backout must be completed outside of DB2 restart by means of the -RECOVER POSTPONED command.

URID UR identifier, the log RBA of the beginning of this unit of work. It is the earliest RBA required to process the UR during Restart.

DAY The day the UR was created, in the format 'ddd'. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

TIME The time the UR was created, in the format 'hh mm ss '. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

DSNR009I STORAGE IS NOT AVAILABLE FOR THE UR DISPLAY TABLE, THE SIZE REQUESTED = *xxxx*, **STORAGE MANAGER REASON CODE =** *yyyyyyyy*

Explanation: There was not enough storage available during the creation of the recoverable UR (RURE) display table (DSNDRUDT).

System action: The system continues to restart but the RESTART... STATUS TABLE is not displayed.

Operator response: Save the console output and inform the system programmer.

System programmer response: Increase the region size of the *ssnm*MSTR address space before restarting DB2 again.

Problem determination: The size requested is approximately 110 bytes for each unit of recovery (UR). See the message DSNR004I to determine the total number of URs to process. Then get the 00E2xxxx reason code from the DSNR009I header. Use these two numbers to determine the reason for the shortage.

DSNR010I AN ERROR OCCURRED IN THE RURE STATUS TABLE SORT/TRANSLATE MODULE (DSNRUSOR), THE ERROR LOCATION CODE = *xxxx*

Explanation: There was an error in processing the RURE display table. One of these ERROR LOCATION CODES tells what function the module was attempting to perform:

X'0001' Trace entry to the module

X'0002' Sort the recoverable UR display table

X'0003' Translate the recoverable UR table

X'0004' Convert the store clock value for display

X'0005' Trace exit from the module

System action: The system continues to restart but the RESTART... STATUS TABLE is not displayed.

Operator response: Save the console output and inform the system programmer.

System programmer response: Use the error code in the message to complete the problem description.

Problem determination: The ERROR CODE tells the problem area and should be used in the problem description. A record was added to the SYS1.LOGREC that contains the PSW and module at the time of the error. This error indicates an application program check of some type.

DSNR011I AN ERROR OCCURRED IN THE RURE STATUS TABLE DISPLAY MODULE (DSNRUDIS), THE ERROR LOCATION CODE = *xxxx*

Explanation: There was an error in processing the RURE display table. One of these ERROR LOCATION CODES tells what function the module was attempting to perform:

X'0001' Trace entry to the module

X'0002' Build header for the display table

X'0003' Build entries for the display table

X'0004' Call DSNRWTO to display the table

X'0005' Trace exit from the module

X'0006' Get message text

System action: The system continues to restart but the RESTART... STATUS TABLE is not displayed.

Operator response: Save the console output and inform the system programmer.

System programmer response: Use the error code in the message to complete the problem description.

Problem determination: The ERROR CODE describes the problem area and should be used in the problem description. A record was added to the SYS1.LOGREC containing the value DSNRUDIS, and the PSW and CSECT at the time of the error. This error indicates an application program check of some type.

Error codes other than X'0006' indicate an application program check of some type. Code X'0006' indicates an error attempting to locate the table text. The trace entry for DSNRUDIS contains the return and reason codes from the Get Text request. That information should be used to complete the problem description.

DSNR014I   EXCLUDED RURE TABLE

Explanation: This message contains a list of RUREs which are being bypassed because of the conditional restart. Either they were to be processed in a phase that was bypassed or they were outside the log range.

The following list describes each column of the table:

T　　　Connection type. The values could be:

　　B　　　Batch: a batch program or QMF user.

　　I　　　Internal: a system service being performed on some user's behalf.

　　S　　　Subsystem: work that originated from an attached subsystem, either CICS or IMS/VS.

CONID
　　　Connection identifier for related URs. Batch connections are not related to any other connection. Subsystem connections with the same identifier indicate URs that originated from the same subsystem.

CORRID
　　　Correlation identifier; unique within a subsystem connection, it relates a UR to a particular CICS or IMS/VS transaction.

AUTHORIZATION ID
　　　Authorization identifier of the user for whom this UR was created.

PLAN　　Name of the plan allocated to the UR.

S　　　Restart status of the UR. When DB2 came down, the UR was in one of these situations:

　　A　　　INABORT: The UR was in the 'must-complete' phase of abort, and is yet to be completed.

　　C　　　INCOMMIT: The UR was in the 'must-complete' phase of commit, and is yet to be completed.

　　D　　　INDOUBT: The UR had completed the first phase of commit, but DB2 had not received the second phase instruction. The UR must be remembered so it can be resolved when the owning subsystem reattaches.

　　F　　　INFLIGHT: The UR had not completed the first phase of commit, and will be backed out.

　　P　　　POSTPONED ABORT: The UR has not yet completed backout. Backout must be completed outside of DB2 restart by means of the -RECOVER POSTPONED command.

URID　　UR identifier, the log RBA of the beginning of this unit of work. It is the earliest RBA required to process the UR during Restart.

DAY　　The day the UR was created, in the format 'ddd'. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

TIME　　The time the UR was created, in the format 'hh mm ss '. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

System action: Restart continues.

DSNR015I   THE CHKPTRBA *xxxxxxxxxxx* **SPECIFIED IN THE CRCR IS INVALID**

Explanation: The checkpoint RBA specified in the conditional restart control record (CRCR) is either not available or is not a begin checkpoint record.

System action: Restart is abended with reason code '00D99001'.

Operator response: Inform the system programmer.

System programmer response: By specifying the STARTRBA and/or ENDRBA on the CRESTART control statement, change log inventory (DSNJU003) utility selects an appropriate RBA.

Problem determination: Print the complete checkpoint records queue by using the print log map (DSNJU004) utility. Select a checkpoint RBA from this queue or let the change log inventory (DSNJU003) utility make the selection.

DSNR016I   **STARTRBA** *xxxxxxxxxxxx* **VALIDATION FAILED**

Explanation: A scan from the STARTRBA failed to find a complete log record. Either the entire log was scanned and no complete record was found, or the log manager found an invalid log record. The log manager sends the error messages to the console indicating a bad log record.

System action: Restart abends.

Operator response: Record the log manager message numbers and inform the system programmer.

System programmer response: Use DSN1LOGP to determine a STARTRBA to use for restart. The recovery manager uses the STARTRBA as a starting point for its search of the beginning of a valid log record.

DSNR017I   **RESTART IS BEING TERMINATED BECAUSE OF CSRONLY SPECIFICATION IN CRCR**

Explanation: The restart procedure only completed the first phase of startup. The status displayed during this partial restart can be used to build the conditional restart control record (CRCR).

System action: Restart is abended.

System programmer response: To prevent the restart control record from being reused during the next restart attempt, the current restart control record must be canceled. Use the change log inventory (DSNJU003) CRESTART CANCEL option, or create a new restart control record. The newly created restart control record will automatically cancel the previous record.

DSNR018I   *csect-name* **RESTART...BACKWARD RECOVERY PROCESSED FROM RBA** *ffffffffffff* **TO RBA** *tttttttttttt*

Explanation: This message indicates the range of log processed during the backward phase of RESTART when postponed backout processing has been requested. Processing began with the LR of RBA *ffffffffffff* and proceeded backwards to the LR of RBA *tttttttttttt*.

System action: This message is followed by messages DSNR006I and DSNR007I. The former tells how many URs had some backout work postponed. The latter, identifies the postponed abort URs. The amount of log that need to be processed to complete backout processing for the postponed abort URs can be gauged by comparing the RBA reached during RESTART's backward processing (RBA *tttttttttttt*), and the URIDs of the postponed abort URs.

DSNR020I   *csect-name* **START MEMBER** *member***, OR REPLY 'NO' OR 'QUIESCED'**

Explanation: During a group restart, the group member issuing this message attempted to access the BSDS for member *member*, and either was not able to access the BSDS, or discovered that the BSDS was modified by the change log inventory (DSNJU003) utility since that member was last started. In either case, this DB2 is not able to perform peer restart for member *member*.

For group restart to continue, the identified member must be started.

If it is not possible to start the other member and you want to terminate this group restart, you can reply 'NO' to this message. Replying 'NO' to this message causes this DB2 to terminate with a reason code of 00D900E1. You then have to reply 'NO' to this message for all remaining members participating in this group restart.

If you are sure that the identified member was quiesced the last time it was started, that is, it was stopped with a -STOP DB2 MODE(QUIESCE) command and it stopped with no indoubt units of recovery and no distributed resynchronization responsibility, then you can reply 'QUIESCED' to this message. Replying 'QUIESCED' to this message causes message DSNR030I to be issued as a reminder that group restart will continue **without using the log** from the identified member.

System action: This group member waits for the identified member to be started or for a valid reply to this message.

- If the identified member is started, group restart continues.
- If the reply is 'NO', this DB2 terminates with a reason code of 00D900E1.

• If the reply is 'QUIESCED', group restart continues **without using the log** from the identified member.

DSNR021I *csect-name* **DB2 SUBSYSTEM MUST PERFORM GROUP RESTART FOR PEER MEMBERS**

Explanation: This DB2 subsystem must perform the group restart phase to recover either the shared communications area (SCA) or the retained locks for nonstarting members in the data sharing group.

This message marks the beginning of group restart processing for peer members. Messages DSNR023I and DSNR024I or messages DSNR025I and DSNR026I are issued for each peer that is processed by this DB2 subsystem. Message DSNR022I marks the end of group restart processing for peer members.

This message is issued before this DB2 member enters the group restart process. The group restart process is synchronized among all group members that are starting and therefore participating in this group restart. After this DB2 member synchronizes with other members, it is possible that no peer processing will be necessary; in that case message DSNR022I immediately follows this message.

If it is necessary to recover both the SCA and retained locks, this message is issued twice; once for SCA recovery and again for the recovery of the retained locks.

System action: The subsystem begins group restart processing.

DSNR022I *csect-name* **DB2 SUBSYSTEM HAS COMPLETED GROUP RESTART FOR PEER MEMBERS**

Explanation: This DB2 subsystem, which previously issued message DSNR021I, completed the group restart phase to recover the shared communications area (SCA) or retained locks for nonstarting DB2 members in the data sharing group.

System action: The subsystem continues with the next restart phase of its own subsystem.

DSNR023I *csect-name* **GROUP RESTART INITIATED TO RECOVER THE SCA FOR GROUP MEMBER**
membname

Explanation: The DB2 subsystem that issued this message is performing the group restart phase to recover the shared communications area (SCA) for nonstarting member *membname* of the data sharing group.

This message is followed by message DSNR024I after the group restart phase to recover the SCA for the identified member is complete.

System action: The subsystem continues the group restart process.

DSNR024I *csect-name* **GROUP RESTART COMPLETED TO RECOVER THE SCA FOR GROUP MEMBER**
membname

Explanation: The DB2 subsystem that issued this message completed the group restart phase to recover the shared communications area (SCA) for member *membname* of the data sharing group.

This message is issued only if messages DSNR021I and DSNR023I were issued.

System action: The subsystem continues the group restart process.

DSNR025I *csect-name* **GROUP RESTART INITIATED TO RECOVER RETAINED LOCKS FOR GROUP
MEMBER** *membname*

Explanation: The DB2 subsystem that issued this message is performing the group restart phase to recover the retained locks for nonstarting member *membname* of the data sharing group.

This message is followed by message DSNR026I after the group restart phase to recover the retained locks for the identified member is complete.

System action: The subsystem continues the group restart process.

User response: Restart the failed member to release the retained locks.

DSNR026I   *csect-name* **GROUP RESTART COMPLETED TO RECOVER RETAINED LOCKS FOR GROUP MEMBER** *membname*

Explanation: The DB2 subsystem that issued this message completed the group restart phase to recover the retained locks for member *membname* of the data sharing group.

This message is issued only if messages DSNR021I and DSNR025I were issued.

System action: The subsystem continues the group restart process.

DSNR030I   *csect-name* **WILL CONTINUE WITHOUT THE** *member* **MEMBER'S LOG, REPLY 'YES' OR 'NO'**

Explanation: The reply to message DSNR020I was 'QUIESCED'. This message is issued as a reminder that group restart will continue **without using the log** from the identified member.

If you are sure that the identified member was quiesced the last time it was started, that is, it was stopped with a -STOP DB2 MODE(QUIESCE) command and it stopped with no indoubt units of recovery and no distributed resynchronization responsibility, then reply 'YES'; otherwise, reply 'NO'.

System action: This group member waits for a valid reply to this message.
- If the reply is 'YES', group restart continues **without using the log** from the identified member.
- If the reply is 'NO', message DSNR020I is reissued to allow you to enter a different response.

DSNR031I   *csect-name* **PROCESSING LOG RECORD AT RBA** *rba1* **TO RBA** *rba2*

Explanation: This message is produced periodically during the forward and backward processing phases of restart. The RBA values in this message can be used to monitor the progress of the phases of restart.

Forward processing occurs between messages DSNR004I and DSNR005I. During forward processing, the log is read in a forward direction.

Backward processing occurs between messages DSNR005I and DSNR006I. During backward processing, the log is read in a backward direction and reading stops at the RBA that matches the URID of the oldest inflight or in-abort UR.

csect-name
>   The name of the control section that issued the message.

rba1   The RBA of the log record that is currently being processed.

rba2   The RBA of the last log record that is read in this phase of restart.

If *rba1* is not progressing, restart might be suspended. Messages DSNR055I through DSN058I might accompany this message and provide additional diagnostic information to help identify the cause of the suspension.

System action: DB2 restart continues.

Operator response: If *rba1* is not progressing, notify the system programmer.

System programmer response: If *rba1* is not progressing, take one of the following actions:
- If messages DSNR055I through DSNR058I indicate the reason for the suspension, follow the instructions for those messages to resolve the restart suspension.
- If messages DSNR055I through DSNR058I are not displayed or do not provide enough information, request SVC dumps of the DB2 and IRLM address spaces for problem determination. For data sharing, request dumps from all members of the data sharing group.

User response: No action is required.

Related tasks:

➡ Requesting DB2 SVC dumps (Collecting data)

Related information:

DSNR034E *csect-name* **THERE CANNOT BE MORE THAN TWO RELEASE LEVELS OF DB2 RUNNING IN THE GROUP**

Explanation: You attempted to start a DB2 that was at a release level that cannot coexist in the data sharing group. There are already two different releases of DB2 currently running in the data sharing group.

System action: DB2 terminates.

System programmer response: If you are sure that the startup procedure is pointing to the correct set of load libraries, you must stop any DB2 subsystems that cannot coexist with the one you are attempting to start.

Related concepts:

➡ Mixed releases in a data sharing group (DB2 Installation and Migration)

➡ Release coexistence in a data sharing environment (DB2 Installation and Migration)

➡ Recommendation for group restart in coexistence (DB2 Installation and Migration)

Related tasks:

➡ Migrating an existing data sharing group to Version 11 conversion mode (DB2 Installation and Migration)

➡ Preparing for DB2 data sharing (DB2 Installation and Migration)

DSNR035I *csect-name* **WARNING - UNCOMMITTED UR AFTER** *number* **CHECKPOINTS - CORRELATION NAME =** *correlation-name* **CONNECTION ID =** *connection-id* **LUWID =** *logical-unit-of-work-id=token* **PLAN NAME =** *plan-name* **AUTHID =** *authorization-id* **END USER ID =** *user-id* **TRANSACTION NAME =** *transaction-name* **WORKSTATION NAME =** *workstation-name*

Explanation: This message indicates that during checkpoint processing, DB2 encountered an uncommitted unit of recovery (UR) that has an inflight or indoubt status.

csect-name
> The name of the control section that issued the message.

number The cumulative number of checkpoints that were taken since the beginning of the UR.

correlation name
> The correlation name for the thread that is associated with the UR.

connection-id
> The connection ID for the thread that is associated with the UR.

logical-unit-of-work-id
> The logical unit of work ID for the thread that is associated with the UR. An asterisk (*) denotes that the thread originated at the local site.
>
> This value consists of a fully qualified LU network name followed by a period and an LUW instance number. The LU network name consists of a one- to eight-character network ID, a period, and a one- to eight-character network LU name. The LUW instance number consists of 12 hexadecimal characters that uniquely identify the unit of work.

token A unique token number that is associated with the logical unit of work ID.

If the UR is inflight, *token* refers to the thread in the report from the DISPLAY THREAD command with the TYPE(ACTIVE) option.

If the UR is indoubt, *token* might not refer to the thread in the report from the DISPLAY THREAD command with the TYPE(INDOUBT) option. However, in this case, you can use the logical unit of work ID to determine the correct association.

plan-name
> The plan for the thread that is associated with the UR.

authorization-id
> The authorization ID for the thread that is associated with the UR.

user-id The user ID that is associated with the UR, if the thread was created with client user information. Otherwise, an asterisk (*).

transaction-name
> The transaction that is associated with the UR, if the thread was created with client user information. Otherwise, an asterisk (*).

workstation-name
> The workstation host name that is associated with the UR, if the thread was created with client user information. Otherwise, an asterisk (*).

System action: Processing continues. If statistics class 3 is turned on, IFCID 0313 is written.

System programmer response: You can see the UR in the DISPLAY THREAD TYPE(ACTIVE) report or the DISPLAY THREAD TYPE(INDOUBT) report. Depending on which report is relevant, take one of the following actions:

If the UR is in the DISPLAY THREAD TYPE(ACTIVE) report:
> Consult with the application programmer to determine whether this UR is problematic.
>
> If the problematic UR is caused by an application program, you can use the CANCEL THREAD command to roll back the UR, if necessary. Input to the command can be the *logical-unit-of-work-id* value or the *token* value. The amount of time required for the backout process depends on the amount of work performed by the UR.
>
> For more information about problems caused by uncommitted URs, see the information about programming for concurrency.

If the UR is in the DISPLAY THREAD TYPE(INDOUBT) report:
> The UR is typically resolved automatically in a reasonable amount of time. However, in cases where the UR is not resolved, consider using the RECOVER INDOUBT command to provide a commit or abort decision. Input to the command can be the *logical-unit-of-work-id* value, the *token* value, or any other appropriate value that is provided in the DISPLAY THREAD TYPE(INDOUBT) report. The amount of time required for the backout process depends on the amount of work performed by the UR.

Programmer response: Ensure that the application commits frequently enough, or consult with your DB2 administrator about decreasing the frequency of the check for uncommitted URs. Use the URCHKTH subsystem parameter.

Related tasks:

▶ Displaying threads by LUWID (DB2 Administration Guide)

▶ Programming for concurrency (DB2 Performance)

Related reference:

▶ UR CHECK FREQ field (URCHKTH subsystem parameter) (DB2 Installation and Migration)

DSNR036I *csect-name* **UNRESOLVED UR ENCOUNTERED DURING CHECKPOINT - CORRELATION NAME = *correlation-name* CONNECTION ID = *connection-id* LUWID = *logical-unit-of-work-id* PLAN NAME = *plan-name* AUTHID = *authorization-id***

Explanation: DB2 detected an INDOUBT unit of recovery (UR) during a checkpoint. Either an INDOUBT UR or POSTPONED ABORT UR exists.

csect-name
> The name of the control section that issued the message.

correlation name
> The correlation name that is associated with the UR.

connection-id
> The connection ID that is associated with the UR.

logical-unit-of-work-id
> The logical unit of work ID. An asterisk (*) denotes that the thread originated at the local site.
>
> This value consists of a fully qualified LU network name followed by a period and an LUW instance number. The LU network name consists of a one- to eight-character network ID, a period, and a one- to eight-character network LU name. The LUW instance number consists of 12 hexadecimal characters that uniquely identify the unit of work.

plan-name
> The plan that is associated with the UR.

authorization-id
> The authorization ID that is associated with the UR.

System action: Processing continues. If statistics class 3 is turned on, IFCID 0313 is written.

System programmer response: Consult with the application programmer to determine if this UR is problematic.

Related tasks:

➡ Displaying threads by LUWID (DB2 Administration Guide)

➡ Resolving indoubt units of recovery (DB2 Administration Guide)

Related reference:

➡ UR CHECK FREQ field (URCHKTH subsystem parameter) (DB2 Installation and Migration)

DSNR037I   *csect-name* **WARNING - THE TCP/IP DRDA PORT NUMBER IS CHANGED FROM** *port-1* **TO** *port-2*

Explanation: The TCP/IP DRDA port number specified in the DB2 BSDS is different from the value used the last time a member of the DB2 data sharing group started. All members of a DB2 data sharing group must use the same TCP/IP DRDA port number.

csect-name
> The name of the control section that issued the message.

port-1   The TCP/IP DRDA port number that was obtained from the coupling facility during DB2 startup. This value is the TCP/IP DRDA port number that was used the last time a member of the data sharing group was started.

port-2   The TCP/IP DRDA port number that was obtained from the DB2 BSDS during DB2 startup.

System action: DB2 accepts the new TCP/IP DRDA port number, and DB2 startup continues.

Operator response: Contact the system programmer.

System programmer response: If the TCP/IP DRDA port number was changed inadvertently, stop DB2 immediately. Use the change log inventory utility (DSNJU003) to correct the PORT value, and restart DB2.

If this message reflects a change that you want to make to the DB2 DRDA port number, no action is required. The DRDA port number stored in the coupling facility is updated, and this new port number will be used when DB2 validates each member's DRDA port number.

DSNR038I   *csect-name* **WARNING - THE TCP/IP RESYNCHRONIZATION PORT NUMBER IS CHANGED FROM** *port_1* **TO** *port_2*

Explanation: The TCP/IP resynchronization port number obtained from the DDF record of the BSDS is different than the value specified the last time DB2 started.

Changing the TCP/IP resynchronization port number can cause problems during distributed 2-phase commit

processing. Remote DRDA partners record DB2's resynchronization port number in their recovery logs. The resynchronization port number is used to reconnect to the appropriate DB2 member when communication failures occur during 2-phase commit processing. If a DB2 member changes its resynchronization port number, the remote DRDA partners may be unable to resolve indoubt units of work.

port_1   The TCP/IP resynchronization port number that was obtained from the DB2 BSDS during DB2 startup.

port_2   The TCP/IP resynchronization port number that was obtained from the coupling facility during DB2 startup. This is the TCP/IP resynchronization port number that was used the last time this member was started.

System action: DB2 accepts the new TCP/IP resynchronization port number, and DB2 startup continues.

Operator response: Contact the system programmer.

System programmer response: If the DB2 resynchronization port number was changed inadvertently, stop DB2 immediately. Use the change log inventory utility (DSNJU003) to correct the RESPORT value, and restart DB2.

If this message reflects a change that you want to make to DB2's resynchronization port number, no action is required. The resynchronization port number stored in the coupling facility for this member is updated, and this new port number will be used when DB2 validates the resynchronization port number for other DB2 members.

DSNR039I   *csect-name* **WARNING - TCP/IP RESYNCHRONIZATION PORT NUMBER** *port_1* **IS THE SAME AS MEMBER** *member_name*

Explanation: The TCP/IP resynchronization port of the starting DB2 is the same as the member indicated by *member_name*.

port_1   The TCP/IP resynchronization port number that was obtained from the DB2 BSDS during DB2 startup.

member_name
        The member name of the DB2 data sharing member that used the same resynchronization port number.

System action: The TCP/IP port number specified in the BSDS is accepted, and DB2 startup continues.

Operator response: Contact the system programmer.

System programmer response: Each member of the DB2 data sharing group should have a unique TCP/IP resynchronization port number. Use the change log inventory utility (DSNJU003) to specify a unique RESPORT value for this DB2 member and the DB2 member identified by *member_name*.

DSNR040I   *csect-name* **TRACKER RESTART ENDLRSN** *lrsn* **DOES NOT MATCH ENDLRSN** *lrsn* **ASSOCIATED WITH THE DATA SHARING GROUP**

Explanation: In a data sharing environment, a TRKRSITE DB2 will not restart if it detects inconsistencies in the reported values for ENDLRSN. The ENDLRSN specified on the Conditional Restart Control Record does not match the ENDLRSN in the system communications area (SCA) structure for the data sharing group.

System action: DB2 startup terminates.

Operator response: Collect the SYS1.LOGREC and SVC dump, and contact the system programmer.

System programmer response: If the ENDLRSN value for this member is incorrect, take the following actions:
- Restore the BSDS and Logs for his Tracker DB2 from the primary DB2
- Ensure the Conditional Restart Control Record is correct
- Restart DB2

If the ENDLRSN value in the SCA is incorrect, delete the SCA structure and restart all members of the TRKRSITE data sharing group with the same ENDLRSN.

DSNR042I   *csect-name* **WARNING - UR ROLLBACK HAS BEEN CANCELED AND IS INCOMPLETE FOR CORRELATION NAME =** *corrid* **CONNECTION ID =** *connid* **AUTHID =** *authid* **PLAN NAME =** *plan-name* **URID =** *urid*

Explanation: DB2 issues this message when rollback for the indicated thread has been canceled by either the CANCEL THREAD NOBACKOUT command or the RECOVER POSTPONED CANCEL command.

System action: Rollback processing for the thread stops and the unit of recovery is logged as canceled and complete.

User response: Examine the console for messages indicating what objects are being marked as REFP,RECP; those objects will need to be recovered.

DSNR043E *csect-name* **DB2 IS STARTING WITH AN UNSUPPORTED CODE LEVEL. CURRENT LEVEL:** *current-code-level* **VALID LEVEL(S):** *valid-code-levels*

Explanation: The DB2 that is starting is not at the proper code level. Some required maintenance, such as a fallback or coexistence SPE, has not been applied. The level information is different for each SPE level and each release. This information is for the use of your IBM service representative.

This message can occur in the following situations:

- The DB2 that is starting has completed a fallback from a later release and does not have the proper fallback SPE applied.
- The DB2 that is starting does not have the proper coexistence SPE to start in a data sharing group, which is at a later release level.
- The DB2 that is starting is not at a release level that supports coexistence with the data sharing group.

System action: DB2 terminates with reason code 00D900FA.

Operator response: Save the console output and inform the system programmer.

System programmer response: Check the DB2 Program Directory and Preventive Service Planning updates (PSP) to determine what SPE you need.

Verify that:

- Coexistence is supported between the releases you are trying to run.
- The DB2 startup procedure points to the correct set of load libraries.
- The correct SPE has been applied.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.
- Unformatted DSN1PRNT output of the DSNDB01.DBD01 page set. Use the method described in Part 5 of *Diagnosis Guide and Reference* to locate the inconsistent DBD in the DSNDB01.DBD01 page set.

DSNR044E *csect-name* **MEMBER** *member-name* **IS STARTED AT A LEVEL OF CODE** *started-code-level* **WHICH IS NOT COMPATIBLE WITH THE STARTING MEMBER. VALID LEVEL(S)** *valid-code-levels*

Explanation: The code levels are not compatible between the DB2 member that has already started and the DB2 member that is starting. You need to apply the correct SPE to the DB2 member that is at the earlier level.

The level information is different for each SPE level and each release. This information is for the use of your IBM service representative.

This message occurs when a starting DB2 member is at a later release level that cannot coexist with an existing DB2 member.

System action: After detecting the first member with which it cannot coexist, DB2 terminates with reason code 00D900FA.

Operator response: Save the console output and inform the system programmer.

System programmer response: Check the Program Directory and Preventive Service Planning updates (PSP) to ensure that all active members in the group have the proper SPE applied.

Verify that:

- Coexistence is supported between the releases you are trying to run.
- The DB2 startup procedure points to the correct set of load libraries.

- The correct SPE has been applied.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.
- Unformatted DSN1PRNT output of the DSNDB01.DBD01 page set. Use the method described in Part 5 of *Diagnosis Guide and Reference* to locate the inconsistent DBD in the DSNDB01.DBD01 page set.

DSNR045I   *csect-name* **DB2 SUBSYSTEM IS STARTING AND IT WAS NOT STARTED IN A PREVIOUS RELEASE WITH THE FALLBACK SPE APPLIED. FALLBACK SPE APAR:** *apar-number* **NEW RELEASE LEVEL:** *release* **KNOWN LEVEL(S):** *level*

Explanation: DB2 was not started with the FALLBACK SPE active before an attempt was made to start DB2 in a new release. The FALLBACK SPE must be applied, and the system must be started before DB2 can be started in a new release.

release level
> The release and level are different for each release, and are for the use of the IBM service representative.

apar-number
> The apar-number is the FALLBACK SPE that is missing.

System action: DB2 terminates with reason code 00D96001.

User response: Ensure that the FALLBACK SPE is applied, and DB2 is started before starting DB2 in a new release.

DSNR046I   *csect-name* **INCOMPLETE UNITS OF RECOVERY EXIST FOR** *subsys*

Explanation: This message is issued during DB2 shutdown if any incomplete units of recovery exist for the referenced member of a data sharing group. The message indicates that retained locks will be held by subsystem *subsys*.

System action: DB2 shutdown continues normally.

User response: Restart DB2 and resolve the incomplete units of recovery to release the retained locks held by subsystem *subsys*.

DSNR047I   *csect-name* **POSTPONED ABORT BACKOUT PROCESSING LOG RECORD AT RBA** *rba1* **TO RBA** *rba2*

Explanation: This message is produced periodically during the backout processing of postponed abort units of recovery, initiated by the -RECOVER POSTPONED command. The RBAs in this message can be used to monitor the backout progress.

During the backout process, DB2 reads the log in a backward direction and stops reading the log at the RBA matching the URID of the oldest postponed-abort UR *rba2*. *rba1* is the RBA of the log record currently being processed.

System action: Backout processing continues.

System programmer response: This is an information-only message. If the message continues to be issued and *rba1* remains the same, make sure there are no outstanding archive tape mounts that may be delaying the log read process. Otherwise, obtain a dump of the DB2 address spaces for problem determination.

The -DISPLAY THREAD TYPE(POSTPONED) command can be issued to display information about the postponed abort units of recovery. It is also possible to stop the backout process with the -RECOVER POSTPONED CANCEL command, but this will leave objects in an inconsistent state.

Related reference:

⮫ -RECOVER POSTPONED (DB2) (DB2 Commands)

⮫ -DISPLAY THREAD (DB2) (DB2 Commands)

DSNR048I	*csect-name* **UR BACKOUT PROCESSING LOG RECORD AT RBA** *rba1* **TO RBA** *rba2* **FOR CORRELATION NAME =** *correlation-name* **CONNECTION ID =** *connection-id* **LUWID =** *logical-unit-of-work-ID*=*token* **PLAN NAME =** *plan-name* **AUTHID =** *authorization-ID* **END USER ID =** *user-ID* **TRANSACTION NAME =** *transaction-name* **WORKSTATION NAME =** *workstation-name*

Explanation: This message is produced periodically during the backout process of an in-abort unit of recovery. The RBAs in this message can be used to monitor the backout progress. Together, CORRELATION NAME, CONNECTION ID, and LUWID identify a thread associated with the UR. If the LUWID is '*', the thread originated at this site. *token* is a unique token number associated with the LUWID. PLAN NAME and AUTHID further identify the thread associated with the UR. If the thread was created with client user information, the user's USERID, TRANSACTION NAME, and WORKSTATION NAME are displayed. Otherwise, these fields contain an '*'.

During the backout process, DB2 reads the log in a backward direction and stops reading the log at the RBA matching the URID of the in-abort UR *rba2*. *rba1* is the RBA of the log record currently being processed.

System action: Backout processing continues.

System programmer response: This is an information-only message. If the message continues to be issued and *rba1* remains the same, make sure there are no outstanding archive tape mounts that might be delaying the log read process. Otherwise, obtain a dump of the DB2 address spaces for problem determination.

It might be possible to stop the backout process with the NOBACKOUT option of the -CANCEL THREAD command, but this leaves objects in an inconsistent state.

Related concepts:

⮫ How DB2 rolls back work (DB2 Administration Guide)

Related reference:

⮫ -CANCEL THREAD (DB2) (DB2 Commands)

DSNR050I	*csect-name* **DB2 STARTED IN SYSTEM RECOVER PENDING MODE**

Explanation: DB2 has started in System Recover Pending mode in preparation for the RESTORE SYSTEM utility. No database update activity is allowed, and DB2 is in restricted access mode.

System action: DB2 startup continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

DSNR051I	*csect-name* **SYSPITR RESTART ENDLRSN** *lrsn* **DOES NOT MATCH ENDLRSN** *lrsn* **ASSOCIATED WITH THE DATA SHARING GROUP**

Explanation: For a System Level Recovery restart, all members of the data sharing group must truncate their logs with the same LRSN value that was specified in the SYSPITR conditional restart parameter.

System action: DB2 startup terminates.

Operator response: Collect the SYS1.LOGREC and SVC dump, and contact the system programmer.

System programmer response: Ensure that the conditional restart control record is correct, and then restart DB2. You can run the Print Log Map utility (DSNJU004) to examine the current conditional restart record.

DSNR052I	*csect-name* **DB2 MEMBER TO REMAIN ACTIVE IN LIGHT MODE TO RESOLVE INDOUBT UNITS OF RECOVERY**

Explanation: This message is issued at the end of restart when the DB2 member has been restarted with the LIGHT(YES) parameter and there are indoubt units of recovery (URs).

System action: The DB2 member remains up and running in light mode. This DB2 member will only allow connections for the purpose of resolving the indoubt URs. Also, certain commands are allowed. Any attempt to

connect to this DB2 to submit new SQL work will be disallowed with reason code 00F300A2.

User response: Ensure that the appropriate commit coordinators are started so that they can resync with the DB2 member to resolve the indoubts. Alternatively the -RECOVER INDOUBT command can be used to manually resolve the indoubts.

DSNR053I *csect-name* **ALL INDOUBT UNITS OF RECOVERY HAVE BEEN RESOLVED. LIGHT MODE DB2 NORMAL TERMINATION TO FOLLOW**

Explanation: This message is issued after all indoubt units of recovery are resolved for this DB2 member, which was restarted with the LIGHT(YES) parameter.

csect-name
 The name of the control section that issued the message.

System action: DB2 subsystem shutdown is initiated.

User response: No action is required.

Related concepts:

➦ Restart light (DB2 Data Sharing Planning and Administration)

DSNR054I **RESTART...BEGIN SCAN FOR CHECKPOINT**

Explanation: DB2 must scan the log backwards to locate the prior checkpoint.

System action: After the prior checkpoint is located on the log, restart continues.

DSNR055I *csect-name* **RESTART SUSPENDED SINCE** *time* **IN** *module offset maint-lvl*

Explanation: This message is produced periodically during the recovery phase of a restart if DB2 detects that the restart processor has remained suspended across a two minute interval. This message provides the time that the restart processor has suspended, and, the module, offset, and maintenance level for the suspend request.

System action: DB2 restart continues.

Operator response: Notify the system programmer.

System programmer response: If the message repeats every two minutes and DB2 restart does not appear to be progressing, look for additional messages that could indicate the reason for the restart stall. If no other messages are found, request SVC dumps of the DB2 and IRLM address spaces for problem determination. For data sharing, request dumps from all members of the data sharing group.

DSNR056I *csect-name* **RESTART SUSPENDED ON IRLM REQUEST. ONE HOLDER OF RESOURCE IS** *ss-name*. **SERVICE INFO: IRLM FUNC** *func-cd*, **RESOURCE** *res-name*.

Explanation: This message is produced periodically during the recovery phase of restart if DB2 detects that the restart processor has remained suspended across a two minute interval on an IRLM request. If DB2 is suspended for an IRLM lock in a data sharing environment, DB2 queries IRLM to determine what DB2 subsystem holds the lock. If successful, *ss-name* contains the DB2 subsystem name. Otherwise, *ss-name* contains a series of eight asterisks ('********'). The function code and resource name for the IRLM request is also provided for serviceability.

System action: DB2 restart continues.

Operator response: Notify the system programmer.

System programmer response: If the message repeats every two minutes and DB2 restart does not appear to be progressing, the DB2 subsystem displayed in the message might need to be recycled to allow this subsystem to restart. Request SVC dumps of the DB2 and IRLM address spaces before recycling any of the DB2 subsystems for problem determination. For data sharing, request dumps from all members of the data sharing group.

DSNR057I   *csect-name* **RESOURCE INFORMATION: DBID X'***dbid***', PSID X'***psid***', PART X'***part#***'**

Explanation: This message is produced periodically during the recovery phase of restart if DB2 detects that the restart processor has remained suspended across a two minute interval on an IRLM request. If DB2 is suspended for an IRLM lock in a data sharing environment, DB2 queries IRLM to determine what DB2 subsystem holds the lock. This message indicates that the locked resource is a type which is held on a data base object. The *dbid* contains the DBID and the *psid* contains the PSID in the case of a table space or ISOBID in the case of an index space. If the object is a partition of a partitioned pageset, then the *part#* contains the partition number, otherwise it is zero.

System action: DB2 restart continues.

Operator response: Notify the system programmer.

System programmer response: If the message repeats every two minutes and DB2 restart does not appear to be progressing, then an agent running on the DB2 subsystem identified in message DSNR056I could be holding on to a lock for an extended period of time, that is associated with the DBID, PSID, PART identified in this message. You might be able to use the data base object identification information in this message to identify the specific object which is encountering the problem. One way to check what object is encountering the problem is to query the DB2 catalog to find the data base name and pageset name, and then to use the DISPLAY DATABASE CLAIMERS or DISPLAY DATABASE LOCKS commands to identify the agent that is holding the lock. If an agent can be identified, you can cancel the agent to release the lock.

DSNR058I   *csect-name* **ACCESSING ARCHIVE LOG** *data-set-name*

Explanation: This message is produced periodically during the recovery phase of restart if DB2 detects that the restart processor has remained suspended across a two minute interval attempting to access an archive log data set. The *data-set-name* contains the data set name of the archive log that DB2 is accessing.

System action: DB2 restart continues.

Operator response: Notify the system programmer.

System programmer response: Look for and resolve any outstanding archive log tape mounts or HSM recall requests that could be causing DB2 restart to stall. If this does not resolve the problem, request SVC dumps of the DB2 and IRLM address spaces for problem determination.

DSNR059I   *csect-name* **GROUP INFORMATION IN THE SCA IS NOT CONSISTENT WITH A SYSPITR RESTART**

Explanation: Group information in the shared communication area (SCA) indicates that the current SYSPITR restart is invalid. All active members of the data sharing group must restart with the same SYSPITR conditional restart value.

System action: DB2 startup terminates.

Operator response: Notify the system programmer.

System programmer response: Make sure that all data sharing structures in the coupling facility have been deleted before the first SYSPITR restart, and that all active members have been restarted with the same SYSPITR conditional restart value.

DSNR060I   *csect-name* **THIS DB2 SUBSYSTEM IS MARKED AS DEACTIVATED AND CANNOT BE STARTED.**

Explanation: An attempt was made to start a DB2 subsystem, but the DB2 subsystem that issued the message cannot be started because its BSDS indicates that it has been deactivated.

csect-name
   The name of the control section that issued the message.

System action: An SVC dump is generated for abend 04E with reason code 00D10130. The restart is terminated.

System programmer response: Message DSNR060I usually indicates that a DB2 subsystem which has been deactivated was started mistakenly. If the DB2 subsystem should not have been deactivated, use the change log inventory utility (DSNJU003) to restore the member.

Problem determination: Collect the following diagnostic items:

- The SVC dump
- The syslog and LOGREC entries from the time of the error

Related reference:

➡ DSNJU003 (change log inventory) (DB2 Utilities)

Related information:

➡ 00D10130 (DB2 Codes)

DSNR061I   *csect-name* **MEMBER ID** *member-id* **OF GROUP** *group-name* **IS INVALID.** *request-type* **REQUEST FAILED.**

Explanation: A request to deactivate or destroy a member of a data sharing group specified an invalid member ID.

csect-name
        The name of the control section that issued the message.

member-id
        The member ID that was specified.

group-name
        The name of the data sharing group.

request-type
        The type of request: deactivate or destroy.

System action: The request is ignored and is purged from the BSDS. Processing continues.

System programmer response: No action is required.

DSNR062I   *csect-name* **MEMBER** *member-name* **ID** *member-id* **OF GROUP** *group-name* **CAN NOT BE** *requested-action* **BECAUSE IT IS NOT QUIESCED**

Explanation: A request was made to deactivate or destroy a data sharing group member, but the member is not quiesced. Only a quiesced member can be deactivated or destroyed.

csect-name
        The name of the control section that issued the message.

member-name **and** *member-id*
        The name and ID of the member that was specified.

group-name
        The name of the data sharing group.

requested-action
        The requested action, either DEACTIVATED (for a deactivation request) or DESTROYED (for a destruction request).

System action: The request is ignored. Processing continues.

System programmer response: No action is required.

DSNR063I   *csect-name* **MEMBER** *member-name* **ID** *member-id* **OF GROUP** *group-name* **SUCCESSFULLY** *requested-action*

Explanation: The request to deactivate or destroy a data sharing group member was successful.

csect-name
        The name of the control section that issued the message.

member-name **and** *member-id*
        The name and ID of the member that was specified.

group-name
        The name of the data sharing group.

DSNR064I

requested-action
    The requested action, either DEACTIVATED (for a deactivation request) or DESTROYED (for a destruction request).

System action: Processing continues.

System programmer response: No action is required.

DSNR064I   *csect-name requested-action* **OF MEMBER** *member-name* **ID** *member-id* **OF GROUP** *group-name* **FAILED. REASON =** *reason-code*

Explanation: A request was made to deactivate or destroy a data sharing group member, but the request was not successful.

csect-name
    The name of the control section that issued the message.

requested-action
    The requested action: DEACTIVATE or DESTROY.

member-name **and** *member-id*
    The name and ID of the member that was specified.

group-name
    The name of the data sharing group.

reason-code
    A hexadecimal code that indicates the reason for the failure.

System action: The request is ignored. Processing continues.

System programmer response: Use the information provided for the reason code to resolve the problem, then deactivate or destroy the member.

Problem determination: The syslog and LOGREC from the time of the failure might be necessary for diagnosis.

Related concepts:

DB2 reason codes (DB2 Codes)

Chapter 14. DSNS messages

Message identifiers that begin with `DSNS` communicate information about the storage manager and indicate the progress of DB2 when storage parameters are being specified.

DSNS001I   *csect_name* **SUBSYSTEM HAS EXCEEDED THE USER SPECIFIED REAL STORAGE USAGE THRESHOLD**

Explanation: You have requested to perform real storage monitoring by specifying a non-zero value for subsystem parameter SPRMRSMX. DB2 has determined that the amount of real storage used has reached or exceeded the amount of combined real and auxiliary storage allowed.

System action: The DB2 subsystem requests to record a DB2 SYS1.LOGREC entry, and an RC00E20033 abend dump is requested before DB2 subsystem termination is initiated. The system terminates with:

```
*DSNV086E = DB2 ABNORMAL TERMINATION
 REASON=00E20033
```

System programmer response: Determine whether the value specified for subsystem parameter, SPRMRSMX is realistic. If you decide to contact IBM support, provide the DB2 SYS1.LOGREC entry and the RC00E20033 dump. Also, you will most likely be asked to provide IFCID225 records, which show historical storage usage, in order to determine what area is responsible for the real frame build up.

DSNS002I   *csect-name* **THE VALUE SPECIFIED FOR SUBSYSTEM PARAMETER REALSTORAGE_MAX IS EQUAL OR GREATER THAN AVAILABLE REAL STORAGE**

Explanation: The value of the REALSTORAGE_MAX subsystem parameter is equal to or greater than the amount of real storage that is available to the address space.

csect-name
> The name of the control section that issued the message.

System action: DB2 continues with normal startup.

System programmer response: Increase the specified value if appropriate. The specified value might be normal if you expect that much of the real storage usage for the installation is to be paged out to auxiliary storage devices.

DSNS003I   *csect-name***SUBSYSTEM WARNING THRESHOLD FOR REAL STORAGE HAS BEEN DETECTED**

Explanation: The system has consumed a critical amount of the real and auxiliary storage that is allotted to this DB2 subsystem by the REALSTORAGE_MAX subsystem parameter.

csect-name
> The name of the control section that issued the message.

System action: Processing continues but DB2 enters contraction mode if it is not already enabled. If real storage growth continues and reaches the limit defined by the value of the REALSTORAGE_MAX subsystem parameter, DB2 terminates.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: If appropriate, increase the value of the REALSTORAGE_MAX subsystem parameter, and determine why the real storage usage is approaching the maximum allowed. If possible, reduce the workload that is entering this DB2 member until the condition is relieved.

DSNS004I

Related reference:

▶ Subsystem parameters that are not on installation panels (DB2 Installation and Migration)

Related information:

"DSNV516I" on page 972

DSNS004I   *csect-name* **SUBSYSTEM THRESHOLD FOR REAL STORAGE RELIEVED**

Explanation: The real storage usage has dropped below the warning threshold.

csect-name
> The name of the control section that issued the message.

System action: Processing continues.

Operator response: Determine why the shortage occurred and try to prevent that from happening again. Use IFCID 0225 to analyze prior trends. The following trace fields are relevant for this investigation:

- QW0225AX
- QW0225ComStg_Aux
- QW0225ComStg_Real
- QW0225HVAuxSlots
- QW0225HVPagesInReal
- QW0225PriStg_Real
- QW0225PriStg_Aux
- QW0225RL
- QW0225ShrStg_Real
- QW0225ShrStg_Aux

Related reference:

▶ Subsystem parameters that are not on installation panels (DB2 Installation and Migration)

Related information:

"DSNS003I" on page 431

Chapter 15. DSNT messages

Message numbers that begin with DSNT communicate information about the service controller.

Messages issued by DSNTINST, DSNTINS1, and DSNTINS2 installation CLISTs use the following internal severity levels:

0	Successful execution: CLIST progress and tracing messages (processing continues)
4	Warning: some adjustments might be made by the installation CLISTs (processing and JCL editing continues)
8	Error: unable to continue (no JCL is edited)

DSNT000I   *program-name* **STARTED FOR** *subsystem-id* **SUBSYSTEM, VERSION** *db2-version* **(***system-mode***)**

Explanation: This informational message is returned at the start of the indicated program and reports information about the DB2 version and the processing mode of the DB2 subsystem that the program has connected to.

program-name
> The name of the program.

subsystem-id
> The DB2 subsystem identifier.

db2-version
> The DB2 version.

system-mode
> The current DB2 mode. Possible values are:
>
> **CM***n*   Conversion mode, where *n* represents the version of DB2 that is being migrated from.
>
> **NFM**   New-function mode

System action: Processing continues.

User response: No action is required.

DSNT001I   *program-name* **INTERNAL ERROR AT LOCATION** *location-name* **BECAUSE OF UNEXPECTED OBJECT OF TYPE** *object-type* **WITH NAME** *object-name*

Explanation: The program was terminated because an object was not expected.

program-name
> The name of the program.

location-name
> Diagnostic information that identifies the function that was involved in the error.

object-type
> The type of object.

object-name
> The name of the object, as either a specific name or a qualified name:
>
> *object-name* **is a specific name**
> > SPECIFIC NAME *name*
>
> *object-name* **is a qualified name**
> > *qualifier.name*

System action: Processing terminates.

      433

User response: Verify that the latest maintenance has been applied, and apply maintenance if necessary. If the problem persists, contact IBM Software Support.

Severity: 12 (severe error)

DSNT002I *program-name* **RECEIVED AN UNEXPECTED SQLCODE OF** *sql-code* **FROM THE** *sql-request* **REQUEST**

Explanation: During processing of an SQL request, the program received an unexpected SQL code. Further processing has been halted.

program-name
> The name of the program.

sql-code   A numeric value that indicates why the SQL statement execution was unsuccessful.

sql-request
> All or part of the SQL statement that was issued. For a long statement, only a portion of the statement is provided.

System action: The program rolls back all SQL changes and terminates.

User response: Examine the standard SQLCA diagnostic messages and output in the SYSPRINT output log at the point of failure to determine the cause of the error. Correct the error and rerun the program.

Problem determination: This message is followed by one or more of the standard SQLCA diagnostic messages: DSNT404I, DSNT408I, DSNT411I, DSNT415I, DSNT416I, DSNT417I, and DSNT418I.

Severity: 12 (severe error)

Related information:

DSNT003I   **DSNT003I DSNTRIN HAS ENCOUNTERED AN ERROR IN THE CONTROL STATEMENT, DD =** *dd-name*, **ERROR =** *error-text*, **LINE =** *line-number*, **STATEMENT =** *error-statement*

Explanation: DSNTRIN, the configuration tool for routines supplied by DB2, encountered an error during configuration control statement processing.

error-text
> Additional information about the error. Expected values include:
>
> - CONTROL STATEMENT HAS AN INVALID FORMAT
> - ROUTINE NAME HAS AN INVALID FORMAT
> - CONTROL STATEMENT EXCEEDS 720 BYTES
> - NO VALUE FOUND FOR THE FIRST|SECOND|THIRD PARAMETER OR ONE OR MORE CONTROL STATEMENTS WAS NOT SEPARATED BY A BLANK LINE
> - ONLY ONE PARAMETER FOUND
> - ACTUAL PARAMETERS: *x*, EXPECTED PARAMETERS: *y*
> - UNEXPECTED PARAMETER *parameter-name* FOUND
> - CONTROL STATEMENT FOR *routine-name* NOT SUPPORTED
> - CONTROL STATEMENT FOR *routine-name* ALREADY PROCESSED
> - CONTROL STATEMENT FOR *routine-name* NOT FOUND

line-number
> The location of the error.

dd-name
> The name of the data definition statement.

error-statement
> The control statement in error.

System action: Processing terminates.

User response: Locate the control statement in error at the indicated line in the input allocated by the data definition indicated by *dd-name*. Analyze the statement for the indicated error. Correct the problem, and resubmit job DSNTIJRT to continue DSNTRIN processing.

Severity: 12 (severe error)

Related concepts:

⇨ Job DSNTIJRT (DB2 Installation and Migration)

DSNT004I   **DSNTRIN HAS SUCCESSFULLY CREATED** *object-type object-name*

Explanation: DSNTRIN has successfully created the SQL object.

object-type
> The type of object.

object-name
> The name of the object, as either a specific name or a qualified name:

> *object-name* **is a specific name**
> > `SPECIFIC NAME` *name*

> *object-name* **is a qualified name**
> > `qualifier.name`

System action: Processing continues.

User response: No action is required.

DSNT005I   **DSNTRIN HAS SKIPPED** *processing-type* **PROCESSING OF** *object-type*, *object-name* **FOR REASON**
                reason-text

Explanation: DSNTRIN could not complete SQL processing for the indicated object.

processing-type
> The type of SQL processing that was attempted.

object-type
> The type of object.

object-name
> The name of the object, as either a specific name or a qualified name:

> *object-name* **is a specific name**
> > `SPECIFIC NAME` *name*

> *object-name* **is a qualified name**
> > `qualifier.name`

reason-text
> `REQUIRES NEW FUNCTION MODE`

System action: Processing continues.

User response: If *reason-code* is 10, run the job again after DB2 enters new-function mode. Otherwise, no action is required.

Related reference:

⇨ Statements (DB2 SQL)

DSNT006I **DSNTRIN HAS GRANTED** *access-type* **ACCESS ON** *object-type object-name* **TO AUTHORIZATION IDS** *authorization-ids*

Explanation: DSNTRIN has granted access on one or more objects to one or more authorization IDs as indicated in the message.

access-type
> The access that was granted.

object-type
> The type of object.

object-name
> The name of the object.

authorization-ids
> The authorization IDs to which access was granted.

System action: Processing continues.

User response: Review the access given to ensure that it is appropriate for each authorization ID that it was granted. If the access is not appropriate, modify the GRANTTO parameter of the configuration control statement for each stored procedure or user-defined function that requires a different access list, and resubmit job DSNTIJRT.

Related concepts:

↪ Job DSNTIJRT (DB2 Installation and Migration)

DSNT007I **DSNTRIN HAS DROPPED EXISTING** *object-type*, *object-name*

Explanation: The indicated object has been dropped. DSNTRIN drops existing objects only in REINSTALL mode or REINITIALIZE mode.

object-type
> The type of object.

object-name
> The name of the object, as either a specific name or a qualified name:

> *object-name* **is a specific name**
> > SPECIFIC NAME *name*

> *object-name* **is a qualified name**
> > *qualifier.name*

System action: Processing continues.

User response: No action is required.

DSNT008I **DSNTRIN HAS SET THE WLM ENVIRONMENT FOR** *routine-type schema.routine-name* **TO** *environment-name*

Explanation: The DB2-supplied stored procedure or user-defined function indicated by *schema.routine-name* has been altered to use the indicated WLM environment.

routine-type
> The type of routine: user-defined function or stored procedure.

object-name
> The name of the object, as either a specific name or a qualified name:

> *object-name* **is a specific name**
> > SPECIFIC NAME *name*

> *object-name* **is a qualified name**
> > *qualifier.name*

environment-name
> The name of the WLM environment.

System action: Processing continues.

User response: Review the WLM environment setting for the routine to ensure that it is appropriate. If the environment is not appropriate for the routine, modify the WLMENV parameter of the configuration control statement for each stored procedure or user-defined function that requires to be run in a different WLM environment, and resubmit job DSNTIJRT.

Related concepts:

⮕ Job DSNTIJRT (DB2 Installation and Migration)

DSNT009I   **DSNTRIN BIND PACKAGE REQUEST SUCCEEDED WITH RETURN CODE** *return-code* **FOR COLLECTION ID** *collection-id.package-id*

Explanation: A bind request for the indicated package was completed.

return-code
> The return code issued by the command.

collection-id.package-id
> The package that was bound.

Other messages related to the bind request might precede this message.

Multiple forms: This message has two forms. The short form is shown above. The long form is: DSNTRIN BIND PACKAGE REQUEST SUCCEEDED WITH RETURN CODE *return-code* FOR COLLECTION ID *collection-id.package-id* WITH OWNER *owner-id*

The long form of this message is issued if the OWNER option was specified for the bind request.

owner-id
> The package owner.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ BIND PACKAGE (DSN) (DB2 Commands)

DSNT010I   *program-name request-type* **REQUEST FOR** *collection-id.package-id* **FAILED WITH RETURN CODE** *return-code*

Explanation: The operation requested for the indicated package was rejected by DB2. Diagnostic messages from DB2 precede this message.

program-name
> The name of the program.

request-type
> The type of request: BIND PACKAGE, BIND PLAN, FREE PACKAGE, or FREE PLAN.

collection-id.package-id
> The package that could not be bound or freed.

return-code
> A numeric value that indicates the reason for the failure.

System action: The request fails and processing terminates.

User response: Review the diagnostic messages from DB2, correct the problem, and rerun the specified program.

Severity: 12 (severe error)

DSNT011I

Related information:

➦ DB2 and related commands (DB2 Commands)

DSNT011I   *program-name* **INPUT PARAMETERS ARE INVALID, REASON CODE =** *reason-code*, **TEXT =**
reason-text

Explanation: The program encountered an error when it attempted to process the input parameters that were
passed to it in the PARM parameter or the SYSIN DD.

program-name
> The name of the program.

reason-code **and** *reason-text*
> The reason code (*reason-code*) describes the cause of the error, and the reason text (*reason-text*) provides
> additional information that is specific to the reason code.

> *reason-code* = **1**
>> The number of parameters specified is incorrect. The *reason-text* provides the following additional
>> information: *number-passed* PARAMETERS WERE PASSED BUT *number-expected* PARAMETERS ARE
>> EXPECTED, where:

>> *number-passed*
>>> The number of parameters that were passed to the program.

>> *number-expected*
>>> The number of parameters that were expected by the program.

> *reason-code* = **2**
>> A required parameter was not passed. The *reason-text* provides the following additional information:
>> REQUIRED PARAMETER *parameter-name* WAS NOT PASSED, where:

>> *parameter-name*
>>> The name of the parameter that was expected, but that was not provided.

> *reason-code* = **3**
>> An unrecognized parameter was passed. The *reason-text* provides the following additional
>> information: UNRECOGNIZED PARAMETER *parameter-name* WAS PASSED, where:

>> *parameter-name*
>>> The name of the parameter that was provided where it was not expected.

> *reason-code* = **4**
>> The value of one of the parameters is not valid. The *reason-text* provides the following additional
>> information: THE VALUE FOR PARAMETER *parameter-name* IS NOT VALID, where:

>> *parameter-name*
>>> The name of the parameter that is invalid where it was provided.

> *reason-code* = **5**
>> The routine specified in DD:SYSIN is not a valid DB2-supplied routine. The *reason-text* provides the
>> following additional information: *schema.routine-name* IS NOT A VALID DB2-SUPPLIED ROUTINE,
>> where:

>> *schema.routine-name*
>>> The name of the routine, qualified by the name of the schema in which the routine was
>>> created.

> **6**
>> A parameter was specified two or more times. The *reason-text* provides the following additional
>> information: PARAMETER *parameter-name* WAS PASSED MORE THAN ONCE.

> **7**
>> The statement exceeded the maximum bytes allowed. The reason-text provides the following
>> additional information: STATEMENT EXCEEDS *number-of-bytes* BYTES

System action: Processing terminates.

User response: Correct the problem and run the program again.

Severity: 12 (severe error)

DSNT012I   *program-name rrsaf-function* **REQUEST FAILED. DSNRLI RETURN CODE =** *dsnrli-return*, **RRSAF RETURN CODE =** *rrsaf-return*, **RRSAF REASON CODE =** *rrsaf-reason*

Explanation: The program encountered an error when it attempted to establish a connection to the DB2 subsystem.

program-name
> The name of the program.

rrsaf-function
> The DB2 Recoverable Resource Services Attachment Facility (RRSAF) function that was called.

dsnrli-return
> A numeric value that indicates the reason for the failure.

rrsaf-return
> The return code issued by DB2 Resource Recovery Services attachment facility (RRSAF).

rrsaf-reason
> The reason code issued by the DB2 Resource Recovery Services attachment facility (RRSAF).

System action: The program terminates.

User response: Use the return codes provided for DSNRLI and RRSAF to determine the problem, and proceed as instructed.

Severity: 12 (severe error)

Related concepts:

▷ RRSAF return codes and reason codes (DB2 Application programming and SQL)

▷ DB2 reason codes (DB2 Codes)

DSNT013I   **DSNTRIN FOR DB2 VERSION** *program-version* **CANNOT BE RUN ON DB2 VERSION** *db2-version*

Explanation: The code level of program DSNTRIN does not match the DB2 code level. For example, during migration to a new version of DB2, the program was executed on a downlevel member in data sharing coexistence mode.

program-version
> The version of the program.

db2-version
> The DB2 version.

System action: Processing terminates.

User response: In DB2 data sharing coexistence mode, run the program on a member that has been migrated to the new version. Otherwise, verify that the load library specified in the JOBLIB or STEPLIB for executing the program has the same code level as DB2.

Severity: 12 (severe error)

DSNT014I   *program-name file-operation* **FILE ERROR FOR DD:***dd-name*. **DIAGNOSTIC TEXT** *diagnostic-text*

Explanation: The program encountered an error for the data set allocated to the data definition statement.

program-name
> The name of the program.

file-operation
> The file operation during which the error occurred: OPEN, READ, WRITE, or CLOSE.

dd-name
> The name of the data definition statement.

diagnostic-text
> Additional diagnostic information for the error.

System action: The file operation fails and the program terminates.

User response: Correct the error indicated by the diagnostics, which are provided by the IBM Language Environment.

Severity: 12 (severe error)

Related reference:

➠ Language Environment Run-Time Messages (z/OS Language Environment Run-Time Messages)

DSNT015I **DSNTRIN PREVIEW OPTION WAS SPECIFIED, THE** *install-type* **ACTION WAS NOT PERFORMED, INSTALL JCL WAS GENERATED AND STORED IN** *stored-location*

Explanation: The DB2-supplied routines and supporting objects were not installed, configured, or bound on the system because the preview option was specified. If a DD statement called JCLOUT was specified in the JCL, then a JCL containing the actual SQL and BIND statements that will be used to install the DB2-supplied routines is generated. A detailed report is also written to SYSPRINT to which objects were included for processing.

Multiple forms: This message has multiple forms. In addition to the standard form shown above, you might see the following form if no JCL job was generated:

```
DSNTRIN PREVIEW OPTION WAS SPECIFIED, THE
install-type ACTION WAS NOT PERFORMED
```

install-type
> The type of installation performed.

stored-location
> The name of the data set and member where the generated installation job was stored.

System action: The DB2-supplied routines and supporting objects were not installed, configured, or bound. A report describing impact to affected objects is written to SYSPRINT. Optionally, if the job included the JCLOUT DD statement, a JCL containing the actual SQL and BIND statements that can be used to install the DB2-supplied routines is generated.

User response: Review the report and verify which DB2-supplied routines and supporting objects are expected to be created then resubmit the DSNTIJRT installation job without the preview option or submit the generated JCL if you are ready to install the DB2-supplied routines and supporting objects.

Related concepts:

➠ Job DSNTIJRT (DB2 Installation and Migration)

Related tasks:

➠ Installing DB2-supplied routines during installation (DB2 Installation and Migration)

➠ Installing DB2-supplied routines during migration (DB2 Installation and Migration)

DSNT016I **DSNTXAZP THE CLIST INPUT MEMBER** *ds-name*(*member*) **WAS SUCCESSFULLY CREATED**

Explanation: DSNTXAZP has successfully created and stored the CLIST defaults input member.

ds-name(*member*)
> The name of the data set and member.

System action: The CLIST defaults input member is generated and a report is written to SYSPRINT.

Processing continues.

User response: No action is required.

DSNT017I **DSNTXAZP AN ERROR WAS ENCOUNTERED DURING PROCESSING, ERROR =** *error-text*

Explanation: The program encountered an error during processing.

error-text
> Additional information about the error. Expected values include:
>
> • THE DB2 VERSION AND THE VERSION OF THE CLIST INPUT MEMBER ARE NOT THE SAME
> • THE DB2 VERSION AND THE VERSION OF DSNTXAZP ARE NOT THE SAME

- FAILED TO CONNECT TO DB2
- DSNWLI CALL FAILED, IFCARC1=*ifcarc1*, IFCARC2=*ifcarc2*, IFCABNM=*ifcabnm*, IFCAGRSN=*ifcagrsn*, IFCAGBNM=*ifcagbnm*
- RETRIEVE FILE INFO FAILED (*diagnostic-text*)
- INVALID ACTION '*action*' WAS SPECIFIED
- UNEXPECTED NUMBER OF PARAMETERS FOUND (ACTUAL=*number-found*, EXPECTED=*number-expected*)
- MAXIMUM LENGTH FOR ACTION IS *max-length*
- MAXIMUM LENGTH FOR SSID IS *max-length*
- INSUFFICIENT MEMORY AVAILABLE
- THE INPUT AND OUTPUT DATA SET MEMBER ARE THE SAME
- VERSION *version* IS NOT SUPPORTED

Tokens used within the *error-text* value include:

action   The action that was specified.

diagnostic-text
        Diagnostic information for the failure.

number-found
        The number of parameters provided.

number-expected
        The number of parameters expected.

max-length
        The maximum length permitted for the indicated object.

ifcagbnm
        A numeric value that indicates the number of bytes that could not fit in the return area.

ifcabnm   A numeric value that indicates the number of bytes that could not fit in the return area.

ifcagrsn   A numeric value that indicates the reason for the failure.

ifcarc1 **and** *ifcarc2*
        Numeric values that indicate the reason for the failure.

version   The version that was specified.

System action: The program terminates and the CLIST input member is not created.

User response: Use the information provided in the message to identify and resolve the problem.

Severity: 12 (severe error)

DSNT018I   **DSNTRVFY - A WARNING WAS ENCOUNTERED DURING CALL TO ROUTINE, WARNING =** *warning-text*

Explanation: The call to routine encountered a warning during processing.

Multiple forms: This message has multiple forms. In addition to the standard form shown above, you might see the following form if no warning message is available:

A WARNING WAS ENCOUNTERED DURING CALL TO ROUTINE

warning-text
        Information about the warning.

System action: The routine is validated.

User response: ook for additional messages that might explain the cause of the warning or check the manuals for additional information on the routine and proceed as instructed by the message or documentation.

Severity: 4 (warning)

Related concepts:

➦ Routines (Introduction to DB2 for z/OS)

DSNT019I   **DSNTRVFY - THE** *item* **FOUND IS** *'value-1'*, **EXPECTED IS** *'value-2'*

Explanation: The current value for an item does not match the expected value.

item   The item where the discrepancy exists.

'value-1'
        The value that was found.

'value-2'
        The value that was expected

System action: Processing continues.

User response: Determine what the expected value of the item is.

Verify that the installation job DSNTIJRT ran with the correct mode of installation.

Severity: 8 (error)

DSNT020I   *program-name* **THE CURRENTLY INSTALLED SERVICE DEFINITION DOES NOT MATCH THE BASEID OF THE EXTRACTED WLM SERVICE DEFINITION**

Explanation: The installed WLM service definition does not match the last extracted WLM service definition. This condition happens if the WLM service definition was updated after the WLM service definition was extracted.

program-name
        The name of the program.

System action: The WLM service definition is not installed.

User response: Resubmit the request.

Severity: 8 (error)

DSNT021I   *program-name* **WLM SERVICE POLICY** *policy-name* **NOT FOUND**

Explanation: The WLM service policy that was supposed to be activated was not found.

program-name
        The name of the program.

policy-name
        The name of the service policy.

System action: The WLM service policy is not activated.

User response: If you want to activate a WLM service policy, specify a name that is already defined to WLM.

Severity: 8 (error)

DSNT022I   *program-name* **WLM APPLICATION ENVIRONMENT** *wlm-environment* **IS ALREADY DEFINED**

Explanation: The WLM application environment that was supposed to be added is defined to WLM.

program-name
        The name of the program.

wlm-environment
        The name of the existing WLM environment.

System action: The WLM application environment is not added.

User response: If you want to add a new WLM application environment, use a name that is not yet defined to WLM.

Severity: 4 (warning)

DSNT023I   *program-name requested-action* **SUCCESSFUL**

Explanation: The action performed by the program was successful.

program-name
> The name of the program.

requested-action
> The action that was successful:
> - ACTIVATE WLM POLICY *policy-name*, where *policy-name* is the WLM policy.
> - ADD WLM APPLICATION ENVIRONMENT *environment-name*, where *environment-name* is the WLM environment name.
> - DISPLAY WLM INFORMATION
> - QUERY SERVICE POLICY

System action: The action is processed.

User response: No action is required.

DSNT024I   *program-name requested-action* **FAILED**

Explanation: The action performed by the program failed.

program-name
> The name of the program.

requested-action
> The action that failed:
> - ACTIVATE WLM POLICY *policy-name*, where *policy-name* is the WLM policy.
> - ADD WLM APPLICATION ENVIRONMENT *environment-name*, where *environment-name* is the WLM environment name.
> - DISPLAY WLM INFORMATION
> - QUERY SERVICE POLICY

Multiple forms: This message has two forms. The short form is shown above. The long form is: `program-name requested-action FAILED - PROGRAM IS NOT APF-AUTHORIZED`

The long form is used when the reason for the failure is APF authorization.

System action: The action is not processed.

User response: Look for additional messages that might explain the cause of the failure and proceed as instructed by the messages.

If the message includes the text `PROGRAM IS NOT APF-AUTHORIZED`, make sure that the program resides in an APF-authorized library and that all libraries in the STEPLIB or JOBLIB concatenation for the job that invokes the program are APF-authorized.

Severity: 8 (error)

DSNT025I   *program-name* **WAS UNABLE TO PROCESS THE BIND OR FREE REQUEST, REASON CODE =** *reason-code*

Explanation: The program was not able to bind or free the package or plan.

program-name
> The name of the program.

reason-code
> A numeric value that indicates the reason for the failure.
>
> **1**   The data definition statement was missing.
>
> **2**   The OPEN request failed.

DSNT026I

3	The READ request failed.
4	The request to retrieve file information failed.
5	The call to the bind interface failed.
771	A GETMAIN request was not successful.
772	Unable to set up the bind interface.
773	Unable to remove the bind interface.

Multiple forms: This message has two forms. The short form is shown above. The long form is: *program-name* WAS UNABLE TO PROCESS THE BIND OR FREE REQUEST, REASON CODE = *reason-code*,DDNAME = *dd-name*,ERROR = *error text*

The long form is used during I/O operations (file opening or closing).

dd-name
> The name of the data definition statement.

error text
> Description of error, generated during I/O operation. This text is sometimes a Language Environment error message.

System action: The request fails and the program terminates.

User response: Correct the error based on the information provided for the *reason-code* value:

1 Verify that the job used to execute the program has allocated the SYSTSPRT data definition as SYSOUT=* and the SYSUT1 as a workfile with these attributes:

```
//SYSUT1 DD UNIT=SYSDA,
//          SPACE=(32000,(30,30)),
//          DCB=(RECFM=VB,LRECL=133)
```

771 Increase region, and resubmit the job.

All other values
> Respond according to the information received in *error text*. The source of the information is:
>
> **2, 3, or 4**
> > IBM Language Environment
>
> **5, 772, or 773**
> > BIND interface

Severity: 12 (severe error)

DSNT026I   **DSNTRVFY VALIDATION FOR** *schema.routine-name* **WAS NOT PERFORMED BECAUSE THE ROUTINE IS NOT ENABLED ON THIS DB2 SUBSYSTEM**

Explanation: The indicated routine is currently defined with the DB2-reserved WLM application environment name of DSNWLM_NOT_ENABLED, which indicates that the routine is not enabled to run on this subsystem. Because the routine is not enabled to run on this subsystem, validation processing is skipped.

schema.routine-name
> The name of the routine, qualified by the name of the schema in which the routine was created.

System action: The routine is not validated.

User response: Verify that routine is intended to be disabled on this DB2 subsystem. If it is not, change the WLM application environment to a valid one.

Severity: 4 (warning)

DSNT027I   DSNTRVFY VALIDATION FOR *schema.routine-name* **SUCCESSFUL**

Explanation: The validation for the indicated routine was successful. All validation steps completed successfully.

schema.routine-name
> The name of the routine, qualified by the name of the schema in which the routine was created.

System action: The routine is validated.

User response: No action is required.

DSNT028I   DSNTRVFY - WLM ENVIRONMENT DETAILS NAME *environment-name* **PROC** *procedure-name* **STATE** *wlm-state* **NUMTCB** *wlm-numtcb*

Explanation: This message provides additional WLM information about the validated routine, and normally follows message DSNT027I, DSNT029I, or DSNT031I.

environment-name
> The name of the WLM environment.

procedure-name
> The name of the WLM catalog procedure.

wlm-state
> The state of the WLM environment, such as ACTIVE.

wlm-numtcb
> The number of task control blocks, which represents the number of tasks that can run simultaneously in a single WLM address space.

System action: Additional WLM information is displayed.

User response: No action is required.

Related information:

"DSNT027I"

"DSNT029I"

"DSNT031I" on page 446

DSNT029I   DSNTRVFY VALIDATION FOR *schema.routine-name* **COMPLETED WITH WARNINGS**

Explanation: The validation for the indicated routine completed with one or more warnings. Some validation steps completed successfully, but others did not.

schema.routine-name
> The name of the routine, qualified by the name of the schema in which the routine was created.

System action: The routine is validated.

User response: Look for additional messages that might explain the cause of the warnings and proceed as instructed by the messages.

Severity: 4 (warning)

DSNT030I   DSNTRVFY - THE ROUTINE IS NOT DEFINED TO SYSIBM.SYSROUTINES

Explanation: The routine that is identified by *schema.routine-name* in message DSNT031I was not defined to the SYSIBM.SYSROUTINES catalog table. This message normally follows message DSNT031I.

System action: Processing continues with the next routine.

User response: Determine if the routine identified in message DSNT031I is the intended one. If it is not, correct the error, and resubmit the DSNTIJRT installation job.

Severity: 8 (error)

Related information:
"DSNT031I"

DSNT031I   **DSNTRVFY VALIDATION FOR** *schema.routine-name* **FAILED**

Explanation: The validation of the routine was not successful.

schema.routine-name
        The name of the routine, qualified by the name of the schema in which the routine was created.

System action: The routine is not validated. Processing continues.

User response: Look for additional messages that might explain the cause of the failure and proceed as instructed by the messages.

Severity: 8 (error)

DSNT032I   **DSNTRVFY - THE CALL TO ROUTINE WAS SUCCESSFUL**

Explanation: The routine was successfully called.

System action: Processing continues.

User response: No action is required.

DSNT033I   *program-name program-type* **ENDED, RETURN CODE =** *return-code*

Explanation: The specified batch program has completed.

program-name
        The type of program executed.

program-type
        The type of program.

return-code
        The most severe return code encountered at any time during processing.

System action: The n program terminates.

User response: If the return code is 4 or higher, look for other messages that might explain the cause of the warning or failure, and proceed as instructed by the message.

DSNT034I   **DSNTRVFY - AN ERROR WAS ENCOUNTERED DURING CALL TO ROUTINE, SQLCODE =**
        sql-code

Explanation: The call to routine encountered an SQL error during processing.

sql-code   A numeric value that indicates why the SQL statement execution was unsuccessful.

Multiple forms: This message has multiple forms. In addition to the standard form shown above, you might see one of the following forms if no SQL return code is available:

AN ERROR WAS ENCOUNTERED DURING CALL TO ROUTINE

AN ERROR WAS ENCOUNTERED DURING CALL TO ROUTINE,
ERROR = *error-text*

error-text
        Information about the error.

System action: The routine is not validated. Processing continues.

User response: Use the information provided for the SQL error code to resolve the problem.

Severity: 8 (error)

Related concepts:

SQL error codes (DB2 Codes)

DSNT035I   DSNTRVFY VALIDATION PROGRAM FAILED TO INITIALIZE, REASON CODE = *reason-code*

Explanation: The validation program failed to complete initialization because of errors.

reason-code
>   A numeric value that indicates the reason for the failure.

> | | |
> |---|---|
> | **1** | There is insufficient memory to run the program. |
> | **2** | This version of DB2 is not supported by the program. |
> | **3** | The program was unable to process the SYSIN DD input. |
> | **4** | An error was encountered during extended console processing. |
> | **5** | The bind request for DSNTRVFY failed. |
> | **6** | The program is not APF-authorized. |

System action: The routine validation program terminates.

User response: Use the reason code to determine the cause of the error. Look for additional messages in the job output, on the system console, or on the operator console. Correct the problem, and run the program again.

Severity: 12 (severe error)

DSNT036I   DSNTRVFY IGNORED ROUTINE *schema.routine-name* **IN LINE** *line-number* **BECAUSE THE ROUTINE ALREADY EXISTS**

Explanation: The routine was already included for validation. The duplicate routine will be ignored.

schema.routine-name
>   The name of the routine, qualified by the name of the schema in which the routine was created.

line-number
>   The line where the second instance of *schema.routine-name* was found.

System action: The duplicate routine is not selected and processing continues.

User response: Remove the duplicate name from the SYSIN DD. If you plan to reuse the job, remove the duplicate name from the SYSIN DD before you rerun the job.

Severity: 4 (warning)

DSNT037I   *program-name action* WLM SERVICE DEFINITION REQUEST FAILED, MACRO = *macro-name*, **RETURN CODE =** *return-code*, **REASON CODE =** *reason-code*

Explanation: An error was encountered during execution of a WLM macro.

program-name
>   The name of the program.

action **and** *macro-name*
>   The action that was attempted and the WLM macro that was used:

action	macro-name
ACTIVATE	IWMPACT
EXTRACT	IWMDEXTR
INSTALL	IWMDINST
QUERY	IWMPQRY

return-code
>   The severity or type of failure.

reason-code
> A numeric value that indicates the reason for the failure.

Multiple forms: This message has two forms. The short form is shown above. The long form is: *program-name action* WLM SERVICE DEFINITION REQUEST FAILED, MACRO = *macro-name*, RETURN CODE = *return-code*, REASON CODE = *reason-code*, VALIDATE RSN = *validate-reason*, VALIDATE OFFSET = *validate-offset*

The long form of the message is issued when *action* is INSTALL.

validate-reason
> A hexadecimal reason code that identifies the specific error in the input service definition found during validity checking.

validate-offset
> Location in the service definition file where the validation error occurred. This value is the offset from the beginning of the service definition (IWMSERVD) to the section of the input service definition where validity check processing found an error described by the reason code returned in *validate-reason*.

System action: The program terminates.

User response: Use the information provided for the macro identified by *macro-name* to resolve the problem.

Severity: 12 (severe error)

Related reference:

▶ IWMDEXTR Return and Reason Codes (MVS Programming: Workload Management Services)

▶ IWMDINST Return and Reason Codes (MVS Programming: Workload Management Services)

▶ IWMPACT Return and Reason Codes (MVS Programming: Workload Management Services)

▶ IWMPQRY Return and Reason Codes (MVS Programming: Workload Management Services)

DSNT038I   DSNTRVFY - THE WLM ENVIRONMENT *environment-name* **IS NOT AVAILABLE**

Explanation: The WLM application environment is either not defined or is not available.

environment-name
> The name of the WLM environment.

System action: The routine is not validated. Processing continues with the next routine.

User response: Issue the following MVS system command to determine if the WLM application environment is defined:

D WLM,APPLENV=*environment-name*

If the environment is not defined, use one of the following methods to define it:

- Use the WLM administrative application, IWMARIN0.
- Modify and submit the sample JCL DSNTIJRW.

If the environment is defined but the status is reported as not available, determine what caused the WLM application environment to become unavailable, correct the problem, and issue the VARY WLM MVS system command to reset the status.

Severity: 8 (error)

DSNT039I   THE CALL TO THE ROUTINE WAS NOT PERFORMED

Explanation: The routine will not be called as part of validation for one or more of the following reasons:

- Optional system support, such as CICS, IMS, or MQSeries, is required.
- External setup on a remote server or workstation is required.
- A call could introduce or modify data unexpectedly in a production environment.

The message that precedes this one identifies the routine that was skipped.

System action: The routine is not called. Processing continues.

User response: No action is required.

Related information:

"DSNT027I" on page 445

"DSNT029I" on page 445

DSNT040I   *time* **DSNTRVFY ROUTINE VALIDATION SUMMARY STATUS SCHEMA SPECIFIC NAME**

Explanation: This message displays a summary of which DB2 routines completed successfully, which ones completed with warnings, and which ones failed.

time   The time at which the message was issued, in the format *hh.mm.ss*.

The message is formatted with column headings and dividing lines:

```
  STATUS   SCHEMA   SPECIFIC NAME
-- -------- -------- ---------------
```

Each line begins with a character that identifies the result:

/   Success

X   Failure

*   Warning

System action: Processing continues.

User response: No action is required.

DSNT041I   **DSNTRVFY VALIDATION FOR** *schema.routine-name* **WAS NOT PERFORMED BECAUSE THE CURRENT MODE IS NOT SUPPORTED**

Explanation: The validation for the routine was skipped because the DB2 level that is required by the routine does not match the current mode.

schema.routine-name
       The name of the routine, qualified by the name of the schema in which the routine was created.

The required mode is dependent on the stored procedure that is being validated.

System action: The routine is not validated. Processing continues.

User response: Wait until DB2 enters the required mode indicated in the message, then run the DSNTIJRV validation job again to validate the routine.

Severity: 4 (warning)

DSNT042I   **DSNTRVFY - THE COLLECTION ID USED TO BIND THE PACKAGE DOES NOT MATCH THE COLLECTION ID DEFINED IN SYSIBM.SYSROUTINES**

Explanation: The collection ID that is defined in the SYSIBM.SYSROUTINES catalog table for the routine does not match the collection ID used to bind the package.

System action: Processing continues.

User response: Determine if the collection ID that was used to bind the package is appropriate, and correct it if it is not.

Severity: 4 (warning)

DSNT043I   **DSNTRVFY - THE WLM APPLICATION ENVIRONMENT IS SHARED WITH ANOTHER
ROUTINE THAT HAS A DIFFERENT SECURITY REQUIREMENT**

Explanation: The following DB2-supplied routines requires RACF program control if the BPX.DAEMON facility
class is active. If the BPX.DAEMON is active and one of the programs loaded into the WLM address space where
these DB2-supplied routines are running is not defined to program control then an error will be returned by the
routine indicating that the operation is not permitted.

- SYSPROC.ADMIN_COMMAND_UNIX
- SYSPROC.ADMIN_JOB_CANCEL
- SYSPROC.ADMIN_JOB_FETCH
- SYSPROC.ADMIN_JOB_QUERY
- SYSPROC.ADMIN_JOB_SUBMIT
- SYSPROC.ADMIN_UTL_MONITOR

System action: Processing continues.

User response: Determine if the BPX.DAEMON is active. If BPX.DAEMON is not active, no action is required.

If it is active, ensure that all programs that have been loaded into the WLM address space where these routines are
defined are controlled by RACF. Refer to the DSNTIJRA job, which provides the commands for defining programs to
RACF program control.

Severity: 4 (warning)

DSNT044I   **DSNTRVFY - THIS ROUTINE REQUIRES A WLM ENVIRONMENT NUMTCB SETTING OF 1**

Explanation: The routine requires NUMTCB=1, but the routine was run on a WLM application environment were
the NUMTCB is not set to 1.

System action: The routine is not validated. Processing continues.

User response: Verify that the NUMTCB setting of the WLM application environment assigned to the routine is 1. If
the NUMTCB setting is not 1, either change the NUMTCB setting to 1 or assign the routine to a WLM application
environment where the NUMTCB setting is 1.

Severity: 8 (error)

DSNT045I   **DSNTRVFY - THE NUMTCB SETTING IS NOT WITHIN THE RECOMMENDED RANGE OF**
recommended-range

Explanation: The routine was assigned to a WLM application environment where the NUMTCB setting is not
within the recommended range.

recommended-range
> The recommended range for NUMTCB. In the message, this value is expressed in one of the following
> forms:
>
> *minimum* **TO** *maximum*
> > The recommended range is indicated by specific lower and upper values.
>
> *minimum* **OR MORE**
> > The recommended range is the indicated value or any greater value.

System action: Processing continues.

User response: Verify the NUMTCB setting of the WLM application environment assigned to the routine. If the
NUMTCB setting is not within the recommended range, either change the NUMTCB setting to any value within the
recommended range or assign the routine to a WLM application environment that has a NUMTCB setting that is
within the recommended range.

Severity: 4 (warning)

DSNT046I   **DSNTRVFY - THE WLM APPLICATION ENVIRONMENT IS NOT APF-AUTHORIZED**

Explanation: The routine requires an APF-authorized environment but the assigned WLM application environment is not APF-authorized.

System action: The routine is not validated. Processing continues.

User response: Determine if the WLM application environment assigned to the routine is APF-authorized. If it is not APF-authorized, either modify the WLM procedure to make it APF-authorized, or assign the routine to a WLM application environment that is APF-authorized.

Severity: 8 (error)

DSNT047I   **DSNTRVFY - THE WLM APPLICATION ENVIRONMENT IS APF-AUTHORIZED**

Explanation: The routine requires an environment that is not APF-authorized, but the assigned WLM application environment is APF-authorized.

System action: The routine is not validated. Processing continues.

User response: Determine if the WLM application environment assigned to the routine is APF-authorized. If it is APF-authorized, either modify the WLM procedure to make it not APF-authorized, or assign the routine to a WLM application environment that is not APF-authorized.

Severity: 8 (error)

DSNT048I   **DSNTRVFY - THE DD STATEMENT** *dd-name* **WAS NOT FOUND**

Explanation: The routine requires that the indicated data definition (DD) statement is defined to the WLM stored procedure address space, but the statement was not allocated.

dd-name
> The name of the data definition statement.

If a diagnostic message from the IBM Language Environment service is available, that message follows DSNT048I. This additional message provides specific information about the failure that can be used to help diagnose the cause.

System action: The routine is not validated. Processing continues.

User response: Determine whether the WLM application environment that is assigned to the routine contains the DD statement indicated by *dd-name*. If it does not contain the required DD statement, either modify the WLM procedure to include the DD statement or assign the routine to a WLM application environment that contains the DD statement.

Otherwise, if an additional message from the IBM Language Environment service is provided, use that message to diagnose and correct the problem.

Severity: 8 (error)

DSNT049I   **DSNTRVFY - JAVA VERSION (***java-version***)**

Explanation: The WLM application environment to which the Java™ routine is assigned is configured correctly for Java, and the indicated version of Java was found.

java-version
> The version of Java that is available to the WLM environment.

System action: Processing continues.

User response: No action is required.

DSNT050I   **DSNTRVFY - JAVA NOT ENABLED ON WLM ENVIRONMENT** *environment-name*

Explanation: The WLM application environment to which the Java routine is assigned is not configured correctly for Java.

environment-name
> The name of the WLM environment.

System action: The routine is not validated. Processing continues.

User response: Modify the WLM catalog procedure that starts the WLM stored procedure address space to include the JAVAENV data definition statement.

Severity: 8 (error)

Related tasks:

⇨ Setting up the environment for Java routines (DB2 Application Programming for Java)

DSNT051I   *program-name* **ACTIVE WLM SERVICE DEFINITION SERVICE DEFINITION NAME** *definition-name* **INSTALLED ON** *timestamp* **INSTALLED BY** *user-id* **INSTALLED FROM** *system-name* **NUMBER OF APPLICATION ENVIRONMENT** *environment-count*

Explanation: This message is returned by the indicated program as a response to a request to display WLM information.

program-name
> The name of the program.

definition-name
> The name of the WLM service definition.

timestamp
> The date and time when the service definition was installed, in the format *yyyy-mm-dd-hh.mm.ss.nnnnnn*.

user-id   The user ID of the service administrator who installed the service definition.

system-name
> The name of the system on which the service definition was installed.

environment-count
> The number of application environments that are defined.

This message is issued in response to either the batch program DSNTWLMB or the stored procedure DSNTWLMS (SYSPROC.DSN_WLM_APPLENV) when WLMINFO is specified.

System action: Processing continues.

User response: No action is required.

DSNT052I   *program-name* **THE AUTHID SPECIFIED,** *authorization-id*, **DOES NOT HAVE INSTALLATION SYSTEM ADMINISTRATOR AUTHORITY**

Explanation: The indicated authorization ID, which was specified in the AUTHID parameter for the indicated program, is not a DB2 installation system administrator ID. In order to be marked as system defined, stored procedures and user-defined functions need to be created or altered using DB2 installation system administrator authority.

program-name
> The name of the program.

authorization-id
> The specified authorization ID.

System action: DB2-supplied stored procedures and user-defined functions that are created or altered by this routine are not marked as system defined.

User response: If DB2-supplied stored procedures and user-defined functions at your site need to be marked as system defined, change the AUTHID parameter to a DB2 installation system administrator ID and run job DSNTIJRT again.

Severity: 4 (warning)

Related concepts:

⇨ Job DSNTIJRT (DB2 Installation and Migration)

⇨ Job DSNTIJRV (DB2 Installation and Migration)

Related reference:

⇨ Common DB2 administrative authorities (Managing Security)

DSNT053I   **DSNTRIN WAS UNABLE TO PROCESS THE IFI READS REQUEST FOR IFCID 106: RETURN CODE =** *return-code***, REASON CODE =** *reason-code***, BYTES NOT MOVED =** *bytes-remaining*

Explanation: DSNTRIN attempted to perform a synchronous read of IFCID trace record 106 but received an unexpected return code from the DB2 Instrumentation Facility Interface (IFI).

return-code
> The return code from IFI.

reason-code
> The reason code from IFI.

bytes-remaining
> The number of bytes that did not fit in the return area and still remain in the buffer.

System action: DSNTRIN terminates.

User response: To diagnose the error, refer to the return and reason codes for information about the IFCARC1 and IFCARC2 values in the instrumentation facility communications area (IFCA). Rerun job DSNTIJRT when the cause has been corrected.

Severity: 8 (error)

Related concepts:

⇨ X'E6......' codes (DB2 Codes)

⇨ Job DSNTIJRT (DB2 Installation and Migration)

Related reference:

⇨ Instrumentation facility communications area (IFCA) (DB2 Performance)

DSNT054I   *program-name* **ONE OR MORE SUPPORTING OBJECTS WERE IDENTIFIED FOR UPDATE, A REBIND MAY BE REQUIRED FOR PACKAGES THAT REFERENCE THESE OBJECTS**

Explanation: The program identified one or more supporting objects for update while in preview mode and the program can not determine whether packages will be invalidated as a result of the update.

program-name
> The name of the program.

System action: Processing continues.

User response: If the ABIND subsystem parameter is set to NO and you use the generated JCL to install or update the DB2-supplied routines, then you need to resubmit the DSNTIJRT installation job afterwards with the MODE(INSTALL-PREVIEW) option. That action identifies and generates BIND statements for any packages that were invalidated. Finally, you must process the generated BIND statements.

If the ABIND subsystem parameter is set to YES, or if you run the DSNTIJRT job without the PREVIEW option, then no action is required because DB2 rebinds any invalidated package automatically.

Severity: 4 (warning)

Related concepts:

⮕ Job DSNTIJRT (DB2 Installation and Migration)

⮕ Changes that invalidate packages (DB2 Application programming and SQL)

⮕ Automatic rebinding (DB2 Application programming and SQL)

Related reference:

⮕ AUTO BIND field (ABIND subsystem parameter) (DB2 Installation and Migration)

DSNT055I **DSNTRIN HAS BYPASSED THE CONFIGURATION CONTROL STATEMENT FOR OBSOLETE ROUTINE** *schema-name.routine-name*

Explanation: Program DSNTRIN encountered a configuration control statement for an obsolete DB2-supplied stored procedure or user-defined function.

schema-name
> The name of the table schema or creator.

routine-name
> The name of the user-defined function or stored procedure.

System action: The statement is ignored. Processing continues.

User response: Remove the configuration control statement for the obsolete routine.

Severity: 4 (warning)

DSNT089I **DSNTXTA HAS RECREATED EXPLAIN TABLE INDEX** *schema-name.index-name* **ON** *schema-name.table-name* **(***key-list***)**

Explanation: The EXPLAIN table index has been dropped and re-created in order to imbed additional keys. These actions are performed only when ALTER INDEX cannot be used to update the index.

schema-name
> The name of the table and index schema or creator.

index-name
> The name of the index.

table-name
> The name of the table.

key-list The names of the columns, in sequence, used to recreate the index.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related tasks:

⮕ Setting up the environment for Java routines (DB2 Application Programming for Java)

DSNT090I **DSNTXTA REQUEST CANCELLED BECAUSE THERE ARE NO EXPLAIN TABLES IN SCHEMA** *schema-name*

Explanation: The request was cancelled because there are no EXPLAIN tables in the specified schema.

schema-name
> The name of the table schema or creator.

System action: Processing terminates.

User response: Verify that you have specified the correct value for the DSNTXTA schema name parameter. If it is correct, then no further action is required for that schema.

Severity: 4 (warning)

DSNT091I   **DSNTXTA HAS ALTERED EXPLAIN TABLE** *schema-name*.*table-name*. **COLUMN** *column-name* **WAS**
             change-description

Explanation: The EXPLAIN table has been modified as indicated.

schema-name
             The name of the table schema or creator.

table-name
             The name of the table.

column-name
             The name of the column.

change-description
             The alteration that was made to the EXPLAIN table:

             RENAMED TO *new-name*
                     The column was renamed.

                     new-name
                             The new name of the column.

             CHANGED FROM *old-type* **TO** *new-type*
                     The data type of the column was changed.

                     old-type   The data type used for the column before the change.

                     new-type
                             The data type used for the column after the change.

             ADDED AS *data-type*
                     The column was added to the database.

                     data-type
                             The data type used for the new column.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNT092I   **DSNTXTA DID NOT COMPLETE PROCESSING OF** *schema-name*.*table-name* **BECAUSE** *reason-text*.

Explanation: The EXPLAIN table could not be updated to the required format for this version of DB2 for the indicated reason.

schema-name
             The name of the table schema or creator.

table-name
             The name of the table.

reason-text
             The reason for that the table could not be updated:

             IT CANNOT BE ALTERED TO A 10-COLUMN TABLE
                     A 9-column DSN_QUERY_TABLE exists in the indicated schema. A 9-column DSN_QUERY_TABLE
                     cannot be altered to a 10-column DSN_QUERY_TABLE because the new column,
                     GROUP_MEMBER, must reside at position 8.

             OF AN UNEXPECTED NAME OR TYPE FOR COLUMN NUMBER *column-number*. **EXPECTED:**
             expected-format. **FOUND:** *actual-format*.
                     While analyzing a table, DSNTXTA encountered a column that has an unexpected name or attribute

                     column-number
                             The location of the unexpected name or attribute.

                     expected-format
                             The format that is required for the name or attribute.

DSNT093I

> *actual-format*
>> The format that is used for the name or attribute.

> **COLUMN** *column-name* **CANNOT BE ADDED PRIOR TO NEW-FUNCTION MODE**
>> DB2 is in a mode prior to new-function-mode. The column cannot be added until DB2 enters new-function-mode

> *column-name*
>> The name of the column.

System action: Processing of the table that is identified by *schema-name.table-name* is discontinued. Processing continues with the next candidate table, if one exists.

User response: Take the action that is indicated for the *reason-text* value:

IT CANNOT BE ALTERED TO A 10-COLUMN TABLE
> Use REXX exec DSNTXTB to generate statements for migrating the 9-column DSN_QUERY_TABLE to a 10-column DSN_QUERY_TABLE. See job DSNTIJXB for an example of how to execute DSNTXTB, and job DSNTIJXC for an example of how to process the generated statements after you have reviewed them and made any required adjustments.

OF AN UNEXPECTED NAME OR TYPE FOR COLUMN NUMBER *column-number*. **EXPECTED:** *expected-format*. **FOUND:** *actual-format*.
> Either the table is not an EXPLAIN table or it is not in the expected format.

> 1. Query SYSIBM.SYSCOLUMNS, and inspect the column names and attributes of the table. For example, use the following query:

```
SELECT NAME, COLNO,
   CASE COLTYPE
     WHEN 'TIMESTMP' THEN 'TIMESTAMP'
     ELSE COLTYPE
   END,
  LENGTH ,LENGTH2
  FROM SYSIBM.SYSCOLUMNS
  WHERE TBCREATOR = 'schema-name'
  AND TBNAME = 'table-name'
  ORDER BY COLNO;
```

> Use the information about SYSIBM.SYSCOLUMNS to help you locate the structural problem.

> 2. If you suspect that the table might not be an EXPLAIN table, review the list of EXPLAIN tables. Compare your table to the expected table of the same name. If the table is not an EXPLAIN table, consider renaming it.

> 3. If the table is an EXPLAIN table, it is not in the expected format and you must correct the discrepancy. You can either:
>   - Alter the table to conform to the format described for the table.
>   - Rename the existing table, create a replacement table with the appropriate name in the expected format, and copy the data from the original table to the replacement. When you have confirmed the contents of the replacement table, drop the original table.

COLUMN *column-name* **CANNOT BE ADDED PRIOR TO NEW-FUNCTION MODE**
> Rerun this job when DB2 enters new-function mode.

Severity: 4 (warning)

DSNT093I   **DSNTXTA PROCESSING COMPLETE. SCHEMAS EXAMINED:** *total-schemas*; **TABLES EXAMINED:** *total-tables*; **TABLES ALTERED:** *total-altered*; **INDEXES EXAMINED:** *total-indexes*; **INDEXES RECREATED:** *total-recreated*; **TOTAL WARNINGS:** *total-warnings*; **TOTAL ERRORS:** *total-errors*

Explanation: All EXPLAIN tables in the specified schema or schemas are now in the current format, except as noted by previous warning messages. This message reports summary statistics for the DSNTXTA job.

total-schemas
> The total number of schemas that were processed.

total-tables
> The total number of tables that were processed.

total-altered
  The total number of tables that were altered.

total-indexes
  The total number of indexes that were processed.

total-recreated
  The total number of indexes that were re-created.

total-warnings
  The total number of warnings that were issued.

total-errors
  The total number of errors that were issued.

System action: Processing terminates.

User response: Review the summary statistics and other DSNT messages that were generated during processing. These messages identify tables that were not updated.

Severity: 0

DSNT094I   **DSNTXTB REQUEST CANCELLED BECAUSE THERE ARE NO EBCDIC EXPLAIN TABLES IN SCHEMA** *schema-name*

Explanation: The request was cancelled because there are no EBCDIC EXPLAIN tables in the specified schema.

schema-name
  The name of the table schema or creator.

System action: Processing terminates.

User response: Verify that you have specified the correct value for the DSNTXTB schema name parameter. If it is correct, then no further action is required for that schema.

Severity: 4 (warning)

DSNT095I   **DSNTXTB REQUEST CANCELLED BECAUSE TABLE** *schema-name*.*table-name* **IS NOT IN THE REQUIRED FORMAT AT COLUMN NUMBER** *column-number*. **EXPECTED:** *expected-format*. **FOUND:** *actual-format*.

Explanation: While analyzing a table, DSNTXTB encountered a column that has an unexpected name or attribute.

schema-name
  The name of the table schema or creator.

table-name
  The name of the table.

column-number
  The location of the unexpected name or attribute.

expected-format
  The format that is required for the name or attribute.

actual-format
  The format that is used for the name or attribute.

System action: Processing terminates.

User response:

1. Query SYSIBM.SYSCOLUMNS, and inspect the column names and attributes of the table. For example, use the following query:
   ```
   SELECT NAME, COLNO,
      CASE COLTYPE
        WHEN 'TIMESTMP' THEN 'TIMESTAMP'
        ELSE COLTYPE
      END,
    LENGTH ,LENGTH2
   ```

```
FROM SYSIBM.SYSCOLUMNS
WHERE TBCREATOR = 'schema-name'
AND TBNAME = 'table-name'
ORDER BY COLNO;
```

Use the information about SYSIBM.SYSCOLUMNS to help you locate the structural problem.

2. If you suspect that the table might not be an EXPLAIN table, review the list of EXPLAIN tables. Compare your table to the expected table of the same name. If the table is not an EXPLAIN table, consider renaming it.

3. If the table is an EXPLAIN table, it is not in the expected format and you must correct the discrepancy. You can either:

   • Alter the table to conform to the format described for the table.

   • Rename the existing table, create a replacement table with the appropriate name in the expected format, and copy the data from the original table to the replacement. When you have confirmed the contents of the replacement table, drop the original table.

Severity: 8 (error)

DSNT096I   **WARNING: THE TARGET DATABASE,** *database-name*, **IS NOT A UNICODE DATABASE**

Explanation: The target database specified for new Unicode explain tables already exists in DB2 and is not a Unicode database by default.

database-name
        The name of the database.

System action: Processing continues.

User response: No action is required.

If you prefer to have the new explain tables created in a Unicode database, change the appropriate target database parameter to specify either an existing Unicode database or the name of a new one for which DSNTXTB should generate a CREATE statement. Then rerun DSNTXTB.

Severity: 0

DSNT097I   **DSNTXTB REQUEST CANCELLED BECAUSE THE SPECIFIED TARGET** *required-type* **TABLE SPACE,** *database-name*.*tablespace-name*, **IS AN EXISTING** *actual-type* **TABLE SPACE**

Explanation: The indicated table space already exists in DB2 as a different type of table space than is required.

required-type
        The type of table space that was specified as a DSNTXTB input parameter: 4KB page table space, 8KB page table space, 16KB page table space, 8KB page LOB TS, or 32KB page LOB TS.

database-name.*tablespace-name*
        The table space that was the target of the DSNTXTB request.

actual-type
        The type of the existing table space that is identified by *database-name*.*tablespace-name*: 4KB page table space, 8KB page table space, 16KB page table space, 8KB page LOB TS, 32KB page LOB TS, or NON-UNICODE.

System action: Processing terminates.

User response: Correct the DSNTXTB input parameter indicated by *required-type* so that it specifies one of the following names:

• An existing, conforming Unicode-encoded table space.

• A new table space for which DSNTXTB should generate a CREATE statement with the required attributes.

Severity: 8 (error)

DSNT098I   DSNTXTB HAS GENERATED STATEMENTS TO CREATE UNICODE EXPLAIN TABLES FOR THE SPECIFIED SCHEMA IN AN EXISTING TARGET TABLE SPACE, *database-name*.*tablespace-name*

Explanation: The indicated table space already exists in DB2. If the control statements that were generated by this execution of DSNTXTB are processed, this table space will be used as the target table space for new Unicode explain tables.

database-name
> The name of the database.

tablespace-name
> The name of the table space.

System action: Processing continues.

User response: No action is required.

If you prefer to have the new explain tables created in a different table space, change the appropriate target table space parameter to specify either a different existing Unicode table space or the name of a new one for which DSNTXTB should generate a CREATE statement. Then rerun DSNTXTB.

Severity: 0

DSNT099I   DSNTXTB HAS GENERATED STATEMENTS FOR MIGRATING TABLE *schema-name*.*table-name* TO UNICODE

Explanation: An EBCDIC-encoded explain table was found in the specified schema. One instance of this message is issued for each EBCDIC-encoded explain table.

schema-name
> The name of the table schema or creator.

table-name
> The name of the table.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNT101I   A VSAM ERROR HAS BEEN ENCOUNTERED WHILE CSECT DSNTSTR2 WAS READING THE DB2 BSDS VSAM RETURN CODE = *return-code*, VSAM REASON CODE = *reason-code*

Explanation: CSECT DSNTSTR2 attempted to read the DB2 bootstrap data set (BSDS), and a VSAM error has occurred. *return-code* is a 1-byte VSAM return code and *reason-code* is a 1-byte VSAM reason code.

System action: The start up process continues. However, any references to password-protected system databases will cause a 'resource unavailable' error. Any agent-related requests for subsystem database resources will result in an SQLCODE -904. Any requests that are not agent-related for unavailable resources will result in the following messages:

DSNI001I	DSNI002I	DSNI003I	DSNI004I
DSNI005I	DSNI006I	DSNT408I	DSNT414I
DSNT415I	DSNT416I	DSNT417I	DSNT500I

Abend X'04E' with reason code X'00E30010' is also produced. However, the error results in a retry. DB2 does not terminate, but a dump is produced.

Operator response: Notify the system programmer.

System programmer response: A VSAM error occurred while reading the BSDS record with key '200000001'. Either the BSDS record does not exist or an I/O error occurred (indicating that the record is bad). Further information may be obtained from a VSAM diagnostics manual.

Problem determination: Collect the following diagnostic items:

DSNT102I

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.

.

Related concepts:

▶ Management of the bootstrap data set (DB2 Administration Guide)

Related information:

▶ Recovering from BSDS failures (DB2 Administration Guide)

▶ -904 (DB2 Codes)

▶ 00E30010 (DB2 Codes)

DSNT102I   DSNTSTR2 - AN INVALID DATABASE SERVICES BOOTSTRAP RECORD HAS BEEN ENCOUNTERED

Explanation: CSECT DSNTSTR2 attempted to read the DB2 bootstrap data set (BSDS), and an invalid record was read by the recovery log manager (RLM) subcomponent. This error is an internal DB2 error.

The invalid record is a database services password record with key '200000001.'

System action: DB2 start-up processing continues.

However, any references to DB2 password-protected system databases cause a resource unavailable condition. Any agent-related requests for unavailable DB2 subsystem database resources result in an SQLCODE -904. Any requests that are not agent-related for unavailable resources result in one or more of the following messages:

- DSNI001I
- DSNI002I
- DSNI003I
- DSNI004I
- DSNI005I
- DSNI006I
- DSNT408I
- DSNT414I
- DSNT415I
- DSNT416I
- DSNT417I
- DSNT500I

Abend X'04E' with reason code X'00E30010' also occurs with message DSNT102I. As a result of this error, DB2 tries the operation again. DB2 does not terminate, but produces a dump.

Operator response: Notify the system programmer.

System programmer response: Collect the information that is identified in the problem determination section and contact IBM Software Support.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.
-

.

Related tasks:

⏩ Collecting data (Collecting data)

Related reference:

⏩ DB2 abend completion codes (X'04E' and X'04F') (DB2 Codes)

Related information:

"DSNI001I" on page 213

"DSNI002I" on page 213

"DSNI003I" on page 213

"DSNI004I" on page 214

"DSNI005I" on page 214

"DSNI006I" on page 215

"DSNT408I" on page 535

"DSNT414I" on page 537

"DSNT415I" on page 537

"DSNT416I" on page 537

"DSNT417I" on page 538

"DSNT500I" on page 555

⏩ -904 (DB2 Codes)

⏩ 00E30010 (DB2 Codes)

DSNT103I   THE CURRENTLY LOADED DSN6SPRM IS NOT THE MVS/XA VERSION

Explanation: A -START DB2 PARM=xxxxxxxx was issued where 'xxxxxxxx' is the version of the system parameter CSECTs. DB2 executes in a MVS/XA environment and will not execute in any other environment.

System action: The message DSNV086E is generated with a reason code of X'00E30008'. DB2 is terminated with abend code X'04F'.

Operator response: Notify the system programmer.

System programmer response: Restart DB2 with a -START DB2 PARM= command, where the parameter block that follows the 'PARM=' parameter specifies an MVS/XA version of the DB2 subsystem parameters.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

.

DSNT104I   THE CURRENTLY LOADED DSN6SPRM IS NOT THE MVS/370 VERSION

Explanation: A -START DB2 PARM=xxxxxxxx was issued, where 'xxxxxxxx' is the version of the system parameter CSECTs. DB2 executes in an MVS/370 environment and does not execute in any other environment.

System action: Message DSNV086E is generated with reason code X'00E30009'. DB2 is terminated with abend code X'04F'.

Operator response: Notify the system programmer.

System programmer response: Restart DB2 with a -START DB2 PARM= command, where the parameter block that follows the PARM= parameter specifies an MVS/370 version of the DB2 subsystem parameters.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

.

DSNT105I   THE LOADED DSN6SPRM CSECT IS OBSOLETE. PLEASE REASSEMBLE THE DSN6SPRM CSECT AND RELINKEDIT THE DSNZPARM LOAD MODULE.

Explanation: DSNTSTR2 loaded an out-of-date system parameter CSECT DSN6SPRM. This error can only occur if code maintenance has been applied and CSECT DSN6SPRM has not been reassembled. This CSECT is contained in the load module specified in the 'PARM=' parameter of the -START DB2 command.

System action: The message DSNV086E is generated with a reason code of X'00E30008'. DB2 is terminated with abend code X'04F'.

Operator response: Notify the system programmer.

System programmer response: Reassemble DSN6SPRM, and relink-edit the DSNZPARM load module. This is done by resubmitting the installation job stream, DSNTIJUZ, that built the load module that is named following the 'PARM=' parameter of the -START DB2 command.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

.

DSNT106I   THE LOADED DSN6SYSP CSECT IS OBSOLETE. PLEASE REASSEMBLE THE DSN6SYSP CSECT AND RELINKEDIT THE DSNZPARM LOAD MODULE.

Explanation: DSNTSTR2 loaded an out-of-date system parameter CSECT DSN6SYSP. This error can only occur if code maintenance has been applied and CSECT DSN6SYSP has not been reassembled. This CSECT is contained in the load module specified in the 'PARM=' parameter of the -START DB2 command.

System action: The message DSNV086E is generated with a reason code of X'00E30008'. DB2 is terminated with abend code X'04F'.

Operator response: Notify the system programmer.

System programmer response: Reassemble DSN6SYSP, and relink-edit the DSNZPARM load module. This is done by resubmitting the installation job stream, DSNTIJUZ, that built the load module that is named following the 'PARM=' parameter of the -START DB2 command.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

DSNT107I   **INVALID AMODE OR RMODE ATTRIBUTE FOUND FOR LOAD MODULE** *module-name*

Explanation: During DB2 startup, the service controller establishes the addressability to the DB2 database services functional recovery and the emergency shutdown routines. Two load modules, DSNTF, and DSNTLCLE, are loaded into CSA. In the MVS/XA environment, DSNTLCLE must be link-edited with attributes AMODE(24) and RMODE(ANY). At least one of the modules did not have the correct addressing attribute when it was loaded. 'module-name' in the error message is the name of the first load module that has an invalid addressing mode.

System action: DB2 database services start up is terminated.

Operator response: Notify the system programmer.

System programmer response: Verify that all installation and maintenance activities against these modules were executed using SMP. Verify that the JCLIN for SMP includes the correct AMODE and RMODE control statements, and that the linkage editor provided with the Data Facility Product (DFP) was invoked by SMP for the link edits. Correct the procedure or JCL which caused the error and rerun the necessary SMP jobs.

Problem determination: A DB2 dump was requested to an MVS/XA SYS1.DUMP data set.

The MVS service aid AMBLIST provides linkage editor module attributes in the summary section of the LISTLOAD function output.

DSNT108I   **DB2 HAS DETECTED A DIFFERENCE BETWEEN THE RECORDED SYSTEM CCSIDS AND THE CCSIDS SPECIFIED BY DSNHDECP**

Explanation: During DB2 startup, DB2 detected a difference between the CCSIDs recorded in the BSDS and the CCSIDs specified by the version of DSNHDECP, or the user-specified application defaults module, that was loaded during startup. DB2 startup terminated to avoid potential data corruption issues.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: Determine the reason for the mismatch between the CCSIDs in the BSDS and the CCSIDs specified by the *dsnhdecp* module that was loaded during startup. Possible reasons for a mismatch include that the procedure used to start DB2 might have incorrect library concatenations, or, the copy of the *dsnhdecp* module that was loaded was incorrectly copied to the libraries specified by the DB2 startup procedure.

Correct the mismatch. If IBM support directs you, you can use the change log inventory (DSNJU003) utility with the DELETE CCSIDS clause to delete the CCSIDs that are specified in the in the BSDS. **Important:** Do not attempt to delete the CCSIDs in the BSDS unless IBM support instructs you.

Related concepts:

➥ Application defaults parameters (DB2 Installation and Migration)

➥ Application defaults module name at start (DB2 Administration Guide)

Related tasks:

➥ Specifying CCSIDs in DB2 (DB2 Internationalization Guide)

DSNT109I   **DB2 HAS DETECTED AN INVALID CCSID IN THE CCSID(S) SPECIFIED FOR ENCODING SCHEME** *encoding-scheme* **BY DSNHDECP**

Explanation: The version of DSNHDECP, or the user-specified application defaults module, that was loaded during DB2 startup has invalid CCSIDs for the *encoding-scheme* encoding scheme. DB2 startup terminated to avoid potential data corruption issues.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer.

System programmer response: If the *dsnhdecp* value for MIXED is NO, the specified CCSIDs must be SBCS CCSIDs. You can find these CCSIDs in the table entitled "Single-byte coded character set identifiers (CCSIDs)".

If the *dsnhdecp* value for MIXED is YES, then you must select ASCII MIXED from the table entitled "ASCII double-byte coded character set identifiers (CCSIDs)", and you must select EBCDIC MIXED CCSIDs from the table entitled "EBCDIC double-byte coded character set identifiers (CCSIDs)".

Unicode CCSID values are pre-filled and you should not change them.

Related concepts:

➥ Application defaults parameters (DB2 Installation and Migration)

➥ Application defaults module name at start (DB2 Administration Guide)

➥ Character conversion (DB2 SQL)

Related tasks:

➥ Specifying CCSIDs in DB2 (DB2 Internationalization Guide)

DSNT200I   *type bind-type* **FOR PLAN** *plan-id* **SUCCESSFUL**

Explanation: A BIND, REBIND or FREE PLAN subcommand has completed successfully.

type Indicates that a test bind was in progress. A test bind means that no plan name was specified. The token is omitted if a normal BIND PLAN was processed.

bind-type
        Type of bind subcommand: BIND, REBIND, or FREE.

plan-id Name of application plan specified in the subcommand.

System action: The plan was successfully bound, rebound, or freed.

DSNT201I   *type bind-type* **FOR PLAN** *plan-id* **NOT SUCCESSFUL**

Explanation: Errors have been detected during the processing of the indicated BIND, REBIND, or FREE PLAN subcommand.

type Indicates that a test bind was in progress. A test bind means that no plan name was specified. The token is omitted if a normal BIND PLAN was processed.

bind-type
        Type of bind subcommand: BIND, REBIND, or FREE.

plan-id Name of application plan specified in the subcommand.

System action: The BIND, REBIND, or FREE PLAN was not performed.

System programmer response: See other messages issued to determine the error that caused the bind subcommand to terminate unsuccessfully.

DSNT202I   **MESSAGE LIMIT EXCEEDED**

Explanation: A fixed amount of space exists in which to generate bind subcommand messages. This message says that the bind subcommand generated more messages than will fit in that space.

System action: For bind subcommands that affect a single plan, the bind subcommand process continues. However, no other messages are generated except for message DSNT200I or DSNT201I. For bind subcommands that affect more than one plan, the bind subcommand process terminates and no other messages are generated.

System programmer response: The 'FLAG' parameter of the bind subcommand can be used to limit the types of messages that are generated.

DSNT203I   *bind-type* **WARNING FOR** *object-type object-name* **DBRMS ARE CONVERTED INTO PACKAGES**
BEFORE *bind-typeobject-typeobject-name***.**

Explanation: An attempt was made to bind DBRMs directly into a plan or to rebind a plan with DBRMs that were
already bound.

bind-type
The type of BIND subcommand, such as BIND or REBIND.

object-type
The type of object, such as PLAN.

object-name
The name of the application plan.

This message can be caused by using the MEMBER option when binding a plan or by rebinding a plan with DBRMs
that were bound in an earlier version of DB2.

System action: The bind process continues and the conversion is completed.

Operator response: No action is required.

Severity: 4 (warning)

DSNT204I   *bind-type* **WARNING FOR** *object-type object-name***,** *option* **IS** *option-status***,** *substitute-option* **WILL BE**
USED.

Explanation: A BIND subcommand was issued with an invalid option.

bind-type
The type of BIND subcommand, such as REBIND.

object-type
The type of object, such as PLAN or PACKAGE.

object-name
The name of the application plan.

option   The unsupported BIND option.

option-status
The problem with *option*. Possible values are:

INVALID
The BIND option is not valid in the context where it is used.

NOT USED
The BIND option syntax is valid, but the option has no effect.

substitute-option
The BIND option that was used instead of *option*.

This message can be caused by the use of a deprecated option.

System action: The bind process continues, using the option that is indicated by *substitute-option*.

Operator response: No action is required.

Severity: 4 (warning)

Related reference:

➡ BIND and REBIND options (DB2 Commands)

DSNT205I   *name* **IS A DUPLICATE** *name-type***, IT WILL BE USED ONLY ONCE**

Explanation: Name 'name' has been specified more than once in a list of names.

name   Name that was specified more than once.

name-type
> Indicates the type of list in which the name was duplicated.
> - MEMBER NAME - List of DBRM Members
> - PLAN-ID - List of plan IDs
> - SYSTEM NAME - List of system connection names on a DISABLE or ENABLE list.
> - PACKAGE NAME - List of package IDs

System action: Duplicates are removed from 'name-type' list and the bind process continues.

System programmer response: Verify the list that contains the duplicate. If the error is only in entering a name incorrectly, which happens to duplicate another entry, enter the name correctly and reissue the bind statement. If the entry is really a duplicate, no action is required.

DSNT206I   *bind-type* **WARNING FOR** *object-type* *object-name* **DEFAULT COLLECTION ID: DSN_DEFAULT_COLLID_***object-name* **IS USED IN THE PACKAGE LIST, POSSIBLE DBRMS ORIGINALLY BOUND WITH THE** *bind-type* **HAVE BEEN CONVERTED INTO PACKAGES DURING EXECUTING THE** *object-type* **PRIOR TO** *bind-type*

Explanation: The default collection ID (DSN_DEFAULT_COLLID_*object-name*) has been detected in the package list of the plan during the rebind process.

The default collection ID is the result of an automatic conversion from DBRMs to packages within a plan. This message is issued when such a plan is rebound again. This condition typically happens when the plan contains DBRMs that were directly bound, but that the DBRMs have been converted into packages. This conversion usually happens during an attempt to use the plan in a newer release of DB2 than the one in which it was created.

bind-type
> The type of BIND subcommand, such as REBIND.

object-type
> The type of object, such as PLAN.

object-name
> The name of the application plan.

System action: The rebind process continues and the plan is rebound with the package list.

Operator response: No action is required.

Severity: 4 (warning)

DSNT207I   **PROGRAM** *program-name* **WITH MARK** *release-dependency-mark* **FAILED BECAUSE IT DEPENDS ON FUNCTIONS OF THE RELEASE FROM WHICH FALL BACK HAS OCCURRED**

Explanation: Program 'program-name' depends on a function of DB2 that is not supported by the current active release.

program-name
> Name of the application program.

release-dependency-mark
> A one-character mark showing the oldest DB2 release supporting this program.

System action: The bind operation for this plan or package is not performed.

System programmer response: Warn users not to use plans or packages containing this program until the DB2 subsystem has been remigrated to the newer release.

User response: The program cannot be used until the DB2 subsystem is remigrated to the newer release. Notify the system programmer.

DSNT208I PROGRAM *program-name* **PRECOMPILED WITH INCORRECT LEVEL FOR THIS RELEASE**

Explanation: Program 'program-name' was precompiled under a release not supported by the current level of DB2, or the contents of the DBRM have been modified after the precompilation phase.

System action: The BIND operation was not performed.

User response: Precompile the named program again using the current precompiler. Reissue the BIND command.

Problem determination: If the application program was precompiled at an appropriate release, and the problem persists, collect the following diagnostic items:

- Hexadecimal print of the first record of the failing DBRM.
- Listing from the precompile job that generated the DBRM.
- Output from the BIND attempt.

.

DSNT209I *bind-type* **FOR PLAN** *plan-id* **WITH MARK** *release-dependency-mark* **FAILED BECAUSE PLAN DEPENDS ON FUNCTIONS OF THE RELEASE FROM WHICH FALL BACK HAS OCCURRED.**

Explanation: Plan 'plan-id' depends on a function of DB2 which is not supported by the currently active release.

bind-type
> REBIND

plan-id Name of the application plan

release-dependency-mark
> A one-character mark showing the oldest release of DB2 which can support this plan. The release dependency mark for the plan is kept in the IBMREQD columns in the DB2 catalog in SYSIBM.SYSPLAN.

System action: The REBIND operation for this plan is not performed.

Operator response: Notify the system programmer.

System programmer response: Warn users not to use the plan until the DB2 subsystem has been remigrated to the newer release.

User response: The plan cannot be used until the DB2 subsystem is remigrated to the newer release.

DSNT210I *bind-type* **AUTHORIZATION ERROR USING** *authorization-id* **AUTHORITY PLAN=***plan-id* **PRIVILEGE=***privilege*

Explanation: The indicated authorization ID does not have the indicated privilege and therefore cannot invoke the indicated BIND subcommand against the indicated application plan. If the access control authorization exit is active and the AUTHEXIT_CHECK system parameter is set to DB2, this error might occur if ACEE cannot be created for the indicated authorization ID.

bind-type
> Type of BIND subcommand: BIND, REBIND, FREE.

authorization ID
> Authorization ID of the plan owner. **Note:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

plan-id Application plan specified in the BIND subcommand.

privilege
> Missing privilege: BIND, BINDADD

System action: The BIND, REBIND, or FREE operation for 'plan-id' is not performed.

System programmer response: If the indicated privilege is BINDADD, then the privilege to invoke the BIND subcommand with the ACTION(ADD) option must be granted to the indicated authorization ID. If the indicated privilege is BIND, the privilege to invoke a BIND subcommand against the indicated application plan must be granted to the indicated authorization ID.

If you use the access control authorization exit, ensure that the indicated authorization ID is defined in RACF and granted the indicated privilege in RACF.

DSNT211I   **REBIND FOR PLAN** *plan-id* **FAILED BECAUSE IBMREQD OF** *ibmreqd* **IS INVALID**

Explanation: The IBMREQD column of the SYSIBM.SYSPLAN catalog table for the named plan contains an unrecognizable character.

System action: The REBIND operation for the plan was not performed.

System programmer response: You must do a BIND ACTION(REPLACE) for this plan.

DSNT212I   *bind-type* **AUTHORIZATION ID** *auth-id* **IN THE OWNER KEYWORD IS INVALID**

Explanation: The authorization ID specified in the OWNER keyword of the BIND or REBIND subcommand is invalid. Specifying an owner that is not the primary authorization ID or one of the secondary IDs of the process requires additional authority.

bind-type
> Type of bind subcommand: BIND or REBIND

auth-id   The authorization ID specified as owner.

> If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: `ROLE: role-name`.

System action: The BIND or REBIND subcommand is not executed.

System programmer response: Rerun the subcommand and specify a valid authorization ID for the OWNER keyword. If necessary, add the required authorization ID for this user to the security system.

Related concepts:

➦ The bind process (DB2 Commands)

Related information:

➦ DB2 and related commands (DB2 Commands)

DSNT213I   **THE DCB BLOCK SIZE OF THE INPUT DBRM LIBRARY IS INVALID**

Explanation: An invalid DCB block size is specified for the DBRM library of the BIND subcommand.

System programmer response: Correct the DCB block size of the DBRM library, and reissue the BIND subcommand.

DSNT214I   **I/O ERROR ON DBRM LIBRARY DBRM MEMBER=** *dbrm-name* **DBRM LIBRARY=** *dbrm-library-name* *xxxxxx*

Explanation: An I/O error occurred during execution of a BIND subcommand when the database request module (DBRM) was read.

dbrm-name
> The name of the DBRM or zFS file that is associated with the error.

dbrm-library-name
> The partitioned data set or the absolute path of a zFS file that is associated with the error.

xxxxxx   A portion of the message text that was generated by the SYNADAF macro instruction to describe the permanent I/O error. This text might contain information such as the job name, step name, unit address, device type, DD name, the operation that was attempted, and a description of the error.

System action: No application plan is created.

System programmer response: Determine if an incorrect file name was specified in the LIBRARY parameter of the BIND subcommand, or an incorrect data set name was specified for the DBRMLIB DD statement. Check correctly specified data sets for I/O errors or modified DCB attributes.

DSNT215I   *csect-name bind-type bind-subtype* **FOR PACKAGE =** *package-name* **FAILED BECAUSE AT LEAST ONE BIND OPTION IS NOT ALLOWED FOR A** *package-type* **PACKAGE**

Explanation: The BIND or REBIND subcommand has been submitted with an invalid combination of bind options. The message tokens provides the following information:

csect-name
        The name of the control section that issued the message.

bind-type
        Type of bind subcommand: BIND or REBIND

bind-subtype
        Type of bind subcommand: COPY, DEPLOY, or BLANK.

package-name
        Name of the package in the format: *location-id.collection-id.package-id(version-id).package-type*

package-type
        Type of package:
        • PROCEDURE
        • NATIVE SQL PL PROCEDURE
        • TRIGGER
        • FUNCTION

        .

System action: The bind fails.

User response: Remove the incompatible bind options and reissue the command.

Related reference:

➥ BIND PACKAGE (DSN) (DB2 Commands)

➥ BIND PLAN (DSN) (DB2 Commands)

➥ REBIND PACKAGE (DSN) (DB2 Commands)

➥ REBIND PLAN (DSN) (DB2 Commands)

Related information:

➥ DB2 and related commands (DB2 Commands)

DSNT216I   **REBIND FOR PACKAGE =** *package-name* **FAILED BECAUSE AT LEAST ONE BIND OPTION SPECIFIED CANNOT BE CHANGED IN CONJUNCTION WITH PLANMGMT.**

Explanation: An attempt was made to rebind a package using the PLANMGMT option, but a new value was indicated for a bind option that cannot be changed from its previous value.

package-name
        The name of the package.

When PLANMGMT is specified, the following bind options cannot be changed from their previous values.
• OWNER
• QUALIFIER
• ENABLE
• DISABLE
• PATH
• PATHDEFAULT
• IMMEDWRITE

System action: Command processing terminates.

Operator response: Remove the conflicting REBIND options and re-invoke the command.

Related concepts:

⬏ Plan management polices (DB2 Performance)

DSNT217I **REBIND SWITCH FOR PACKAGE =** *package_name* **FAILED BECAUSE THE PREVIOUS OR ORIGINAL COPY DOES NOT EXIST.**

Explanation: An attempt was made to revert to a non-existent PREVIOUS or ORIGINAL copy of a package.

System action: Command processing terminates.

DSNT218I **REBIND FOR PACKAGE =** *package_name* **FAILED BECAUSE SWITCH CANNOT BE SPECIFIED WITH ANY OTHER REBIND OPTIONS**

Explanation: When attempting to SWITCH to a PREVIOUS or ORIGINAL copy of a package, no other bind options can be specified.

System action: Command processing terminates.

Operator response: Remove the conflicting bind options and re-invoke the command.

DSNT219I **REBIND FOR PACKAGE =** *package_name* **SUCCEEDED BUT A PREVIOUS AND/OR ORIGINAL PACKAGE COPY WAS NOT CREATED/REPLACED BECAUSE THE CURRENT PACKAGE COPY WAS INVALID.**

Explanation: When a REBIND PACKAGE command is invoked on an invalid package, any PLANMGMT= BASIC or PLANMGMT=EXTENDED options are ignored. DB2 rebinds the package successfully. However, any previously existing or original copies are not modified.

System action: Command processing terminates.

Operator response: No action is required.

DSNT220I **BIND ADD ERROR USING** *auth-id* **AUTHORITY PLAN** *plan-id* **ALREADY EXISTS**

Explanation: An attempt was made to use the BIND subcommand with the 'ACTION(ADD)' option that specified a 'plan-id' that already exists.

auth-id Authorization ID of the owner of the plan. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

plan-id Application plan specified in the BIND subcommand.

System action: No application plan is created.

System programmer response: Check the SYSIBM.SYSPLAN catalog table for names of existing application plans. Reinvoke the BIND subcommand with a 'plan-id' that is not in use.

DSNT221I *bind-type* **ERROR USING** *auth-id* **AUTHORITY PLAN** *plan-id* **DOES NOT EXIST**

Explanation: The indicated bind subcommand was issued against an application plan that does not exist. The individual variable fields contain:

bind-type
     Type of bind subcommand: REBIND or FREE.

auth-id Authorization ID of the plan owner.

plan-id Application plan specified in the BIND subcommand.

System programmer response: Check the SYSPLAN catalog table for the correct 'plan-id' to use.

DSNT222I   *csect-name bind-type* **WARNING FOR** *object-type* = *object-name* **USE OF OPTHINT RESULTS IN** *count-1* **STATEMENTS WHERE OPTHINT FULLY APPLIED** *count-2* **STATEMENTS WHERE OPTHINT NOT APPLIED OR PARTIALLY APPLIED** *count-3* **STATEMENTS WHERE OPTHINT IS NOT FOUND**

Explanation: The OPTHINT option of Bind or Rebind was specified. This message gives counts of the number of statements where a hint was fully applied, not fully applied, or not found.

csect-name
> The name of the csect from which the message was issued.

bind-type
> Type of BIND subcommand: BIND or REBIND.

object-type
> PLAN or PACKAGE

object-name
> If *object-type* is PLAN, *object-name* is the name of the plan.
>
> If *object-type* is PACKAGE, *object-name* is the name of the package in the format 'location-id.collection-id.package-id'.

count 1   The number of statements where hints were found, and all hints found were successfully applied.

count 2   The number of statements where hints were found and at least one of those hints could not be successfully applied.

count 3   The number of statements where there were no matching optimization hints found.

System action: Processing continues normally.

Programmer response: Examine the counts to ensure that the specified optimization hints were applied to the intended statements. Refer to SQL codes +394 and +395 for more information about individual statements.

DSNT223I   *csect-name bind-type* **WARNING FOR** *object-type* = *object-name* **USE OF OPTHINT IS DISALLOWED BY A DB2 SUBSYSTEM PARAMETER. THE DEFAULT OPTHINT VALUE IS USED.**

Explanation: The DB2 subsystem parameters have not been configured to enable the use of OPTHINT in the *bind-type* subcommand. The default OPTHINT value of all blanks is used.

bind-type
> Type of BIND subcommand: BIND or REBIND.

object-type
> PLAN or PACKAGE

object-name
> If *object-type* is PLAN, *object-name* is the name of the plan.
>
> If *object-type* is PACKAGE, *object-name* is the name of the package in the format 'location-id.collection-id.package-id'.

System action: Processing continues normally using the default OPTHINT value.

System programmer response: Enable the use of OPTHINT by changing the value of OPTIMIZATION HINTS on the DB2 Installation panel, DSNTIP8.

If, after further consideration, you do not want to use OPTHINT, remove OPTHINT from the *bind-type* subcommand or set the value to all blanks.

DSNT224I   *csect-name bind-type* **ERROR FOR** *object-type object-name* **CCSID =** *ccsid* **IS INVALID FOR ENCODING SCHEME =** *scheme*

Explanation: The system does not have a valid CCSID defined for the specified encoding scheme, or the specified CCSID is not defined on the system.

bind-type
> Type of bind subcommand: BIND or REBIND.

object-type
> PLAN or PACKAGE.

object-name
> If object-type is PLAN, object-name is the name of the application. Otherwise, it is the name of the package in 'location.collection.package' format.

scheme The encoding scheme specified in the ENCODING bind option on the subcommand or defaulted from the system default Application Encoding Scheme. If *scheme* is UNKNOWN, then the CCSID value specified in the ENCODING bind option or defaulted from the system default Application Encoding Scheme is not defined, and therefore the encoding scheme is not known.

ccsid The numeric CCSID corresponding to the specified *ccsid-type*.

System action: The *bind-type* operation is not performed

System programmer response: If the bind option ENCODING was not specified, and instead the value for ENCODING was defaulted from the system default Application Encoding Scheme, verify the Application Encoding Scheme is set to a defined CCSID.

If the value specified for the ENCODING option was ASCII, EBCDIC or UNICODE, verify the corresponding subsystem parameters on the installation panel DSNTIPF has been set to a defined CCSID. Otherwise, inform the user of the correct CCSID.

User response: Verify that the correct encoding scheme was specified in the ENCODING bind option on the *bind-type* subcommand. If correct, see System Programmer Response and contact the System Programmer. If incorrect, specify the correct scheme in the ENCODING option and reissue the *bind-type* subcommand.

Severity: 8 (error)

Related concepts:

Introduction to character conversion (DB2 Internationalization Guide)

Related tasks:

Setting up DB2 to ensure that it interprets characters correctly (DB2 Internationalization Guide)

Related reference:

Application programming defaults panel 1: DSNTIPF (DB2 Installation and Migration)

ENCODING bind option (DB2 Commands)

DSNT225I   *bind-type* **ERROR FOR** *object-type object-name*, *bind-option* **IS NOT SUPPORTED.**

Explanation: The BIND or REBIND command specified invalid options.

bind-type
> They type of bind request: BIND, BIND COPY, or REBIND.

object-type
> The object being bound: PLAN or PACKAGE.

object-name
> The name of the application plan or the package.

bind-option
> The unsupported bind option.

System action: The bind process fails.

User response: Correct the bind option, and rerun the BIND or REBIND command.

Problem determination: One condition under which this message is issued is when DBPROTOCOL(PRIVATE) is either specified or implied, and private protocol is not allowed.

Related reference:

BIND and REBIND options (DB2 Commands)

DSNT226I   *bind-type* **WARNING FOR** *object-type object-name*. *bind-option* **OPTION IS NOT RECOMMENDED WHEN BINDING PLANS OR PACKAGES**

Explanation: The indicated subcommand was issued with a bind option that is no longer recommended for use.

bind-type
>   Type of bind subcommand: BIND COPY | BIND REPLACE | REBIND | BIND PLAN | REBIND PLAN.

object-type
>   PLAN or PACKAGE.

object-name
>   The name of the application plan or package.

bind-option
>   The bind option that is not recommended.

System action: The bind process continues.

User response: If DBPROTOCOL(PRIVATE) was explicitly specified for the bind-option that is not recommended, then DB2 will issue a warning during the bind-type operation on object-type. To avoid this warning, rerun the bind-type operation on object-type specifying DBPROTOCOL(DRDA) or allow the bind-type operation to default.

DSNT227I   *bind-action* **WARNING FOR SQL PROCEDURE PACKAGE:** *location-name*.*collection-id*.*package-id*.**(***version-id***) THE TARGET SERVER WAS NOT ABLE TO RESOLVE STRUCTURE INFORMATION IN THE SOURCE PACKAGE. ADDITIONAL AUTOMATIC REGENERATION PROCESSING WILL OCCUR ON THE TARGET SERVER.**

Explanation: A stored procedure was created in one version of DB2 and bound for use on an earlier version. Some structural information in the new package must be resolved by the target server during the bind process. If the BIND request succeeded, you can use the resulting package on the earlier DB2 version.

bind-action
>   The requested BIND action, such as BIND DEPLOY.

location-name.*collection-id*.*package-id*.**(***version-id***)**
>   Identifies the package that was bound.

System action: Processing continues.

User response: No action is required on the Version 11 subsystem. When you execute the package on the target server, monitor the console output for SQLCODE -904 which indicates that the automatic processing was unsuccessful.

Related tasks:

⇥ Deploying a native SQL procedure to another DB2 for z/OS server (DB2 Application programming and SQL)

Related reference:

⇥ BIND PACKAGE (DSN) (DB2 Commands)

⇥ BIND and REBIND options (DB2 Commands)

DSNT228I   **BIND ERROR, ATTEMPTING TO REPLACE PACKAGE =** *package_name* **WITH version=** *(version2)* **BUT THIS VERSION ALREADY EXISTS**

Explanation: An attempt is made to create a version of a package that already exists. The version specified in the REPLVER keyword is different from the version specified for the precompile. The version specified for the precompile already exists in the catalog. The combination of 'location.collection.package.version' must be unique in the SYSIBM.SYSPACKAGE catalog table. A common mistake is that the user may believe that the version he is creating is the one specified in the REPLVER keyword. This is not the case. The version specified in the REPLVER keyword is the name of the version being replaced. The version that will be created is the version that was given to the program when it was precompiled.

package_name
>   Fully qualified package name

version2
      Version-id of package that is to be created

System action: The bind will fail.

System programmer response: There are two approaches to solve this problem. The first is to re-precompile the program with a new version name and re-issue the original BIND subcommand. The other approach is not to do the precompile but re-issue the BIND subcommand without the REPLVER parameter.

DSNT229I   **BIND ERROR, ATTEMPTING TO REPLACE PACKAGE =** *package-name* **THERE ARE ENABLE OR DISABLE ENTRIES CURRENTLY ASSOCIATED WITH THE PACKAGE**

Explanation: The BIND subcommand was issued to replace a package that has ENABLE or DISABLE entries currently associated with the package. The individual variable fields contain:

package-name
      Name of the package (location.collection.package)

System action: Package not bound.

System programmer response: FREE the package first and then BIND the package.

DSNT230I   **BIND DBRM MEMBER NAME ERROR USING** *auth-id* **AUTHORITY** *object-type = object-name* **MEMBER** *dbrm-name* **NOT FOUND IN PDS SEARCH ORDER**

Explanation: The indicated 'dbrm-name' was not found in the search order in the PDSs specified on the BIND PLAN or BIND PACKAGE subcommand.

auth-id Authorization ID of the plan or package owner. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

object-type
      The type of object being bound: PLAN or PACKAGE

object-name
      If 'object-type' is PLAN, 'object-name' is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

dbrm-name
      Member name of the DBRM that was not found.

System action: The bind is unsuccessful.

System programmer response: Determine if an incorrect 'dbrm-member-name' or PDS was specified. Reinvoke the BIND subcommand specifying the correct names.

DSNT231I   **BIND DBRM ERROR USING** *auth-id* **AUTHORITY** *object-type = object-name* **DBRM** *dbrm-name* **IS IN AN INCONSISTENT STATE**

Explanation: The contents of a DBRM have been modified after the precompilation process, or the DBRM contains incorrect information in the header.

auth-id Authorization ID of the invoker of the BIND subcommand. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

object-type
      The type of object being bound: PLAN or PACKAGE

object-name
      If 'object-type' is PLAN, 'object-name' is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

dbrm-name
> Name of the DBRM that is in error.

System action: The bind is unsuccessful.

System programmer response: Precompile the application program associated with the indicated DBRM. Reinvoke the BIND subcommand.

DSNT232I   SUCCESSFUL *bind-type* **FOR PACKAGE =** *package-name*

Explanation: A BIND, REBIND or FREE PACKAGE subcommand has completed successfully.

bind-type
> Type of bind subcommand: BIND, REBIND, or FREE

package-name
> Name of the package in the following format: 'location.collection.package.(version)'

System action: The package was successfully bound, rebound, or freed.

DSNT233I   UNSUCCESSFUL *bind-type* **FOR PACKAGE =** *package-name*

Explanation: Errors have been detected during the processing of the indicated BIND, REBIND, or FREE PACKAGE subcommand.

bind-type
> Type of bind subcommand: BIND, REBIND, or FREE

package-name
> Name of the package in the following format: 'location.collection.package.(version)'

System action: The indicated package was not bound, rebound, or freed.

System programmer response: See other messages issued to determine the error that caused the subcommand to terminate unsuccessfully.

DSNT234I   MESSAGE LIMIT EXCEEDED

Explanation: A fixed amount of space exists in which to generate bind subcommand messages. This message says that the bind subcommand generated more messages than will fit in that space.

System action: For bind subcommands that affect a single package, the bind subcommand process continues. However, no other messages are generated except for completion message DSNT232I or DSNT233I. For bind subcommands that affect more than one package, the bind subcommand process terminates and no other messages are generated.

System programmer response: The 'FLAG' parameter of the bind subcommand may be used to limit the types of messages that are generated.

DSNT235I   *csect-name bind-type* AUTHORIZATION ERROR USING *auth-id* AUTHORITY PACKAGE = *package-name* **PRIVILEGE =** *privilege*

Explanation: The authorization ID given does not have the *privilege* indicated and cannot invoke the bind subcommand on the named package. If the access control authorization exit is active and the AUTHEXIT_CHECK system parameter is set to DB2, this error might occur if ACEE cannot be created for the indicated authorization ID.

bind-type
> Type of bind subcommand: BIND, REBIND, or FREE

auth-id Authorization ID of the invoker of the BIND subcommand. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

package-name
> Name of the package in the following format: 'location.collection.package.(version)'

privilege
Name of the privilege not held:

BINDADD or BINDADD *
The authority to create a new package using BIND with the ADD option.

BIND The authority to BIND (REPLACE) or REBIND a package.

BIND *
The authority to add a new version of an existing package using BIND with the ADD option.

FREE There is no FREE privilege, but the user needs the proper authority to free the package.

COPY The authority to COPY from the indicated package.

CREATE IN or CREATE IN *
The authority to create a package in the indicated collection.

A blank and asterisk (*) in the privilege values 'BINDADD *', 'BIND *', and 'CREATE IN *' in the message indicates that the BINDNV subsystem parameter has the value BIND. When ' *' does not appear in these privilege values, the BINDNV subsystem parameter has the value BINDADD.

The BINDNV subsystem parameter affects the authorization required to add a new package or a new version of an existing package to a collection.

System action: The indicated package is not bound, rebound, or freed.

System programmer response: You must grant the indicated privilege to the authorization ID of the package owner. If you use the access control authorization exit, ensure that the indicated authorization ID is defined in RACF and granted the indicated privilege in RACF.

Related concepts:

[→] Access control authorization exit routine (Managing Security)

Related tasks:

[→] Binding an application (DB2 Application programming and SQL)

Related reference:

[→] BIND NEW PACKAGE field (BINDNV subsystem parameter) (DB2 Installation and Migration)

[→] Explicit package privileges (Managing Security)

DSNT236I **REBIND FOR PACKAGE** *package-name* **FAILED BECAUSE IBMREQD OF** *ibmreqd* **IS INVALID**

Explanation: The IBMREQD column of the SYSIBM.SYSPACKAGE catalog table for the named package contains an unrecognizable character. This usually means that the package depends on a function of DB2 which is not supported by the currently active release.

package-name
Name of the package in the following format: 'location.collection.package.(version)'

System action: The package was not rebound.

System programmer response: Warn users not to use the package until the DB2 subsystem is remigrated to the newer release.

DSNT237I *csect-name* **BIND** *sub-type* **ERROR USING** *auth-id* **AUTHORITY PACKAGE =** *package-name* **ALREADY EXISTS**

Explanation: An attempt is made to add a package that already exists. The combination of 'location.collection.package.version' must be unique in the SYSIBM.SYSPACKAGE table. In addition, the combination of 'location.collection.package.consistency-token' must be unique.

sub-type
The sub-type can have the values: COPY, ADD or REPLACE.

auth-id Authorization ID of the owner of the package. **Attention:** If you are using a trusted context, the token

auth-id might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: `ROLE:` *role-name.*

package-name
Name of the package in the following format: 'location.collection.package.(version)'

System action: No package is created.

System programmer response: Check the SYSIBM.SYSPACKAGE catalog table for names of existing application packages. Re-invoke the BIND subcommand with a 'location.collection.package.version' that is not in use.

DSNT238I   *bind-type error-type* **USING** *auth-id* **AUTHORITY PACKAGE =** *package-name* **DOES NOT EXIST**

Explanation: The indicated subcommand was issued against a package that does not exist. The individual variable fields contain:

bind-type
Type of bind subcommand (BIND COPY | BIND REPLACE | REBIND | FREE| BIND PLAN | REBIND PLAN | DROP PACKAGE). If BIND or REBIND PLAN is specified, the package-name refers to a package on the PKLIST for the plan.

error-type
Type of error: WARNING or ERROR.

auth-id   Authorization ID of the invoker of the BIND subcommand.

package-name
Name of the package in the following format 'location.collection.package.(version)'.

System action: Package not rebound or freed.

System programmer response: Check the SYSPACKAGE catalog table for the correct 'location.collection.package.version' to use.

DSNT239I   **PLAN** *plan-id* **CONTAINS NO DBRMS AND NO PACKAGES**

Explanation: A plan with no DBRMs was rebound with the NOPKLIST option specified. This creates an empty plan. No SQL can be executed. The individual variable fields contain:

plan-id   Plan name specified on the REBIND command.

System action: This is a warning message. The REBIND process will continue.

System programmer response: If an empty plan is not expected, reissue the REBIND command with the PKLIST keyword specifying a new list.

DSNT240I   *csect-name bind-type* **ERROR, ASTERISK NOT ALLOWED ON REMOTE** *bind-type* **PACKAGE**

Explanation: When remotely rebinding or freeing a package, an asterisk (*) cannot be specified anywhere in the name of a package. In other words, it is not possible to remotely rebind or free multiple packages.

System programmer response: Remove the asterisk and rebind, or free a specific package.

Related tasks:

➡ Binding packages at a remote location (DB2 Application programming and SQL)

Related reference:

➡ REBIND PACKAGE (DSN) (DB2 Commands)

➡ FREE PACKAGE (DSN) (DB2 Commands)

DSNT241I *bind-type* **AUTHORIZATION** *error-type* **ON PLAN** *plan-id* **AUTHORIZATION ID** *auth-id* **NOT AUTHORIZED TO EXECUTE PACKAGE** *package-name*

Explanation: The indicated authorization ID does not have EXECUTE authority on the specified package in the package list. If the access control authorization exit is active and the AUTHEXIT_CHECK system parameter is set to DB2, this error might occur if ACEE cannot be created for the indicated authorization ID.

bind-type
> Type of bind subcommand: BIND or REBIND

error-type
> Type of error: WARNING or ERROR. If VALIDATE(BIND) was specified, this is an ERROR message. Otherwise, it is a WARNING message.

plan-id Name of application plan specified in the subcommand.

auth-id Authorization ID of the plan owner. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

package-name
> Name of the package (collection.package). The location of the package is the local DB2 location name.

System action: If the bind subcommand was invoked with VALIDATE(BIND), the BIND or REBIND operation is not performed. If VALIDATE(RUN) was used then the BIND or REBIND operation will continue.

System programmer response: The indicated plan owner should be granted the EXECUTE privilege on the package. If you use the access control authorization exit, ensure that the indicated authorization ID is defined in RACF and granted the indicated privilege in RACF.

DSNT242I *bind-type* **ERROR USING** *auth-id* **AUTHORITY NO BIND OR REBIND SUBCOMMANDS ARE ALLOWED FOR THIS AUTHID**

Explanation: The authorization ID is not allowed to perform the indicated 'bind-type' against a plan or package because entries in the RLST table prohibit binding and rebinding by this authorization ID, or all authorization IDs.

bind-type
> Type of bind subcommand: BIND or REBIND

auth-id Authorization ID of the plan or package owner. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned, if a role was in effect and the authorization checking is performed against the role rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: ROLE: *role-name*.

System action: The plan or package is not bound or rebound.

System programmer response: If the indicated authorization ID should be allowed to bind, change the entry in the active RLST table.

DSNT243I **BIND ERROR FOR PACKAGE =** *pkg-id* **CONTOKEN =** ′*contoken*′**X IS NOT UNIQUE SO IT CANNOT BE CREATED**

Explanation: An attempt is made to add or replace a package with a consistency token that is not unique for that package. In other words, the combination of 'location.collection.package.consistency-token' already exists.

pkg-id Fully qualified name of the package.

contoken
> Consistency token in hexadecimal.

System action: The BIND will fail.

System programmer response: Check the SYSIBM.SYSPACKAGE catalog table for names of existing application packages with the indicated consistency token. Re-issue the BIND subcommand such that the 'location.collection.package.consistency-token' is unique within the catalog. The following SQL statement can be used to query the catalog,

```
SELECT COLLID,NAME
FROM loc-id.SYSIBM.SYSPACKAGE
WHERE HEX(CONTOKEN) = 'contokn';
```

DSNT244I   *bind-type* **ERROR** *keyword* **KEYWORD IS NOT SUPPORTED FOR REMOTE PROCESSING**

```
CLUSTERED          = clustered
CLUSTERRATIO       = clusterratio
CLUSTERRATIOF      = clusterratiof
FIRSTKEYCARD       = firstkeycard
FIRSTKEYCARDF      = firstkeycardf
NLEAF              = nleaf
NLEVELS            = nlevels
SPACEF             = spacef
DATAREPREATFACTORF = datarepeatfactorf
```

Explanation: The indicated keyword is not supported for the remote bind or rebind subcommand.

bind-type
> Type of bind subcommand: BIND or REBIND

keyword
> Name of the keyword that is not supported.

Programmer response: Correct the syntax and reissue the command.

Related reference:

↪ BIND PACKAGE (DSN) (DB2 Commands)

↪ BIND PLAN (DSN) (DB2 Commands)

↪ REBIND PACKAGE (DSN) (DB2 Commands)

↪ REBIND PLAN (DSN) (DB2 Commands)

DSNT245I   **LOCATION** *location* **CANNOT BE SPECIFIED. THE LOCAL DB2 IS NOT DEFINED**

Explanation: Since the local DB2 location name is undefined, no location name may be given when specifying a package name on a bind subcommand.

location
> Location specified on the bind subcommand.

\

System action: The bind subcommand was not performed.

System programmer response: The location name of the local DB2 must be defined in BSDS if the location is used in the BIND, REBIND, or FREE subcommand.

DSNT246I   *bind-type* **FOR PACKAGE** *package-name* **WITH MARK** *release-dependency-mark* **FAILED BECAUSE PACKAGE DEPENDS ON FUNCTIONS OF THE RELEASE FROM WHICH FALL BACK HAS OCCURRED**

Explanation: The named package depends on a function of DB2 which is not supported by the currently active release.

bind-type
> Type of bind subcommand: REBIND

package-name
> Name of the package in the following format: 'location.collection.package.(version)'

release-dependency-mark
> A one-character mark showing the oldest release of the DB2 which can support this package. The release dependency mark for the package is kept in the IBMREQD column in the DB2 catalog in SYSIBM.SYSPACKAGE.

System action: The 'bind-type' operation for this package is not performed.

System programmer response: Warn users not to use plans containing this package until the DB2 subsystem has been remigrated to the newer release.

User response: The package cannot be bound until the DB2 subsystem is remigrated to the newer release.

DSNT247I   *bind-type* **ERROR** *object-type* **NOT FOUND FOR** *object-type = object-name*

Explanation: You attempted to FREE or REBIND multiple plans or packages using the asterisk (*) but no plans or packages were found.

bind-type
> Type of bind subcommand: REBIND or FREE

object-type
> Type of object to be freed or rebound: PLAN or PACKAGE

object-name
> If object-type is PLAN, object-name is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

System action: The REBIND or FREE operation is not performed.

User response: Check the plan or package name. If it is correct, no plans or packages were found and no operation can be performed; otherwise, correct the plan or package name and reissue the REBIND or FREE subcommand.

DSNT248I   *csect-name bind-type* **ERROR FOR** *object-type object-name* **CCSID =** *ccsid* **IS INVALID FOR ENCODING**
            SCHEME = *scheme*

Explanation: The system does not have a valid CCSID defined for the specified encoding scheme, or the specified CCSID is not defined on the system.

bind-type
> Type of bind subcommand: BIND or REBIND.

object-type
> PLAN or PACKAGE

object-name
> If *object-type* is PLAN, *object-name* is the name of the application. Otherwise, it is the name of the PACKAGE in 'location.collection.package' format.

scheme   This field displays the encoding scheme specified in the ENCODING bind option on the *bind-type* subcommand or defaulted from the system default Application Encoding Scheme. If *scheme* is UNKNOWN, then the CCSID value has not been defined in the ENCODING bind option or by the system default Application Encoding Scheme, and therefore the encoding scheme is not known.

ccsid   This field displays the numeric CCSID specified in the ENCODING bind option or the numeric CCSID corresponding to the scheme specified in the ENCODING bind option. Otherwise, *ccsid* displays the numeric CCSID corresponding to the specified *ccsid-type*.

System action: The *bind-type* operation is not performed

System programmer response: If the bind option ENCODING was not specified, and the value for ENCODING was defaulted from the system default Application Encoding Scheme instead, verify the Application Encoding Scheme is set to a defined CCSID.

If the value specified for the ENCODING option was ASCII, EBCDIC or UNICODE, verify the corresponding subsystem parameters on the installation panel DSNTIPF has been set to a defined CCSID. Otherwise, inform the user of the correct CCSID.

User response: Verify that the correct encoding scheme was specified in the ENCODING bind option on the *bind-type* subcommand. If correct, see System Programmer Response and contact the system programmer. If incorrect, specify the correct scheme in the ENCODING option and reissue the *bind-type* subcommand.

Severity: 8 (error)

Related concepts:

⇨ Introduction to character conversion (DB2 Internationalization Guide)

Related tasks:

⇨ Setting up DB2 to ensure that it interprets characters correctly (DB2 Internationalization Guide)

Related reference:

⇨ Application programming defaults panel 1: DSNTIPF (DB2 Installation and Migration)

⇨ ENCODING bind option (DB2 Commands)

DSNT249I   *csect-name bind-type* **WARNING FOR PLAN** *plan-name keyword* **LOCATION-ID** *location-id* **NOT FOUND IN SYSIBM.LOCATION**

Explanation: A 'location-id' was specified on the PKLIST or CURRENTSERVER keyword on a bind subcommand. That location is not defined in the SYSIBM.LOCATIONS table.

bind-type
> Type of bind subcommand: BIND or REBIND

plan-name
> Name of the plan

keyword
> The bind subcommand containing the nonexistent location: PKLIST or CURRENTSERVER

location-id
> The nonexistent location

System action: The bind process will continue.

System programmer response: Define the appropriate 'location-ids' in the SYSIBM.LOCATION catalog table prior to executing the plan.

DSNT250I   **BIND UNABLE TO OPEN DBRM LIBRARY USING** *authorization-id* **AUTHORITY** *object-type = object-name*

Explanation: The BIND subcommand was unable to open the DBRM partitioned data set(s).

authorization-id
> Authorization ID of the invoker of the BIND subcommand.

object-type
> PLAN | PACKAGE

object-name
> If 'object-type' is PLAN, 'object-name' is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

System programmer response: Determine why the BIND subcommand was unable to open the DBRM library, and correct the problem.

User response: Contact the system programmer, providing all material associated with the error.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

.

DSNT251I   **BIND UNABLE TO LOCATE DBRM LIBRARY MEMBERS USING** *authorization-id* **AUTHORITY**
object-type = object-name

Explanation: The BIND subcommand received an I/O error while attempting to locate the specified DBRM library member(s).

authorization-id
> Authorization ID of the invoker of the BIND subcommand.

object-type
> PLAN | PACKAGE

object-name
> If object-type is PLAN, object-name is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

System programmer response: Correct the reason for the I/O error, and reissue the BIND subcommand.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

.

DSNT252I   *csect-name bind-type* **OPTIONS FOR PLAN** *plan-name*

bind-option   *option-value*
...
bind-option   *option-value*

Explanation: This message shows the BIND or REBIND options that were used for the plan during bind or rebind processing.

csect-name
> The name of the control section that issued the message.

bind-type
> The type of operation: BIND or REBIND.

plan-name
> The plan that was processed.

bind-option
> The name of a BIND PLAN or REBIND PLAN option.

option-value
> The value of a BIND PLAN or REBIND PLAN option.

Severity: 0 (informational)

DSNT253I   *csect-name bind-type* **OPTIONS FOR PLAN** *plan-name*

Explanation: This message shows the BIND or REBIND options that were used for the plan during bind or rebind processing.

csect-name
> The name of the control section that issued the message.

bind-type
> The type of operation: BIND or REBIND.

plan-name
> The plan that was processed.

The message heading text is followed by option names and settings:

```
csect-name bind-type OPTIONS FOR PLAN plan-name
(DEFER | NODEFER)      PREPARE
CACHESIZE              option-1
QUALIFIER              option-2
CURRENTSERVER          option-3
CURRENTDATA            option-4
DEGREE                 option-5
SQLRULES               option-6
DISCONNECT             option-7
REOPT                  option-8
KEEPDYNAMIC            option-9
IMMEDWRITE             option-10
DBPROTOCOL             option-11
OPTHINT                option-12
ENCODING               option-13(option-14)
PATH                   option-15
CONCURRENTACCESSRESOLUTION
                       option-16
```

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNT254I   *csect-name bind-type* **OPTIONS FOR**

PACKAGE	=*package-name*
ACTION	(ADD | REP)
OWNER	*owner*
QUALIFIER	*qualifier*
VALIDATE	(BIND | RUN)
EXPLAIN	(NO | YES)
ISOLATION	(RR | CS | RS | UP | blank)
RELEASE	(COMMIT | DEALLOCATE | blank)
COPY	*package-name*

Explanation: This message shows the BIND or REBIND options that were used for the local package specified by *package-name* during bind or rebind processing.

Severity: 0 (informational)

DSNT255I   *csect-name bind-type* **OPTIONS FOR PACKAGE =** *package-name*

Explanation: This message shows the BIND or REBIND options that were used for the local package during bind or rebind processing.

csect-name
> The name of the control section that issued the message.

bind-type
> The type of operation: BIND or REBIND.

package-name
> The package that was processed.

The message heading text is followed by a list of option names and settings.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

▐➔ BIND PACKAGE (DSN) (DB2 Commands)

▐➔ REBIND PACKAGE (DSN) (DB2 Commands)

DSNT256I   **DBRM MEMBER** *member-name* **IS EMPTY, A PACKAGE CANNOT BE BOUND**

Explanation: The named DBRM member is empty, a package cannot be created.

System action: No package is created.

DSNT257I   **BIND ERROR, PACKAGE** *package-name* **CANNOT BE CREATED BECAUSE IT HAS THE SAME NAME AS THE PACKAGE BEING COPIED**

Explanation: The location and collection-id of the package being created are the same as the local location and the collection-id specified with the COPY keyword. You cannot create a package that is the same name as the package being copied.

package-name
> Name of the package (location.collection.package.(version))

System action: The package is not bound.

System programmer response: Choose a different collection id for the package you are creating and resubmit the subcommand.

DSNT258I   *bind-type* **WARNING FOR PLAN** *plan-name* **THE LOCATION-ID** *location-id* **SPECIFIED IN CURRENTSERVER MAY NEED A PACKAGE LIST**

Explanation: If binding a PLAN with a nonlocal location specified in the CURRENTSERVER, a package list is required using the PKLIST keyword to access any packages at the location specified.

bind-type
> Type of bind subcommand: BIND or REBIND

plan-name
> Name of plan

location-id
> Location of CURRENTSERVER

System action: The bind process will continue.

System programmer response: Add a package list with package(s) that reference the location specified in the CURRENTSERVER or delete the CURRENTSERVER from the BIND or REBIND subcommand.

DSNT259I   *bind-type* **WARNING FOR PLAN** *plan-name* **NO PACKAGES IN THE PACKAGE LIST REFERENCE THE LOCATION-ID** *location-id* **SPECIFIED IN CURRENTSERVER**

Explanation: There are no packages in the package list that reference the location specified in the CURRENTSERVER keyword.

bind-type
> Type of bind subcommand: BIND or REBIND

plan-name
> Name of plan

location-id
> Location of CURRENTSERVER

System action: The bind process will continue.

System programmer response: Change 'location-ids' for the packages in the package list to reference the location in the CURRENTSERVER or change the location specified in CURRENTSERVER.

DSNT260I   *csect-name bind-type* **- THE SUBMITTED PARAMETERS RESULT IN AN INVALID COMBINATION OF** *bind-option1* **AND** *bind-option2* **FOR** *object-type* = *object-name*

Explanation: The BIND or REBIND subcommand has been submitted with an invalid combination of bind option or parameter values.

bind-type
>   Type of bind subcommand: BIND or REBIND.

object-type
>   PLAN or PACKAGE

object_name
>   If *object-type* is PLAN, *object-name* is the name of the application plan. If *object_type* is PACKAGE, *object_name* is the name of the package in the format *location-id.collection-id.package-id*.

bind-option1 **and** *bind-option2*
>   The invalid combination of bind option values, where *bind-option1* and *bind-option2* are each an option name and value.
>
>   If *object-type* is PLAN, then one of the following applies:
>   - *bind-option1* was used on the REBIND subcommand and *bind-option2* exists in the SYSIBM.SYSPLAN catalog table for PLAN *object-name*, or
>   - *bind-option2* was used on the REBIND subcommand and *bind-option1* exists in the SYSIBM.SYSPLAN catalog table for PLAN *object-name*.
>
>   If *object-type* is PACKAGE, then one of the following applies:
>   - *bind-option1* was used on the BIND COPY or REBIND subcommand and *bind-option2* exists in the SYSIBM.SYSPACKAGE catalog table for PACKAGE *object-name*, or
>   - *bind-option2* was used on the BIND COPY or REBIND subcommand and *bind-option1* exists in the SYSIBM.SYSPACKAGE catalog table for PACKAGE *object-name*.

The combination of *bind-option1* and *bind-option2* is not supported by DB2.

System action: The BIND or REBIND process is terminated for the indicated plan or package.

User response: Remove the invalid combination of bind options and reissue the BIND or REBIND command.

Related reference:

�B▶ BIND PACKAGE (DSN) (DB2 Commands)

▶ BIND and REBIND options (DB2 Commands)

▶ BIND PLAN (DSN) (DB2 Commands)

▶ REBIND PACKAGE (DSN) (DB2 Commands)

▶ REBIND PLAN (DSN) (DB2 Commands)

▶ SYSIBM.SYSPACKAGE table (DB2 SQL)

▶ SYSIBM.SYSPLAN table (DB2 SQL)

DSNT261I   *bind-type* **WARNING FOR** *object-type* = *object-name* **ALL CONNECTION TYPES ARE DISABLED**

Explanation: All possible connection types are disabled. The plan or package cannot be executed.

bind-type
>   Type of bind subcommand: BIND or REBIND

object-type
>   PLAN or PACKAGE

object-name
>   If object-type is PLAN, object-name is the name of the application plan. Otherwise, it is the name of the package in 'location.collection.package' format.

System action: The bind process will continue.

System programmer response: Delete one or more connection types from DISABLE clause or add a list of connection names associated with one of the connection types (DLIBATCH, CICS, IMSBMP, IMSMPP, or REMOTE. REMOTE is only valid for BIND/REBIND PACKAGE)

DSNT262I   *csect-name bind-type* **WARNING - ISOLATION** *old-iso* **NOT SUPPORTED. CHANGING TO ISOLATION** *new-iso*

Explanation: DB2 does not support the isolation level specified on the local BIND or REBIND PACKAGE subcommand. The isolation level specified by *new-iso* is used instead.

System action: Processing continues

User response: If the new isolation level is not satisfactory, change the ISOLATION option on the BIND or REBIND PACKAGE subcommand and rebind the package. Use a value that DB2 supports.

Severity: 4

DSNT263I   *csect-name bind-type* **FOR** *object-type* = *object-name* **FAILED BECAUSE THE** *bind-type* **DEPENDS ON FUNCTIONS IN A LATER DB2 RELEASE WITH MARK** *release-dependency-mark* **OR HIGHER**

Explanation: The current DB2 does not support a bind option or feature specified in the indicated bind subcommand. The bind option or feature depends on functions in a later DB2 release.

bind-type
> Type of bind subcommand: BIND or REBIND

object-type
> PLAN or PACKAGE

object-name
> If *object-type* is PLAN, *object-name* is the name of the plan.
>
> If *object-type* is PACKAGE, *object-name* is the name of the package in the format `location-id.collection-id.package-id`.

release-dependency-mark
> A one-character mark indicating the oldest DB2 release that can support the bind subcommand.

This message can be issued in one of the following situations:
- When the DSN command explicitly or implicitly uses the name of a DB2 subsystem that does not support a bind option or feature used in the DSN bind subcommand *bind-type*.
- When the DSN command uses a group attachment name for a DB2 data sharing group and the receiving DB2 data sharing member does not support a bind option or feature used in the DSN bind subcommand *bind-type*.
- When attempting to use new function before new function mode has been enabled.

System action: The *bind-type* operation is not performed.

System programmer response: See the 'User Response' section of this message.

User response: In the DSN command, specify the name of a DB2 subsystem having release dependency mark *release-dependency-mark* or higher that supports all of the bind options or features used in the *bind-type* subcommand.

The IBMREQD columns in the SYSIBM.SYSPLAN or SYSIBM.SYSPACKAGE catalog tables contains the release dependency marks.

Related reference:

➡ DB2 catalog tables (DB2 SQL)

➡ SYSIBM.SYSPLAN table (DB2 SQL)

➡ SYSIBM.SYSPACKAGE table (DB2 SQL)

DSNT264I   *csect-name bind-type bind-subtype* **ERROR USING** *auth-id* **AUTHORITY PACKAGE** *package-name* **IS A**
    package-type **PACKAGE**

Explanation: If this error is issued for TRIGGER package, then:

- To rebind a trigger package, you must use the DSN subcommand REBIND TRIGGER PACKAGE.
- To free a trigger package, you must use the SQL statement DROP TRIGGER.
- You cannot copy a trigger package.

If the error is issued for NATIVE SQL PL PROCEDURE package, then:

- To free a native SQL PL PROCUEDURE package, you must use the SQL statement DROP PROCEDURE.

The message tokens are as follows:

bind-type
        Type of bind subcommand: REBIND, or FREE

bind-subtype
        Subtype of bind subcommand: TRIGGER or DEPLOY

auth-id   Authorization ID attempting the prohibited action

package-name
        Name of the package in the following format:

        `'collection.package'`

package-type
        Type of package: TRIGGER, NATIVE SQL PL PROCEDURE.

System action: The package is not copied, rebound, or freed.

System programmer response: Check the SYSIBM.SYSPACKAGE catalog table for names of existing trigger packages or native SQL PL procedure packages. Call the bind subcommand again, with a 'collection.package' that is not a trigger or native SQL PL procedure package.

DSNT265I   *csect-name bind-type bind-subtype* **ERROR USING** *auth-id* **AUTHORITY PACKAGE** *package-name* **IS**
    NOT A *package-type* **PACKAGE**

Explanation: The subcommand REBIND TRIGGER PACKAGE can be used only for trigger packages. The subcommand BIND PACKAGE DEPLOY can be used only for native SQL PL procedure packages. The subcommand FREE PACKAGE can be used only for non-trigger and non-native SQL PL procedure packages.

The message tokens are as follows:

bind-type
        Type of bind subcommand: REBIND, FREE

bind-subtype
        Subtype of bind subcommand: TRIGGER, DEPLOY

auth-id   Authorization ID attempting the prohibited action

package-name
        Name of the package in the following format:

        `'collection.package'`

package-type
        Type of package: TRIGGER, NATIVE SQL PL PROCEDURE

System action: The package specified is not rebound, deployed, or freed.

System programmer response: For a package that is not a trigger package, rebind using the subcommand REBIND PACKAGE. Re-invoke the bind subcommand with a 'collection.package' that is an accepted package.

DSNT266I   *csect-name bind-type* **FOR** *object-type* = *object-name* **FAILED. THE TOTAL LENGTH OF THE PATH SCHEMA NAME LIST EXCEEDS 254 CHARACTERS WITH SCHEMA NAME** *schema-name*

Explanation: DB2 uses the schema name list you specified in the bind option PATH to set the DB2 SQL Path that will be used for the bind process. The SQL Path is a VARCHAR(254) string containing the list of schema names you specified, where each schema name is delimited with double quotes and separated from the next schema name by a comma. The total length of this SQL Path string, including the delimiters and commas, cannot exceed 254 characters. The indicated schema name *schema-name* in the bind option PATH causes the SQL Path string to exceed 254 characters.

bind-type
>   Type of bind subcommand: BIND or REBIND

object-type
>   PLAN or PACKAGE

object-name
>   If *object-type* is PLAN, *object-name* is the name of the plan.
>
>   If *object-type* is PACKAGE, *object-name* is the name of the package in the format 'location-id.collection-id.package-id.(version-id)'.

schema-name
>   The schema name specified in the bind option PATH causing the SQL Path to exceed 254 characters.

System action: The *bind-type* operation is not performed.

System programmer response: See the 'User Response' section of this message.

User response: In the bind option PATH, specify a schema name list that does not cause the SQL Path to exceed 254 characters.

For more information about the bind option PATH and the schema name list, please see: Command Reference.

DSNT267I   *csect-name* **A TRIGGER PACKAGE CANNOT BE FREED USING THE FREE PACKAGE COMMAND**

Explanation: A FREE PACKAGE command was received that identified the trigger package to be freed. Trigger packages can only be deleted using the DROP TRIGGER SQL statement.

System action: The TSO Attachment Facility is ready for more input.

User response: Use the SQL statement DROP TRIGGER to delete the trigger package.

DSNT268I   *csect-name* **A BIND COMMAND USING THE COPY OPTION NAMED A TRIGGER PACKAGE AS THE SOURCE FOR THE BIND**

Explanation: A trigger package cannot be used as the source for a BIND COPY command.

System action: The TSO Attachment Facility is ready for more input.

User response: Correct the BIND command and resubmit.

DSNT269I   *csect-name* **DB2 SWITCHED TO THE** *copy-indicator* **COPY OF PACKAGE** *package-name*. **THIS COPY IS NOT EXECUTABLE WITHOUT AN EXPLICIT REBIND OR AUTOBIND (REASON=***reason-code***).**

Explanation: The command was successfully processed, but the invalid copy has replaced the current package copy. This copy cannot be used.

csect-name
>   The name of the control section that issued the message.

copy-indicator
>   Indicates whether the invalid copy that was used was the previous or original package copy. Expected values are PREVIOUS or ORIGINAL.

package-name
>   The invalid package, in the format *location-id.collection-id.package-id.(version-id)*.

| *reason-code*
| A numeric value that indicates why the copy is not executable.

| **01** The copy is invalid.

| **02** The copy is from a prior release that is no longer executable without an explicit REBIND or
| autobind.

| **System action:** The command was successfully processed, but the invalid copy has replaced the current package
| copy, which cannot be used.

| **User response:** Determine why the package is invalid and resolve that condition before invoking the package.
| Otherwise, the package will be automatically rebound at the first invocation.

| **Severity:** 4 (warning)

| **Related concepts**:

| ⬆ Automatic rebinding (DB2 Application programming and SQL)

| ⬆ Changes that invalidate packages (DB2 Application programming and SQL)

| **Related reference**:

| ⬆ SYSIBM.SYSPACKAGE table (DB2 SQL)

| ⬆ REBIND PACKAGE (DSN) (DB2 Commands)

DSNT270I   THE FOLLOWING SQLCA INFORMATION WAS RETURNED FROM *subsystem-id*
- SQLCODE = *sql-code*
- SQLSTATE = *sql-state*
- SQLERRMT = *sqlca-tokens*
- SQLWARN *warn_fields*

Explanation: A warning or error has been received from a database server using Distributed Relational Database
Architecture (DRDA) during remote program BIND processing.

subsystem-id
The database server.

sql-code A numeric value that indicates why the SQL statement execution was unsuccessful.

sql-state The corresponding SQLSTATE for the SQLCODE.

sqlca-tokens
Tokens that are passed in the message portion of the SQLCA for problem determination.

warn_fields
The warning information indicators.

System action: See the System Action associated with the indicated SQLCODE.

System programmer response: See the Programmer Response associated with the indicated SQLCODE and/or
SQLSTATE. Take note of the warning or error (as indicated in the SQLCODE, SQLSTATE, or one or more of the
SQLWARN fields), and determine if a program change is necessary. A change to the application might be necessary
or the system programmer might have to be consulted depending on the situation.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and
reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
  time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Related concepts:

▶ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

▶ Description of SQLCA fields (DB2 SQL)

DSNT271I   THE FOLLOWING SQLCA INFORMATION WAS RETURNED FROM *subsystem-id*

- SQLCODE = *sql-code*
- SQLSTATE = *sql-state*
- SQLERRMT = *sqlca-tokens*
- SQLWARN *warn_fields*
- STATEMENT = *statement_number*

Explanation: A warning or error has been received from a database server using Distributed Relational Database Architecture (DRDA) during remote program BIND processing.

subsystem-id
> The database server.

sql-code   A numeric value that indicates why the SQL statement execution was unsuccessful.

sql-state   The corresponding SQLSTATE for the SQLCODE.

sqlca-tokens
> Tokens that are passed in the message portion of the SQLCA for problem determination.

warn_fields
> The warning information indicators.

statement_number
> The statement number of the failing statement.

System action: See the System Action associated with the indicated SQLCODE.

System programmer response: Take note of the warning or error (as indicated in the SQLCODE, SQLSTATE, or one or more of the SQLWARN fields), and determine if a program change is necessary. Depending on the situation, a change to the application might be necessary or the system programmer might have to be consulted.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Related concepts:

▶ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

▶ Description of SQLCA fields (DB2 SQL)

DSNT280I   *bind-command* QUERY FOR *query-type* **=** *query-identifier* **SUCCESSFUL**

Explanation: A BIND operation has completed successfully.

bind-command
> The type of command that resulted in the message:

> **BIND**   For BIND QUERY commands.

> **FREE**   For FREE QUERY commands.

> **LOOKUP**
>> For BIND QUERY commands when the LOOKUP(YES) option is in effect.

query-type
> Information that identifies the query:

>> **QUERYNO**
>>> The query is identified by the query number (QUERYNO) in the table DSN_USERQUERY_TABLE.

>> **QUERYID**
>>> The query is identified by the query ID (QUERYID) in the table SYSIBM.SYSQUERY.

query-identifier
> The query number or query ID, depending on the *query-type* value.

When *bind-command* is LOOKUP, an instance of the DSNT280I message is issued for each row in the DSN_USERQUERY_TABLE that does not have matching rows in both the SYSIBM.SYSQUERY and SYSIBM.SYSQUERYPLAN catalog tables.

System action: Processing continues.

User response: No action is required.

Related tasks:

▶ Specifying optimization parameters at the statement level (DB2 Performance)

Related reference:

▶ BIND QUERY (DSN) (DB2 Commands)

▶ DSN_USERQUERY_TABLE (DB2 Performance)

▶ SYSIBM.SYSQUERY table (DB2 SQL)

Related information:
"DSNT281I"
"DSNT290I" on page 496
"DSNT291I" on page 497

DSNT281I   *bind-command* **QUERY FOR** *query-type* **=** *query-identifier* **NOT SUCCESSFUL. REASON:** *reason-text*

Explanation: Errors were detected during processing of a BIND QUERY or related operation on a query.

bind-command
> The type of command that resulted in the message:

> **BIND**   For BIND QUERY commands.

> **FREE**   For FREE QUERY commands.

> **LOOKUP**
>> For BIND QUERY commands when the LOOKUP(YES) option is in effect.

query-type
> Information that identifies the query:

>> **QUERYNO**
>>> The query is identified by the query number (QUERYNO) in the table DSN_USERQUERY_TABLE.

>> **QUERYID**
>>> The query is identified by the query ID (QUERYID) in the table SYSIBM.SYSQUERY.

query-identifier
> The query number or query ID, depending on the *query-type* value.

reason-text
> *reason-text* is one of the following values:

>> **1. PACKAGE INFORMATION IS INVALID**
>>> The specified package information is not valid

2. THE HINT, OPTION, OR OVERRIDE WAS NOT FOUND

The LOOKUP option was specified and the specified hint, option, or override was not found in the catalog tables.

3. THE STATEMENT TEXT CONTAINS AN ERROR

The statement text contains an error.

4. THE STATEMENT TYPE IS NOT SUPPORTED

The statement is of a type that is not supported by the BIND QUERY command.

5. BIND QUERY DOES NOT SUPPORT *package-type* **PACKAGES**

The query is from a package of a type that is not supported by the BIND QUERY command. Example values for *package-type* might include:

- 'TRIGGER'
- 'PROCEDURE'
- 'FUNCTION'

6. THE *qualified-table-column* **COLUMN CONTAINS AN INVALID VALUE =** *invalid-value*

The input table column that is identified by *qualified-table-column* contains an invalid value, which is provided as *invalid-value*.

7. A REQUIRED *object-type qualifier*.*object-name* **DOES NOT EXIST**

The specified object does not exist.

8. THE *qualified-table-column* **COLUMN FOR PREDNO** *predno-value* **CONTAINS AN INVALID VALUE =** *invalid-value*

The input table column that is identified by *qualified-table-column* contains an invalid value, which is provided as *invalid-value*. *predno-value* identifies the PREDNO value of the row that contains the invalid value.

9. THE *qualified-table-column* **COLUMN CONTAINS AN INVALID COMBINATION OF VALUES:** *invalid-combination*

qualified-table-column identifies a column that contains an invalid combination of values in several rows. Example values for *invalid-combination* include:

NON-MATCHING WEIGHT VALUES FOR ROWS WITH THE SAME INSTANCE VALUE: *instance-value*

For each query, every row with the same instance value must of have the same weight value.

SUM OF INSTANCE WEIGHTS > 1.0

For each query, the sum of weights for all instance values must be less-than or equal to 1.0.

NUMBER OF DISTINCT INSTANCE VALUES *instance-count* **IS GREATER THAN** *max-instances***.**

Too many instances where specified. DB2 supports only as many instances as the value specified by *max-instances*

10. THE *qualified-table-column* **COLUMN CONTAINS AN INVALID COMBINATION OF VALUES FOR PREDNO =** *predno-value***:** *invalid-combination*

The input table column that is identified the *qualified-table-column* value contains an invalid combination of values in several rows. Example *invalid-combination* values might include:

- MULTIPLE ROWS CONTAIN THE SAME INSTANCE VALUE *instance-value*
- DUPLICATE ROWS EXIST FOR THE SAME COMPOUND PREDICATE

The *predno-value* identifies the PREDNO value of the rows that contain the invalid combination of values.

11. INPUT TABLE COLUMNS CONTAIN INCOMPATIBLE VALUES *additional-information*

The value of *additional-information* is a string that identifies the reason that the command failed. Examples might include:

- THE VALUE OF THE STARJOIN COLUMN CANNOT BE 'Y' WHEN MORE THAN ONE SELECTIVITY INSTANCE EXISTS
- A VALUE IN THE PREDNO COLUMN OF DSN_SELECTIVITY TABLE DOES NOT EXIST IN DSN_PREDICATE_TABLE

| • NONE OF THE FOLLOWING VALUES CONTAINS A 'Y' VALUE: ACCESSPATH_HINT,
| SELECTIVITY_OVERRIDE, AND OPTION_OVERRIDE
| • ONE OF THE FOLLOWING COLUMNS CONTAINS A BLANK VALUE: ACCESSPATH_HINT,
| SELECTIVITY_OVERRIDE, AND OPTION_OVERRIDE. ALL MUST CONTAIN VALUES WHEN
| ANY IS SPECIFIED

| When *bind-command* is LOOKUP, an instance of the DSNT281I message is issued for each row in the
| DSN_USERQUERY_TABLE that has matching rows in both the SYSIBM.SYSQUERY and SYSIBM.SYSQUERYPLAN
| catalog tables.

| **System action:** The query is not processed.

| **System programmer response:** Use the information provided in this code and other messages issued in conjunction
| with this one to determine the reason why the query could not be processed.

| **Related tasks**:

| ➡ Specifying optimization parameters at the statement level (DB2 Performance)

| ➡ Overriding predicate selectivities at the statement level (DB2 Performance)

| **Related reference**:

| ➡ BIND QUERY (DSN) (DB2 Commands)

| ➡ DSN_USERQUERY_TABLE (DB2 Performance)

| ➡ SYSIBM.SYSQUERY table (DB2 SQL)

| ➡ SYSIBM.SYSQUERYPLAN table (DB2 SQL)

| ➡ SYSIBM.SYSQUERYOPTS table (DB2 SQL)

| ➡ SYSIBM.SYSQUERYSEL table (DB2 SQL)

| ➡ SYSIBM.SYSQUERYPREDICATE table (DB2 SQL)

| **Related information**:
| "DSNT280I" on page 490
| "DSNT290I" on page 496
| "DSNT291I" on page 497

DSNT282I   *bind-command* **QUERY AUTHORIZATION ERROR USING** *authorization-id* **AUTHORITY**

Explanation: A BIND operation was invoked by an authorization ID that does not have the required authorization.

bind-command
        The BIND operation: BIND, REBIND, or FREE.

authorization-id
        The authorization ID of the invoker of the command.

System action: The operation is not performed.

User response: Resubmit the command from an authorization ID that has the required authorization.

DSNT284I   **THE** *command-name* **COMMAND ONLY PROCESSES DYNAMIC SQL QUERIES. ANY HINTS
                (SOURCE = 0) WERE NOT PROCESSED.**

Explanation: An attempt was made to rebind or free a static SQL statement or a hint.

command-name
        The BIND command that was issued: REBIND or FREE.

System action: The requested operation is not performed.

User response: No action is required.

DSNT285I *csect-name bind-type* **FOR PACKAGE =** *package-name* **USE OF APCOMPARE RESULTS IN** *count-1*
STATEMENTS WHERE COMPARISON IS SUCCESSFUL *count-2* **STATEMENTS WHERE**
COMPARISON IS NOT SUCCESSFUL *count-3* **STATEMENTS WHERE COMPARISON COULD**
NOT BE PERFORMED.

Explanation:

csect-name
> The name of the control section that issued the message.

bind-type
> The type of bind subcommand: BIND or REBIND

package-name
> The name of the package in the format: *location-id.collection-id.package-id*

count-1 The number of statements where previous access path was identical to the incoming access path.

count-2 The number of statements where previous access path was not identical to the incoming access path.

count-3
> The number of statements where the comparison could not be performed. This could happen either because
> the previous access path was not found, or because the no new access path was generated.

The sum of the total counters in message DSNT285I might be larger than the number of static statements in the
package. The sum of counters is equal to number of the statements in the SYSIBM.SYSPACKSTMT table. If the sum is
larger than the number of static statements in the package, one or more statements might have been bound twice,
which occurs in the following situations:

• SYSTIMESENSITIVE is set to YES, and the statement references a system-period temporal table or bitemporal table.

• The ARCHIVESENSITIVE bind option is set to YES, and the statement references an archive table.

For more information about why these statements are bound twice, see the APCOMPARE bind option information.

System action: If APCOMPARE(WARN) was specified, the command proceeds normally. If APCOMPARE(ERROR)
was specified, the command aborts. If APREUSE(ERROR) was specified, and *count-2* is a non-zero value, the
command aborts.

User response: Determine whether the APCOMPARE usage reported in the message matches the expected result. If
EXPLAIN(YES) or EXPLAIN(ONLY) was used, refer to the PLAN_TABLE.REMARKS column for more information
about individual statements that failed the access path comparison.

Related tasks:

⮕ Reusing and comparing access paths at bind and rebind (DB2 Performance)

⮕ Analyzing access path changes at bind or rebind (DB2 Performance)

Related reference:

⮕ APCOMPARE bind option (DB2 Commands)

⮕ ARCHIVESENSITIVE bind option (DB2 Commands)

⮕ SYSTIMESENSITIVE bind option (DB2 Commands)

⮕ PLAN_TABLE (DB2 Performance)

⮕ SYSIBM.SYSPACKSTMT table (DB2 SQL)

DSNT286I *csect-name bind-type* **FOR PACKAGE =** *package-name***, USE OF APREUSE RESULTS IN** *count-1*
STATEMENTS WHERE APREUSE IS SUCCESSFUL *count-2* **STATEMENTS WHERE APREUSE IS**
EITHER NOT SUCCESSFUL OR PARTIALLY SUCCESSFUL *count-3* **STATEMENTS WHERE**
APREUSE COULD NOT BE PERFORMED *count-4* **STATEMENTS WHERE APREUSE WAS**
SUPPRESSED BY OTHER HINTS.

Explanation: This message reports information about the reuse of access paths.

csect-name
> The name of the control section that issued the message.

bind-type
> The type of bind subcommand: BIND or REBIND

package-name
> The name of the package in the format *location-id.collection-id.package-id*

count-1 The number of statements where the previous the access path was successfully applied.

count-2 The number of statements where the previous access path could not be applied. Reuse fails for the package when this value is not zero.

count-3 The number of statements where the previous access path could not be reused. This situation happens either because the previous access path was not found or because a new access path was not generated. Some common reasons are that the query used REOPT(ALWAYS) or VALIDATE(RUN) bind behavior, or the package was bound before Version 9.

count-4 The number of statements where the previous access path was not used because of the use of other types of hints such as PLAN_TABLE or SYSIBM.SYSQUERYPLAN hints.

The sum of the total counters in message DSNT286I might be larger than the number of static statements in the package. The sum of counters is equal to number of the statements in the SYSIBM.SYSPACKSTMT table. If the sum is larger than the number of static statements in the package, one or more statements might have been bound twice, which occurs in the following situations:

- SYSTIMESENSITIVE is set to YES, and the statement references a system-period temporal table or bitemporal table.
- The ARCHIVESENSITIVE bind option is set to YES, and the statement references an archive table.

For more information about why these statements are bound twice, see the APREUSE bind option information.

System action: The command terminates and no changes are applied.

User response: Determine whether the APREUSE usage reported in the message matches the expected result. If EXPLAIN(YES) or EXPLAIN(ONLY) was used, refer to the PLAN_TABLE.HINT_USED column for more information about individual statements.

Related tasks:

Reusing and comparing access paths at bind and rebind (DB2 Performance)

Analyzing access path changes at bind or rebind (DB2 Performance)

Related reference:

APREUSE bind option (DB2 Commands)

ARCHIVESENSITIVE bind option (DB2 Commands)

SYSTIMESENSITIVE bind option (DB2 Commands)

PLAN_TABLE (DB2 Performance)

SYSIBM.SYSPACKSTMT table (DB2 SQL)

DSNT287I   *module-name command-name* **FOR QUERYID =** *query-id* **FAILED BECAUSE THE CURRENT COPY DOES NOT EXIST**

Explanation: An attempt was made to rebind or free a static SQL statement or a hint.

module-name
> The name of the DB2 module that issued this message.

command-name
> The BIND command that was issued.

query-id The query ID, as specified in the SYSIBM.SYSQUERY table.

System action: Processing continues.

User response: No action is required.

DSNT288I REBIND PACKAGE DID NOT PRESERVE THE PREVIOUS AND/OR ORIGINAL ACCESS PATHS FOR STATIC SQL STATEMENTS IN PACKAGE = *package-name* BECAUSE THEY WERE IDENTICAL TO THE NEW ACCESS PATHS

Explanation: The new access path is identical to the previous access path, the original access path, or both. The older access paths were not saved.

package-name
    The name of the package.

System action: Processing continues.

User response: No action is required.

DSNT289I *bind-command* QUERY COMMAND INVOKED BY THE DSNE PROCESSOR UNDER THIS ENVIRONMENT. THE COMMAND CAN PROCESS ONLY THE FIRST APPLICABLE QUERY. ALL OTHER QUERIES ARE NOT PROCESSED.

Explanation: Multiple queries were issued, but the command is able to process only the first query.

bind-command
    The BIND operation: BIND, REBIND, or FREE.

System action: The first query is processed. Remaining queries are not processed.

User response: No action is required.

DSNT290I *bind-command* QUERY COMMAND COMPLETED

Explanation: A BIND QUERY or related command has completed.

bind-command
    The type of command that resulted in the message:

    BIND For BIND QUERY commands.

    FREE For FREE QUERY commands.

    LOOKUP
        For BIND QUERY commands when the LOOKUP(YES) option is in effect.

When *bind-command* is LOOKUP, a single instance of the DSNT290I message is issued if the SYSIBM.SYSQUERY and SYSIBM.SYSQUERYPLAN catalog tables contain some rows that match the rows in DSN_USERQUERY_TABLE.

System action: Processing continues.

System programmer response: Look for other messages that might indicate that individual queries were not processed successfully.

Related tasks:

Specifying optimization parameters at the statement level (DB2 Performance)

Related reference:

BIND QUERY (DSN) (DB2 Commands)

DSN_USERQUERY_TABLE (DB2 Performance)

SYSIBM.SYSQUERY table (DB2 SQL)

Related information:
"DSNT280I" on page 490
"DSNT281I" on page 491
"DSNT291I" on page 497

| **DSNT291I** *bind-command* **QUERY COMMAND NOT SUCCESSFUL. REASON:** *reason-text*

| **Explanation:** The command could not be initiated because of one or more errors.

| *bind-command*
| The type of command that resulted in the message:

| **BIND** For BIND QUERY commands.

| **FREE** For FREE QUERY commands.

| **LOOKUP**
| For BIND QUERY commands when the LOOKUP(YES) option is in effect.

| *reason-text*
| Additional information that explains why the operation was unsuccessful. Example reason text values
| include:

| • 1. A REQUIRED *object-type qualifier.object-name* DOES NOT EXIST

| • 2. *qualifier.table-name* IS EMPTY

| • ZPARM OPTHINTS IS SET TO NO

| When *bind-command* is LOOKUP, a single instance of the DSNT291I message is issued if the SYSIBM.SYSQUERY and
| SYSIBM.SYSQUERYPLAN catalog tables contain no rows that match the rows in DSN_USERQUERY_TABLE.

| **System action:** No queries are processed.

| **System programmer response:** Use the information provided in the *reason-text* and other messages issued in
| conjunction with this one to determine the reason why the command terminated.

| **Related tasks**:

| ➡ Specifying optimization parameters at the statement level (DB2 Performance)

| **Related reference**:

| ➡ BIND QUERY (DSN) (DB2 Commands)

| ➡ DSN_USERQUERY_TABLE (DB2 Performance)

| ➡ SYSIBM.SYSQUERY table (DB2 SQL)

| **Related information**:
| "DSNT280I" on page 490
| "DSNT281I" on page 491
| "DSNT290I" on page 496

DSNT292I *csect-name bind-type* **PACKAGE =** *package-name*, **DB2 WAS UNABLE TO FIND A PRE-EXISTING
PACKAGE TO OBTAIN PRIOR ACCESS PATHS. APCOMPARE AND/OR APREUSE OPTIONS
WERE IGNORED.**

Explanation: The package referred to in the message was created on a release prior to Version 9. For APCOMPARE
and APREUSE to work, the package must have been bound on Version 9 or later.

csect-name
That name of the control section that issued the message.

bind-type
The type of bind subcommand: BIND or REBIND

package-name
The name of the package in the following format: '*location-id.collection-id.package-id*

System action: Processing continues.

User response: No action is required.

Related tasks:

➤ Reusing and comparing access paths at bind and rebind (DB2 Performance)

➤ Analyzing access path changes at bind or rebind (DB2 Performance)

DSNT294I   *csect-name bind-type* **PACKAGE =** *package-name*, **TO PROCESS APCOMPARE AND/OR APREUSE, PRIOR ACCESS PATHS FROM VERSION** *prior-version* **WERE USED.**

Explanation:

csect-name
> That name of the control section that issued the message.

bind-type
> The type of bind subcommand: BIND or REBIND

package-name
> The name of the package in the following format: '*location-id.collection-id.package-id*

prior-version
> The version of the package that was used for the prior access paths.

System action: Processing continues.

User response: No action is required.

Related tasks:

➤ Reusing and comparing access paths at bind and rebind (DB2 Performance)

➤ Analyzing access path changes at bind or rebind (DB2 Performance)

DSNT295I   *bind-type* **WARNING FOR QUERY =** *query-identifier* **REASON:** *reason-text*

Explanation: A warning condition was encountered during BIND QUERY processing.

bind-type
> Whether the LOOKUP option was specified in the BIND subcommand that resulted in the message.

> **BIND**   The LOOKUP option was not specified.

> **LOOKUP**
> > The LOOKUP option was specified.

query-identifier
> The unique identifier for the query that corresponds to the values of the QUERYID columns in the and *user-ID*.DSN_USERQUERY_TABLE SYSIBM.SYSQUERY tables.

reason-text
> A string value that described the reason for the message. Example *reason-text* values include:

> **1. SYMBOL & IS ENCOUNTERED**
> > The statement text for a statement-level optimization hint contains the ampersand symbol (&). The presence of the ampersand symbol indicates that the hint applies to a cached dynamic SQL statement and that the CONCENTRATE STATEMENTS WITH LITERAL VALUES option is expected when the statement is prepared.

> **2.** *qualifier*.**DSN_USERQUERY_TABLE.ACCESSPATH_HINT AND** *qualifier*.**DSN_USERQUERY_TABLE.OPTION_OVERRIDE ARE BOTH SET TO BLANK. THE** *qualifier*.**DSN_USERQUERY_TABLE OPTIONS ARE IGNORED**
> > Values were found in the some of the DSN_USERQUERYR_TABLE columns that specify option values, but the value of the OPTION_OVERRIDE column does not indicate that option overrides are specified. The option values are ignored.

System action: The hints is created successfully, or the lookup is successful.

User response: Verify that the specified statement is eligible for literal replacement, and the CONCENTRATE STATEMENTS WITH LITERAL VALUES is specified when the statement is prepared.

| Related concepts:

| ➡ Reoptimization for statements with replaced literal values (DB2 Performance)

| ➡ Conditions for statement sharing (DB2 Application programming and SQL)

| Related reference:

| ➡ PREPARE (DB2 SQL)

| ➡ BIND QUERY (DSN) (DB2 Commands)

| **DSNT298I** *csect-name* **ATTEMPT TO USE COMMAND OR OPTION** *command-option* **WHEN THE**
| **APPLICATION COMPATIBILITY IS SET FOR A PREVIOUS LEVEL**

| **Explanation:** A BIND or REBIND command was issued which requires a DB2 feature which is not supported by the
| current application compatibility setting.

| *csect-name*
| The name of the control section that issued the message.

| *command-option*
| The command name or command option which generated the error.

| **System action:** The command cannot be processed.

| **System programmer response:** Correct the command or command option to remove the dependency on the DB2
| feature which is not supported by the current application compatibility setting. The application compatibility setting
| is set by one of the following methods:

| 1. Issuing a BIND or REBIND of the package with the APPLCOMPAT option

| 2. Issuing a BIND or REBIND of the package without the APPLCOMPAT option which defaults to the
| APPLCOMPAT value from SYSIBM.SYSPACKAGE

| 3. Issuing a BIND or REBIND of the package which defaults to the APPLCOMPAT system parameter value

| Related concepts:

| ➡ Application compatibility of packages (DB2 Application programming and SQL)

| Related reference:

| ➡ BIND and REBIND options (DB2 Commands)

| ➡ SYSIBM.SYSPACKAGE table (DB2 SQL)

DSNT300I **AUTH-ID** *authorization-id* **NOT AUTHORIZED TO PERFORM** *function* **ON DATABASE** *dbname*.
REQUEST REJECTED

Explanation: The invoker's 'authorization-id' as designated by AUTHORIZATION ID is not authorized to perform
the specified database control command function on this database. The 'function' is either -START, -STOP, or
-DISPLAY. The name of the database is 'dbname'.

If the operation involves an implicit database or an object in an implicit database, then the applicable privilege is not
held on the implicit database, on database DSNDB04. In this case, the owner of the object is not involved in
performing the operation.

System action: The request is rejected.

System programmer response: The privilege to issue the command must be granted to the indicated
'authorization-id'.

User response: Notify the system programmer.

If the table space is in an implicit database, then obtain the appropriate authorization privileges on the implicit
database or DSNDB04.

DSNT301I   **INVALID DATABASE** *dbname*

Explanation: The databases named in the -START, -STOP, or -DISPLAY DATABASE command do not exist. The name of the database is 'dbname'.

System action: The request is rejected. This message may be repeated.

System programmer response: Reissue the command with a valid name.

DSNT302I   **INVALID NAME** *spname*

Explanation: In the message, 'spname' is the name of the table space or index space. The 'spname' specified in the SPACENAM keyword on the -START, -STOP, or -DISPLAY DATABASE command does not exist.

System action: The request is rejected. This message may be repeated.

System programmer response: Reissue the command with a valid table space name or index name.

DSNT304I   *csect-name* **INSTALLATION SYSOPR IS NOT AUTHORIZED TO CHANGE OBJECT STATUS ON DATABASE** *dbname*. **REQUEST IS REJECTED.**

Explanation: INSTALLATION SYSOPR is not authorized to perform the START DATABASE command function on this database because the command can change an objects status in this database. INSTALL SYSOPR can use the START DATABASE command only to recover LPL pages or GRECP objects. The name of the database is *dbname*.

System action: The request is rejected.

Operator response: Reissue the -START DATABASE command using the correct access mode or contact your system programmer.

DSNT305I   **SUBSYSTEM FAILURE DURING** *command-name* **DATABASE COMMAND PROCESSING. RC =** *return-code*, **REASON =** *reason-code*

Explanation: An error occurred while processing a DB2 command.

command-name
> The name of the command that was being processed when the failure occurred. *command-name* can have one of the following values:

> **DISPLAY**
> > DISPLAY DATABASE command

> **START**   START DATABASE command

> **STOP**   STOP DATABASE command

return-code **and** *reason-code*
> The reason for the error.

> *return-code* is a 4-byte return code.

> *reason-code* is a 4-byte reason code.

> The values of *return-code* and *reason-code* depend on which control section (CSECT) issued the message, as follows:

> **CSECT DSNTDCST:**
> > • For relational data system (RDS) subcomponent errors:
> > > – *return-code* is 8.
> > > – *reason-code* is abend code 00E30085.
> > • For data manager (DM) subcomponent errors:
> > > – *return-code* consists of a 2-byte return code and a 2-byte reason code that is returned by the data manager function that is invoked by DSNTDCST.
> > > – *reason-code* is abend code 00E30086.

A record is written to SYS1.LOGREC, and an SVC dump is requested before the execution unit is abended.

CSECT DSNTDDIS:

- *return-code* is 12.
- *reason-code* is abend code 00E30089.

The request is abended.

CSECTs DSNTDMUL or DSNTDMSG:

- *return-code* is a return code that is returned by the message generator.
- *reason-code* is abend code 00E30084.

The request is abended.

CSECT DSNTDSTP:

- *return-code* is 00000074.
- *reason-code* is internal reason code 00E30064.

The STOP DATABASE command has failed.

This internal reason code indicates that an unexpected error occurred during DB2 catalog lookup for the STOP DATABASE command. Review the current messages in the DB2 MSTR address space and retry the command after any related error conditions have been corrected.

System action: The actions are described in the explanation for each CSECT.

System programmer response: Refer to the information for the particular abend reason code that was issued.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Source listing of the failing application program.

Related reference:

-DISPLAY DATABASE (DB2) (DB2 Commands)

-START DATABASE (DB2) (DB2 Commands)

-STOP DATABASE (DB2) (DB2 Commands)

Related information:

00E30084 (DB2 Codes)

00E30085 (DB2 Codes)

00E30086 (DB2 Codes)

00E30089 (DB2 Codes)

DSNT306I   RESPONSE MESSAGE LIMIT HAS BEEN REACHED

Explanation: A fixed amount of space exists in which to generate DISPLAY, START, or STOP database command messages. This message indicates that the DISPLAY, START, or STOP database command that was specified generated more messages than will fit in that space.

User response: Specify individual databases and/or table spaces rather than specifying '*' (that is, all names).

DSNT307I   INVALID RANGE SPECIFICATION *string1* **IS GREATER THAN** *string2*

Explanation: A range was specified on a DISPLAY DB(string1:string2) or DISPLAY DB(..)
SPACENAM(string1:string2) command or on a database command (START DB, STOP DB, or DISPLAY DB) on
PART(string1:string2) keywordwhere the first string, specified alphanumerically, collates after the second string.

User response: Specify the command again. Make sure the first string does not alphanumerically collate after the
second string specified.

DSNT308I   AFTER KEYWORD ONLY VALID FOR A SINGLE DATABASE OR A SINGLE TABLESPACE/
INDEXSPACE WITHIN A DATABASE. REQUEST REJECTED

Explanation: The AFTER keyword may be specified by one of the following:
• A single database name after the keyword DATABASE
• A single table space name or index space name after the keyword SPACENAM.

User response: Specify the command correctly.

DSNT309I   *csect-name object_name* **IS CURRENTLY** *state*.

Explanation: One of the following is already in the requested state:
• A database listed in START DB(list) or STOP DB(list)
• A table space listed in START DB(name) SPACENAM(list) or STOP DB(name) SPACENAM(list)
• An index space listed in START DB(name) SPACENAM(list) or STOP DB(name) SPACENAM(list)
• A partition of a partitioned table space listed in START DB(list) SPACENAM(list) PART(n), STOP DB(list)
  SPACENAM(list) PART(n), START DB(list) SPACENAM(list), or STOP DB(list) SPACENAM(list)

In the message:
• The 'object' is the database, table space, index space or partition number with table space name or the index space
  name.
    – TABLESPACE, if the specified table space was currently in the requested state
    – INDEXSPACE, if the specified index space was currently in the requested state
    – DATABASE, if the database was currently in the requested state
    – PARTITION, if the specified partition of a partitioned table space or partitioned index was currently in the
      requested state
• The 'state' is:
    – START RW, if the specified object is currently started read/write
    – START RO, if the specified object is currently started read only
    – START UT, if the specified object is currently started for utility processing
    – STOPPED, if the specified object is currently stopped or is stopping
    – STOP PENDING, if the specified object currently is in stop pending status.

If there are pages in the logical page list (LPL) for the object, or if the object is in group buffer pool recovery pending
(GRECP) status, a START DATABASE ACCESS(RO) or START DATABASE ACCESS(RW) command triggers a
recovery of that object even if this message is issued for that object. For each object that has LPL or GRECP recovery
that is initiated, DB2 issues a DSNI006 message followed by either a DSNI021 message (when the recovery completes
successfully) or a DSNI005 message (when the recovery is unsuccessful).

User response: No response is necessary. This message is informational only.

DSNT310I   *csect-name* **ALL** *object* **CURRENTLY** *state*.

Explanation: A database and all of its table spaces and index spaces are already in the requested state. This occurred
following the processing of one of these commands:
• START DB(*)
• STOP DB(name)
• START DB(name) SPACENAM(*)

- STOP DB(name) SPACENAM(*).

In this message:

- The 'object' is either:
  - TABLESPACE(S) if the specified table space was currently in the requested state
  - INDEXSPACE(S) if the specified index space was currently in the requested state
  - DATABASE(S) if the database was currently in the requested state.
- The 'state' is either:
  - START RW if the specified table space is currently started read/write
  - START RO if the specified table space is currently started read only
  - START UT if the specified table space is currently started for utility processing
  - STOPPED if the specified table space is currently stopped, has a stop pending against it, or is stopping.

If there are pages in the logical page list (LPL) for the object, or if the object is in group buffer pool recovery pending (GRECP) status, a START DATABASE ACCESS(RO) or START DATABASE ACCESS(RW) command triggers a recovery of that object even if this message is issued for that object. For each object that has LPL or GRECP recovery that is initiated, DB2 issues a DSNI006 message followed by either a DSNI021 message (when the recovery completes successfully) or a DSNI005 message (when the recovery is unsuccessful).

User response: No response is necessary. This message is informational only.

DSNT311I   MESSAGE LIMIT EXCEEDED. DISPLAY IS TERMINATED.

Explanation: The number of messages generated exceeds the maximum number of messages specified in the LIMIT keyword or exceeds the system default.

DSNT312I   *object name* **MUST BE STARTED RW. NO ACTION TAKEN.**

Explanation: The ACCESS keyword is either RO or UT, and the object must be started ACCESS (RW).

In the message, 'object' is either:

- TABLESPACE(S) IN if the START DB(name) SPACENAM(...) is specified, or
- DATABASE if the START DB(name) is specified.

DSNT313I   ACCESS(FORCE) IS VALID ONLY WITH EXPLICIT TABLESPACE OR INDEXSPACE NAMES.
                COMMAND IS REJECTED

Explanation: The ACCESS(FORCE) keyword can be specified when either:

- A single table space or index space name appears after the keyword SPACENAM in a START DATABASE command.
- A list of table space and/or index space names appears after the keyword SPACENAM in a START DATABASE command.

Operator response: Specify the command correctly.

DSNT314I   *csect_name* **INVALID PARTITION SPECIFICATION FOR OBJECT** *object-name*

Explanation: The PART keyword is invalidly specified in a START DATABASE, STOP DATABASE, or DISPLAY DATABASE command.

The PART keyword is invalid if it is specified for:

- A nonpartitioned table space
- A nonpartitioned type 1 index
- A type 2 index not defined on a partitioned table space

The object *object-name* can be either a table space or an index space.

No action for this table space or index space is performed by the START DATABASE, STOP DATABASE or DISPLAY DATABASE command.Any other valid table space or index space specified is processed by the command.

System action: The command is not processed for the table space or index space in question. All other valid table spaces or index spaces specified in the command are processed.

User response: Resubmit the command for the nonpartitioned table space or index space without using the PART keyword.

Problem determination: SYSIBM.SYSTABLESPACE in the DB2 catalog indicates whether a table space is partitioned. SYSIBM.SYSINDEXSPACE in the DB2 catalog indicates whether an index space is partitioned and whether the index space is a type 1 or a type 2 index. Type 2 nonpartitioned indexes can be started, stopped, or displayedat the logical partition level; type 1 cannot.

Severity: 8 (error)

DSNT315I *csect_name* **INVALID PARTITION NUMBER OR RANGE SPECIFIED -** *part-num***FOR OBJECT** *object-name*

Explanation: An invalid partition number or rangewas specified in the PART keyword for a START DATABASE, STOP DATABASE or DISPLAY DATABASE command.

Partition number or range *part-num*is invalid for the object *object-name*.

System action: The part number or rangeis ignored. All other valid partitions specified are processed if PART was specified as list.

User response: Determine the correct partition specification for the table space or index that was the subject of the specified command and resubmit the command.

Problem determination: SYSIBM.SYSTABLESPACE in the DB2 catalog indicates the correct range of partition numbers for the table space. SYSIBM.SYSINDEXSPACE in the DB2 catalog indicates whether an index space is partitioned and whether the index space is a type 1 or a type 2 index. Type 2 indexes can be started, stopped, or displayedat the logical partition level; type 1 cannot.

Severity: 8 (error)

DSNT316I *csect-name* **A LOGICAL PARTITION OF AN INDEX WHICH HAS PAGES IN THE LPL OR HAS BEEN MARKED GRECP CANNOT BE STARTED FOR RW OR RO ACCESS**

Explanation: The START DATABASE command was specified for a logical partition of a nonpartitioned index which has pages in the logical page list (LPL) or is marked group buffer pool recovery pending (GRECP). To start a nonpartitioned index with one of these exception conditions, the START DB command must be specified without the PART keyword.

System action: The command is not processed for the index space. All other valid table spaces or index spaces specified in the command are processed.

User response: Submit the command for the nonpartitioned index space without using the PART keyword.

Problem determination: If more information on the status of the index space is required, use the -DISPLAY DATABASE command.

Severity: 8 (error)

DSNT317I *csect-name* **ANY USER-DEFINED INDEXES ON DSNDB06 COULD NOT BE** *cmd-type* **DUE TO A RESOURCE UNAVAILABLE ON THE CATALOG OR DIRECTORY.**

Explanation: If user-defined indexes exist on DSNDB06, they could not be displayed, started or stopped (depending on *cmd-type*) because the catalog or directory was not available. DB2-defined indexes and table spaces are not affected by this unavailability.

System action: Any user-defined indexes on DSNDB06 were not affected by the command issued. All other table spaces and index spaces were processed by the command.

Programmer response: When you get this message, correct the resource unavailable problem and reissue the command.

DSNT318I   *csect-name* **PLAN=**_plan-name_ **WITH CORRELATION-ID=**_correlation-id_ **CONNECTION-ID=**_connection-id_
LUW-ID=_luw-id_ **THREAD-INFO=**_thread-information_ **TIMED OUT BECAUSE OF A P-LOCK HELD BY**
MEMBER _member-name_

Explanation: A plan cannot get an IRLM lock because the resource is held by a P-lock in the data sharing group,
and the maximum amount of time to wait for the locked resource was exceeded.

csect-name
>   The name of the control section that issued the message.

plan-name
>   The name of the executing plan that is associated with this thread.

correlation-id
>   The correlation identifier of the thread that is associated with the plan identified by *plan-id*. Correlation IDs
>   that begin with values between 001 and 028 identify system agents.

connection-id
>   The connection identifier of the plan identified by *plan-name*.

luw-id   The logical unit of work identifier (LUWID) of *plan-name*.

>   *   The thread does not have a LUWID.

>   *luw-id=token*
>   >   The logical unit of work ID and a five-digit token value. The token value can be used in any DB2
>   >   command that accepts a LUWID as input.

thread-information
>   Information about the thread. The information is presented in a colon-delimited list that contains the
>   following segments:

>   - The primary authorization ID that is associated with the thread.

>   In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily
>   the user's ID.

>   - The name of the user's workstation.

>   - The ID of the user.

>   - The name of the application.

>   - The statement type for the currently executing statement: dynamic or static.

>   - The statement identifier for the currently executing statement, if available. The statement identifier can be
>   used to identify the particular SQL statement. For static statements, the statement identifier correlates to
>   the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement
>   identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.

>   - The name of the role that is associated with the thread.

>   - The correlation token that can be used to correlate work at the remote system with work performed at the
>   DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three
>   components, separated by periods:
>   – A 3 to 39 character IP address.
>   – A 1 to 8 character port address.
>   – A 12 character unique identifier.

>   An asterisk (*) in any segment indicates that the information is not available.

member-name
>   The name of the member of theDB2 subsystem that holds the conflicting P-lock.

System action: Processing continues.

The SQLCA that corresponds to the correlation ID, connection ID, and LUWID contains the name of the resource that
is involved in the timeout and one of the following SQL codes: -904, -911 , -913, or -923.

System programmer response: Identify the member that holds the P-lock and determine why the P-lock is not
being released. If the problem persists and no other method of releasing the P-lock can be found, consider recycling
the member of the group holding the P-lock.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SYSABEND, SYSUDUMP, or SYSMDUMP output.

Related information:

▶ -904 (DB2 Codes)

▶ -911 (DB2 Codes)

▶ -913 (DB2 Codes)

▶ -923 (DB2 Codes)

DSNT331I *csect-name* **DATABASE** *name* **AND ANY SPACE IT CONTAINS CANNOT BE STOPPED. IT IS THE ACTIVE COMMUNICATIONS DATABASE**

Explanation: The database specified in the message cannot be stopped because it is the active communications database. In addition, spaces within the active communications database cannot be stopped.

System action: Nothing is stopped.

Operator response: If the database or any space within it must be stopped, you must stop the distributed data facility (STOP DDF). In a DB2 data sharing environment, you must stop the distributed data facility on all members of the DB2 data sharing group.

DSNT332I **TABLESPACES AND INDEXSPACES ARE UNAVAILABLE BECAUSE DATABASE name IS STOPPED**

Explanation: The database 'name' is currently marked as stopped. Consequently, its table spaces and index spaces are unavailable even though they may have been started.

DSNT350I **-luw-id ACCESSING DATA FOR location**

Explanation: This message follows message DSNT394I or DSNT396I for each thread reported that is participating in distributed data access (database access thread). This additional line reports the distributed identification as follows:

luw-id=token
> The logical-unit-of-work identifier (LUW-ID) is a unique identifier that is assigned to the thread followed by an '=' sign and a token. The token can be used in place of luw-id in any DB2 command that accepts luw-id as input.

location
> The LOCATION name assigned to the requesting site.

The message number and subsystem recognition character are not included in this message.

DSNT352I **AGENT TOKEN** *agent token*

Explanation: When -DISPLAY DATABASE is issued with the LOCKS or CLAIMERS keywords, the agent token will display holders or claimers if the agent information is available. This message immediately follows a DSNT396I or DSNT389I message.

Operator response: This token can be used to cancel a lock holder or claimer via -CANCEL THREAD or to gather more information via -DISPLAY THREAD.

DSNT358I *csect_name* **INVALID PARTITION SPECIFICATION FOR OBJECT** *object-name*

Explanation: Specification of the PART keyword in a DISPLAY DATABASE command is invalid.

The PART keyword is invalid if it is specified for:

- A nonpartitioned table space

- A nonpartitioned type 1 index
- A type 2 index not defined on a partitioned table space

The object *object-name* can be either a table space or an index space.

No action for this table space or index space is performed by the DISPLAY DATABASE command. Any other valid table space or index space specified is processed by the command.

System action: The command is not processed for the table space or index space in question. All other valid table spaces or index spaces specified in the command are processed.

User response: Resubmit the command for the nonpartitioned table space or index space without using the PART keyword.

Problem determination: SYSIBM.SYSTABLESPACE indicates whether a table space is partitioned. SYSIBM.SYSINDEXSPACE indicates whether an index space is partitioned and whether the index space is a type 1 or a type 2 index. Type 2 nonpartitioned indexes can be displayed at the logical partition level; type 1 indexes cannot.

Severity: 8 (error)

DSNT359I   *csect_name* **INVALID PARTITION NUMBER SPECIFIED -** *part-num* **FOR OBJECT** *object-name*

Explanation: An invalid partition number was specified in the PART keyword for a DISPLAY DATABASE command.

Partition number *part-num* is invalid for the object *object-name*.

System action: The part number is ignored. All other valid partitions specified are processed.

User response: Determine the correct partition specification for the table space or index that was the subject of the specified command and resubmit the command.

Problem determination: SYSIBM.SYSTABLESPACE in the DB2 catalog indicates the correct range of partition numbers for the table space. SYSIBM.SYSINDEXSPACE in the DB2 catalog indicates whether an index space is partitioned and whether the index space is a type 1 or a type 2 index. The PART keyword can be used to display the status of logical partitions of type 2 indexes but not type 1 indexes.

Severity: 8 (error)

DSNT360I   ******************************

Explanation: This message consists of 35 asterisks that begin the display of any database and the display of space names for any database.

DSNT361I   *****DISPLAY DATABASE SUMMARY**** report-type-list

Explanation: This message is a multiple-line message that labels the requested set of databases, table spaces, and index spaces.

In the message, 'report-type-list' lists all options specified on the command delimited by spaces. Possible values are:

ACTIVE
          All allocated databases are listed.

GLOBAL
          All requested database information is listed. This title is produced if ACTIVE, DISABLED, and RESTRICT are omitted.

LOCKS
          The connection identifiers and correlation identifiers of all applications holding, retaining, or waiting for locks on the specified table spaces or index spaces are displayed.

RESTRICT
          All databases with restricted usage (that is, RO, started UT) are listed.

USE   The connection identifiers and correlation identifiers of all allocated applications are listed.

DSNT362I

CLAIMERS
> The connection identifiers and correlation identifiers of all applications holding claims on the specified database are displayed.

LPL The page number ranges of the pages that are included in the logical page list for the table space or index space.

OVERVIEW
> All table spaces and index spaces that are contained within the databased that are specified.

DSNT362I   DATABASE = *database-name* **STATUS =** *status* **DBD LENGTH =** *dbd-length*

Explanation: This message signals the beginning of display information from the DISPLAY DATABASE command. It lists the name, status, and DBD length of the database.

database-name
> The name of the database.

status   The status of the database. The value can be one or more of the following strings delimited by a comma. The maximum length of the string is 18 characters.

> **RO**   The database is started for read activity only.
>
> **RW**   The database is started for read and write activity.
>
> **STOP**   The database is stopped.
>
> **STOPP** A stop operation for this database is in progress, is pending or has failed.
>
> **UT**   The database is started for utility processing only.
>
> **UTRO** A utility that allows only RO access is in process on the database. If the utility was canceled before the object was drained, the object can allow SQL access because the object was not altered by the utility.
>
> **UTRW** A utility that allows RW access is in process on the database.
>
> **UTUT** A utility that allows only UT access is in process on the database. If the utility was canceled before the object was drained, the object can allow SQL access because the object was not altered by the utility.

dbd-length
> The length of the database descriptor (DBD) in bytes.

System action: Processing continues.

Message DSNT362I is followed by messages DSNT392I, DSNT394I, or DSNT396I. All table space and index space names that are listed in those messages are contained within the database *database-name*.

Administrator response: If the object seems to be in a persistent STOPP status, check whether the STOP operation is still in progress. You can check this progress by looking at message DSNI003I or output from the DISPLAY THREAD command with the TYPE(SYSTEM) option.

- Message DSNI003I lists objects that could not be stopped.
- In the DISPLAY THREAD output, look for message DSNV501I with STOP DATABASE listed as the command that is executing. If you see one of these messages, the STOP DATABASE command is still in progress for the thread that is identified in DSNV402I.

If the STOP operation has timed out or failed, resolve the error by using the START DATABASE command.

Related reference:

➡ -DISPLAY DATABASE (DB2) (DB2 Commands)

➡ -DISPLAY THREAD (DB2) (DB2 Commands)

➡ -START DATABASE (DB2) (DB2 Commands)

Related information:
"DSNI003I" on page 213
"DSNT392I" on page 522
"DSNT394I" on page 523
"DSNT396I" on page 525
"DSNV401I" on page 942

DSNT363I   PHYSICAL ERROR PAGE RANGE FOR DATA SET data-set-name LOW=lowphy, HIGH=hiphy

Explanation: This message is the response to a -DISPLAY DATABASE command when the SPACENAM keyword is specified in order to notify the user of any physical I/O errors encountered on a data set.

In the message, 'data-set-name' is the name of the data set with errors. The variables 'lowphy' and 'hiphy' identify the range of physical pages, within the data set, that were being read when the I/O errors occurred. Physical page numbers are assigned in relationship to the beginning of the data set in which the pages reside.

This message is always preceded by messages DSNT360I and DSNT361I, which identify the name of the database and page set that the specified data set supports.

Page sets (table spaces or index spaces) are made up of supporting data sets. Physical pages are pages in supporting data sets that are numbered from the beginning of the particular data set in which they reside. This message refers to physical pages in a data set that support (are part of) some page set.

System action: See the Explanation section for this message.

DSNT364I   _csect-name_ KEYWORD _ONLY_ IS USED INCORRECTLY

Explanation: Keyword _ONLY_ is used incorrectly in the DISPLAY DATABASE command. When the SPACENAM keyword is specified, keyword _ONLY_ is valid only if one of the following keywords is also used:
- LOCKS
- LPL
- CLAIMERS
- WEPR
- USE

System action: The DISPLAY DATABASE command is not processed.

User response: Add one of the keywords mentioned above in the explanation to the DISPLAY DATABASE command and issue the command again.

DSNT365I   NO DATABASES FOUND

Explanation: This message is produced when a DISPLAY DB(name) is entered and no databases were found to satisfy the request.

System action: The process continues normally.

DSNT366I   DATABASE name NOT type

Explanation: A database listed in the DISPLAY DB(...) is not of the correct type. The variables are as follows:

name   The name of the object.

type   The type of request specified: ACTIVE, RESTRICTED, ACTIVE OR RESTRICTED

Note that either DSNT301I or DSNT302I is produced if no object is found and the keywords ACTIVE and RESTRICT are omitted.

System action: Processing continues normally.

DSNT367I   **NO INFORMATION AVAILABLE**

Explanation: This message indicated there is no LOCK, USE, LPL, WEPR or CLAIMER information available when DISPLAY DB(...) SPACENAM(...) ONLY is entered with the keywords LOCKS, USE, LPL, WEPR or CLAIMERS.

System action: The process continues normally.

DSNT368I   *csect-name* **MORE THAN ONE MUTUALLY EXCLUSIVE KEYWORD WAS SPECIFIED IN THE DISPLAY DATABASE COMMAND.**

Explanation: The keywords, LOCKS, LPL, CLAIMERS, WEPR and USE are mutually exclusive. Only one should be specified in a DISPLAY DATABASE command. The precedence order for which keyword takes precedence over the others is:

1. LPL
2. CLAIMERS
3. LOCKS
4. USE
5. WEPR

System action: The DISPLAY DATABASE command is processed with this warning message.

User response: Remove keywords from DISPLAY DATABASE command until only one of these mutually exclusive keywords is specified.

DSNT375I   **PLAN=***plan-name1* **WITH CORRELATION-ID=***correlation-id1* **CONNECTION-ID=***connection-id1* **LUW-ID=***luw-id1* **THREAD-INFO=***thread-information1* **IS DEADLOCKED WITH PLAN=***plan-name2* **WITH CORRELATION-ID=***correlation-id2* **CONNECTION-ID=***connection-id2* **LUW-ID=***luw-id2* **THREAD-INFO=** *thread-information2* **ON MEMBER** *member-name*

Explanation: A plan has been denied an IRLM lock because of a detected deadlock.

plan-name1 **and** *plan-name2*
> The names of the executing plans.

> *plan-name1*
>> The plan that was denied an IRLM lock.

> *plan-name2*
>> One of the members of the deadlock.

>> DB2 does not attempt to identify all survivors of a deadlock or all participants in a deadlock in this message. This plan might be only one of several plans that are holding locks on the resource.

>> If multiple deadlocks are detected during the same deadlock-detection interval, the lock holder identified by *plan-name2* in one instance of this message might be selected to break a deadlock identified by a different instance of this message. The lock holder identified by *plan-name2* is not guaranteed to be a winner in all deadlocks in which it participates.

> If you receive a system plan in this message for either *plan-name1* or *plan-name2*, see the information about system plans in message DSNT376I.

correlation-id1 **and** *correlation-id2*
> The correlation identifiers of the threads that are associated with, respectively, *plan-name1* and *plan-name2*. Correlation IDs that begin with values between 001 and 028 identify system agents.

connection-id1 **and** *connection-id2*
> The connection identifiers of, respectively, *plan-name1* and *plan-name2*.

luw-id1 **and** *luw-id2*
> The logical unit of work identifiers (LUWIDs) of, respectively, *plan-name1* and *plan-name2*.

>   *   The thread does not have a LUWID.

> *luw-id=token*
> > The logical unit of work ID and a five-digit token value. The token value can be used in any DB2 command that accepts a LUWID as input.

thread-information1 **and** *thread-information2*
> Information about the threads for, respectively, *plan-name1* and *plan-name2*. The information is presented in a colon-delimited list that contains the following segments:

> * The primary authorization ID that is associated with the thread.

>   In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.

> * The name of the user's workstation.
> * The ID of the user.
> * The name of the application.
> * The statement type for the currently executing statement: dynamic or static.
> * The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.
> * The name of the role that is associated with the thread.
> * The correlation token that can be used to correlate work at the remote system with work performed at the DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, separated by periods:
>   - A 3 to 39 character IP address.
>   - A 1 to 8 character port address.
>   - A 12 character unique identifier.

> An asterisk (*) in any segment indicates that the information is not available.

member-name
> The name of the member of the DB2 subsystem on which the thread associated with *plan-id2* is executing.

System action: The SQLCA that corresponds to the correlation ID, connection ID, and LUWID for *plan-name1* contains the name of the resource that is involved in the deadlock and one of the following SQL codes: -904, -911 , -913, or -923.

DB2 can recover from a lock denial in some cases and can complete successfully. In such a case, the SQLCA is not filled in. One example is during setup of performance enhancements during bind time for execution time.

Problem determination: A DB2 IFCID 172 trace record contains the information needed to determine all participants in, and survivors of, a given deadlock.

Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
* Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
* IRLM trace output.

.

DSNT376I

Related concepts:

⬅️ Lock contention (DB2 Performance)

Related information:

"DSNT376I"

⬅️ -904 (DB2 Codes)

⬅️ -911 (DB2 Codes)

⬅️ -913 (DB2 Codes)

⬅️ -923 (DB2 Codes)

DSNT376I **PLAN=**_plan-name1_ **WITH CORRELATION-ID=**_correlation-id1_ **CONNECTION-ID=**_connection-id1_ **LUW-ID=**_luw-id1_ **THREAD-INFO=**_thread-information1_ **IS TIMED OUT. ONE HOLDER OF THE RESOURCE IS PLAN=**_plan-name2_ **WITH CORRELATION-ID=**_correlation-id2_ **CONNECTION-ID=**_connection-id2_ **LUW-ID=**_luw-id2_ **THREAD-INFO=** _thread-information2_ **ON MEMBER** _member-name_

Explanation: A plan has been denied an IRLM lock because of a _timeout_. The maximum amount of time to wait for a locked resource was exceeded.

plan-name1 **and** _plan-name2_
> The names of the executing plans.

> _plan-name1_
>> The plan that was denied an IRLM lock.

> _plan-name2_
>> One of the plans holding a lock on the resource. The listed plan is not necessarily responsible for the timeout condition.

> Several different system plans can appear in _plan-name1_ and _plan-name2_. These system plans are not documented in SYSIBM.SYSPLAN. If you receive a system plan in this message, you are in contention with another user. If you are not doing an automatic rebind, message DSNT500I or DSNT501I might accompany this message and identify that resource. If you need additional information, see the SQLCA and any additional bind messages. (If you are performing an automatic rebind, the SQLCA is not available.) The system plans are:
> - ACT
> - BCT
> - DISTSERV
> - DSNBIND
> - STARTCT
> - DSNUTIL

correlation-id1 **and** _correlation-id2_
> The correlation identifiers of the threads that are associated with, respectively, _plan-name1_ and _plan-name2_. Correlation IDs that begin with values between 001 and 028 identify system agents.

connection-id1 **and** _connection-id2_
> The connection identifiers of, respectively, _plan-name1_ and _plan-name2_. If the execution of the plan was switched by DB2 to a system agent, the connection identifier is the DB2 subsystem name.

luw-id1 **and** _luw-id2_
> The logical unit of work identifiers (LUWIDs) of, respectively, _plan-name1_ and _plan-name2_.

> *   The thread does not have a LUWID.

> _luw-id=token_
>> The logical unit of work ID and a five-digit token value. The token value can be used in any DB2 command that accepts a LUWID as input.

thread-information1 **and** *thread-information2*

Information about the threads for, respectively, *plan-name1* and *plan-name2*. The information is presented in a colon-delimited list that contains the following segments:

- The primary authorization ID that is associated with the thread.

  In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.

- The name of the user's workstation.

- The ID of the user.

- The name of the application.

- The statement type for the currently executing statement: dynamic or static.

- The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.

- The name of the role that is associated with the thread.

- The correlation token that can be used to correlate work at the remote system with work performed at the DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, separated by periods:
  - A 3 to 39 character IP address.
  - A 1 to 8 character port address.
  - A 12 character unique identifier.

An asterisk (*) in any segment indicates that the information is not available.

member-name

The name of the member of the DB2 subsystem on which the thread associated with *plan-id2* is executing.

System action: Processing continues.

The SQLCA that corresponds to the correlation ID, connection ID, and LUWID for *plan-name1* contains the name of the resource that is involved in the timeout and one of the following SQL codes: -904 or -923.

DB2 can recover from a lock denial in some cases and can complete successfully. In such a case, the SQLCA is not filled in. One example is during setup of performance enhancements during bind time for execution time.

Operator response: Notify the system programmer.

System programmer response: To resolve the timeout, follow the instructions in Investigating and resolving timeout situations (DB2 Performance). See also SQLCODEs -911 and -913.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).

- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

- Listing of DSNZPARM module used.

- IRLM trace output.

DSNT377I

Related concepts:

➡ Automatic rebinding (DB2 Application programming and SQL)

Related tasks:

➡ Programming applications for performance (DB2 Performance)

Related information:

"DSNT500I" on page 555

"DSNT501I" on page 556

➡ -904 (DB2 Codes)

➡ -911 (DB2 Codes)

➡ -913 (DB2 Codes)

➡ -923 (DB2 Codes)

DSNT377I   **PLAN=***plan-name* **WITH CORRELATION-ID=***correlation-id* **CONNECTION-ID=***connection-id* **LUW-ID=***luw-id* **THREAD-INFO=***thread-information* **IS IN CONFLICT WITH AN INDOUBT THREAD ON MEMBER** *member-name*

Explanation: A plan cannot gain an IRLM lock because a required resource is currently undergoing recovery.

plan-name
> The name of the executing plan that is associated with this thread.
>
> If you receive a system plan in this message, see the information about system plans in message DSNT376I.

correlation-id
> The correlation identifier of the thread that is associated with the plan identified by *plan-id*. Correlation IDs that begin with values between 001 and 028 identify system agents.

connection-id
> The connection identifier of the plan identified by *plan-name*.

luw-id   The logical unit of work identifier (LUWID) of *plan-name*.

> *   The thread does not have a LUWID.

> *luw-id=token*
>> The logical unit of work ID and a five-digit token value. The token value can be used in any DB2 command that accepts a LUWID as input.

thread-information
> Information about the thread. The information is presented in a colon-delimited list that contains the following segments:
>
> - The primary authorization ID that is associated with the thread.
>
>   In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.
>
> - The name of the user's workstation.
> - The ID of the user.
> - The name of the application.
> - The statement type for the currently executing statement: dynamic or static.
> - The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.
> - The name of the role that is associated with the thread.
> - The correlation token that can be used to correlate work at the remote system with work performed at the DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, separated by periods:
>   – A 3 to 39 character IP address.

– A 1 to 8 character port address.

– A 12 character unique identifier.

An asterisk (*) in any segment indicates that the information is not available.

member-name

The name of the member of theDB2 subsystem on which the indoubt thread is executing.

This message is issued by the service controller subcomponent.

System action: Processing continues.

The SQLCA that corresponds to the correlation ID, connection ID, and LUWID of the plan identified by *plan-name* contains the name of the resource that is involved in the timeout and one of the following SQL codes: -904, -911 , -913, or -923.

Problem determination: Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).

* SYSABEND, SYSUDUMP, or SYSMDUMP output.

Related information:

"DSNT376I" on page 512

▶ -904 (DB2 Codes)

▶ -911 (DB2 Codes)

▶ -913 (DB2 Codes)

▶ -923 (DB2 Codes)

DSNT378I   *csect-name* **PLAN=***plan-name* **WITH CORRELATION-ID=***correlation-id* **CONNECTION-ID=***connection-id* **LUW-ID=***luw-id* **THREAD-INFO=***thread-information* **TIMED OUT BECAUSE OF A RETAINED LOCK HELD BY MEMBER** *member-name*

Explanation: A plan cannot get an IRLM lock because the resource is held by a retained lock on behalf of another member in the data sharing group, and the amount of time to wait for the locked resource was exceeded. The member holding the retained lock has either failed, or is in the process of restarting to recover its retained locks.

plan-name

The name of the executing plan that is associated with this thread.

correlation-id

The correlation identifier of the thread that is associated with the plan identified by *plan-id*. Correlation IDs that begin with values between 001 and 028 identify system agents.

connection-id

The connection identifier of the plan identified by *plan-name*.

luw-id   The logical unit of work identifier (LUWID) of *plan-name*.

*   The thread does not have a LUWID.

luw-id=token

The logical unit of work ID and a five-digit token value. The token value can be used in any DB2 command that accepts a LUWID as input.

thread-information

Information about the thread. The information is presented in a colon-delimited list that contains the following segments:

* The primary authorization ID that is associated with the thread.

  In many distributed configurations, the primary authorization ID that is used with DB2 is not necessarily the user's ID.

* The name of the user's workstation.

DSNT379I

- The ID of the user.
- The name of the application.
- The statement type for the currently executing statement: dynamic or static.
- The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement. For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table. For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.
- The name of the role that is associated with the thread.
- The correlation token that can be used to correlate work at the remote system with work performed at the DB2 subsystem. The correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, separated by periods:
  - A 3 to 39 character IP address.
  - A 1 to 8 character port address.
  - A 12 character unique identifier.

An asterisk (*) in any segment indicates that the information is not available.

member-name
  The name of the member of the failed or restarting DB2 subsystem that holds the conflicting retained lock.

This message is issued only if the RETLWAIT installation parameter in DSN6SPRM is set to YES. If RETLWAIT is set to NO, which is the default setting, then any IRLM lock that conflicts with a retained lock will not wait, but instead will be rejected immediately. (See reason codes 00C900BE and 00C20254.)

System action: Processing continues.

The SQLCA that corresponds to the correlation ID, connection ID, and LUWID of the plan identified by *plan-name* contains the name of the resource that is involved in the timeout and one of the following SQL codes: -904, -911 , -913, or -923.

System programmer response: Restart the member holding the retained lock, if it has not already been restarted.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- SYSABEND, SYSUDUMP, or SYSMDUMP output.

Related information:

➚ -904 (DB2 Codes)

➚ -911 (DB2 Codes)

➚ -913 (DB2 Codes)

➚ -923 (DB2 Codes)

➚ 00C20254 (DB2 Codes)

➚ 00C900BE (DB2 Codes)

DSNT379I **CANNOT IDENTIFY TO IRLM. INVALID SSNAME OR IRLM IS NOT CURRENTLY ACTIVE**

Explanation: Either an attempt was made to identify the IRLM with a subsystem name not recognized by MVS, or the IRLM is not currently active. This message is also issued if common service area (CSA) storage is insufficient.

System action: DB2 abends with a reason code of X'04E'.

Operator response: Notify system programmer.

System programmer response: This is probably a user error. Change the IRLMSID parameter in the DB2 DSN6SPRM initialization macro to match the MVS subsystem table, or start IRLM if IRLM is not already up. See return code 00E30079 for further information about determining the cause of the error.

DSNT380I   CANNOT IDENTIFY TO IRLM, SSNAME ALREADY IN USE

Explanation: Either DB2 previously terminated and did not successfully sign off from IRLM, or another subsystem is identified to IRLM with this installation-defined SSNAME. This message accompanies abend reason code X'00E30071'.

System action: DB2 is abended with an abend code of X'04E'.

Operator response: Notify the system programmer.

System programmer response: Either manually stop IRLM, or change the installation-defined SSNAME.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.
- IRLM trace output.

.

DSNT381I   IRLM QUIT REQUEST FAILED, RC = *reason-code* **, SC =** *sub-code*

Explanation: During the QUIT (or signoff) request to the IMS/VS Resource Lock Manager, a return code and subcode were returned indicating an unsuccessful completion of the request.

System action: This message indicates that the DB2 subsystem shutdown could not be completed normally.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of the failure from the return code and subcode, and correct the error. DB2 has terminated but has not signed off from IRLM. It will be necessary to issue the 'F irlmproc, ABDUMP' command to stop the IRLM. This must be done before issuing another -START DB2 command.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.
- IRLM trace output.

Related concepts:

▶ IRLM return and reason codes (IRLM)

DSNT382I   IRLM AUTO-START FAILED

Explanation: During the IDENTIFY to IRLM, the AUTO-START of IRLM using SVC 34 failed. This message accompanies abend code X'00E30080'.

System action: DB2 is terminated with an abend code of X'04E'.

Operator response: Notify the system programmer.

System programmer response: Determine the cause of failure from the SVC 34 return code preserved in register 2.

DSNT383I   *csect-name* IRLM AUTO-STOP FAILED

Explanation: During a sign off from IRLM caused by either DB2 normal shutdown (CSECT DSNTLQUI) or DB2 subsystem termination (CSECT DSNTLCLE), an AUTO-STOP of IRLM using SVC 34 failed. If this occurred during normal sign off, this message accompanies abend reason code X'00E30075'.

System action: DB2 abends.

Operator response: Notify the system programmer.

System programmer response: Manually stop IRLM.

Problem determination: Determine the cause of failure from the SVC return code preserved in register 2.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

.

DSNT384I   IRLM AUTO-START WAIT TIME EXCEEDED

Explanation: An attempt was made to automatically start IRLM. However, the wait time specified was exceeded.

System action: DB2 is abended.

Operator response: Notify the system programmer.

System programmer response: Either manually start IRLM or modify the installation-defined auto-start wait time. Restart DB2.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of DSNZPARM module used.

.

Check the IRLMID given in the IRLM procedure against the IRLMSID in macro DSN6SPRM which was in the DSNZPARM used at startup.

DSNT386I   NAME TYPE PARTS

Explanation: This message forms the headings of a display for a set of messages. This message is always followed by multiple DSNT387I messages. These DSNT387I messages are used to display information that is received as a result of a DISPLAY DATABASE command when the keyword OVERVIEW is specified. Refer to message DSNT387I for a description of the data these headings identify.

DSNT387I   *space type parts*

Explanation: This message contains the results from a DISPLAY DATABASE command when the keyword OVERVIEW is specified. This message contains data that describes the status of a tablespace or index space in the database that is listed in the preceding DSNT362I message. The data in this message is always preceded by the column headings in message DSNT386I. The message number for this message is not displayed. However, messages DSNT386I, DSNT387I, and DSNT390I are always produced with a multiple line WTO and are never interrupted.

space   Table space name, index space name, or table OBID.

type   TS for a table space, IX for an index space, LS for a LOB tablespace, TB for a table, or XS for XML table space

parts   Indicates the number of partitions in the table space, index space, LOB table space, or XML table space. It is blank for a simple table space, simple index space, or segmented table space.

DSNT388I   NAME TYPE PART STATUS CONNID CORRID CLAIMINFO

Explanation: This message is the heading for a set of messages that display information as a result of a -DISPLAY DATABASE command when the keyword CLAIMERS is specified. The message number for this message is not displayed. This message is always followed by multiple DSNT389I messages containing the column data. Refer to message DSNT389I for a description of the data these headings identify.

DSNT389I   *space type part status conn-id corr-id claims*

Explanation: This message contains the display information that results from a -DISPLAY DATABASE command when the keyword CLAIMERS is specified. This message contains the column data describing the status of a table space or index space in the database identified in the preceding DSNT362I message. The column headings that correspond to these values display as message DSNT388I.

space   The table space name or index space name.

type   An indication of object type:

> **IX, IXB*n*, or IXC*n***
> Index space
>
> **LS, LSB*n*, or LSC*n***
> LOB table space
>
> **TS, TSB*n*, or TSC*n***
> Table space

> If the object is involved in cloning, then the object type is appended with a B (to identify a base object) or a C (to identify a clone object) followed by a data set instance number. For example, a base object that is no longer involved with cloning has B2 appended to the type indicator if it is currently represented by an instance number 2 data set. A base object that is no longer involved with cloning has no suffix.

> The CLONE column in the SYSIBM.SYSTABLESPACE catalog table indicates if a table space, and therefore its objects, is involved with cloning. A CLONE='Y' value indicates that cloning is active. The INSTANCE column value indicates the data set instance number for the current base objects.

part   The partition number. For a nonpartitioned table space or an index on a nonpartitioned table space, it is blank. For nonpartitioned indexes on a partitioned table space, it is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value will be the character D (for example, D0001).

status   Blank, or one or more of the following values, delimited by commas. The maximum length of the string is 17 characters.

> In the final summary line for partitioned objects where *space* is either blank or L*, *status* can be only blank, DBETE or DBETE,RECP.

> **ACHKP**
> The table space is in aux check pending state. ACHKP on a table space indicates that CHECK DATA detected an orphan or missing LOB or an out-of-sync condition between the row in the base table and the LOB in the LOB table space.

> **ARBDP**
> Advisory rebuild pending status is set when indexes with varying length columns are changed from VARYING NO to VARYING YES.

> **AREO***   The table space is placed in advisory reorg status, possibly because of a change in the table or index information. It also occurs because the table space, index, or partition identified in this status should be reorganized for optimal performance.

> **AREST**   The object (a table space, index space, or a physical partition of a table space or index space) is in an advisory restart pending state. If backout activity against the object is not already underway, it should be initiated either by issuing the RECOVER POSTPONED command or by recycling the system with the system parameter LBACKOUT=AUTO.

> **AUXW**   The table space is in auxiliary warning status. AUXW on a LOB table space indicates that one or

more LOBs in the LOB table space is invalid. AUXW on a base table space indicates that CHECK DATA detected an orphan or missing LOB or an out-of-sync condition between the row in the base table and the LOB in the LOB table space.

CHKP The object (a table space or a partition within a table space) is in check pending status.

COPY The object (a table space or a partition within a table space) is in copy pending status. An image copy is required for this object.

DBETE Either:

- The DBET status for the object (a table space, a partition within a table space, an index space, an index partition, or a logical index partition) is in error. The error occurred when the DBET states were being modified during log apply or must-complete processing. This condition prohibited DB2 from successfully updating the DBET states.

- DB2 encountered an unexpected page set access error during restart, unrelated to processing a DBET exception state.

GRECP

The object (a table space, a table space partition, index space, or index space partition) is marked recovery pending because of the failure of a group buffer pool. This status value appears only on spaces. It does not apply to databases.

ICOPY The object is in informational copy pending status.

LPL The object has pages or ranges of pages that are unavailable because of logical or physical damage.

LSTOP The logical partition is stopped.

PRO The table space partition is in Persistent Read Only restricted status. Read access to the table space partition is allowed, but updates are prohibited.

PSRBD

The index space is in page set rebuild pending status.

RBDP The object (an index space, index partition or logical index partition) is in rebuild pending status. For logical partitions, the RBDP status can appear as RBDP*.

RBDP* Indicates that the entire index is inaccessible to SQL applications. However, only the logical partition needs to be rebuilt.

RECP The object (a table space, table space partition, index space, index partition or logical index partition) is in recovery pending status.

REFP The object (a table space, a table space partition, index space, or index space partition) is in refresh pending status. For a table space, run LOAD REPLACE or the RECOVER utility to resolve this restricted status, or use the REBUILD INDEX utility to rebuild the index from the data.

REORP

The object (a table space or table space partition) is in REORG-pending status.

RESTP The object is in the Restart Pending state. Backout activity against the object must be completed by using the -RECOVER POSTPONED command, or by recycling the system with system parameter LBACKOUT=AUTO.

RO The table space, table space partition, index space or index space partition is started for read-only activity.

RW The table space, table space partition, index space or index space partition is started for read and write activity.

STOP The table space, table space partition, index space or index space partition is stopped.

STOPE The table space or index space was implicitly stopped because there is a problem with the log RBA in a page. Message DSNT500I or DSNT501I is issued when the error is detected to indicate the inconsistency.

STOPP A stop is pending for the database, table space, table space partition, index space or index space partition.

UT The table space, table space partition, index space or index space partition is started for utility processing only.

UTRO A utility is in process on the table space, table space partition, index space, or index space partition that allows only RO access.

UTRW A utility is in process on the table space, table space partition, index space or index space partition that allows RW access.

UTUT A utility is in process on the table space, table space partition, index space or index space partition that allows only UT access.

WEPR This object has pages or ranges of pages that are unavailable because of physical damage. To reset this flag, you can use either the RECOVER utility with the option as ERROR RANGE or the LOAD utility with the option as REPLACE.

If the status for a nonpartitioned index is blank, then the status information is displayed at the logical partition level. If the type is TB, the status is blank.

conn-id A connection identifier for the thread.

corr-id A correlation identifier of the thread associated with the space name.

claims The type and duration of currently held claims. The claim is composed of a claim class descriptor and a claim duration descriptor.

The claim class descriptor has one of the following values:

CS Cursor stability claim class

RR Repeatable-read claim class

WR Write claim class

The claim duration descriptor describes the duration of the claims as follows:

A The claim is held until deallocation.

C The claim is held until the next commit point.

H The claim is held across commit.

Related concepts:

⯈ Failure scenarios (DB2 Data Sharing Planning and Administration)

⯈ Recovering data (DB2 Data Sharing Planning and Administration)

Related reference:

⯈ Advisory or restrictive states (DB2 Utilities)

⯈ -DISPLAY DATABASE (DB2) (DB2 Commands)

⯈ -RECOVER POSTPONED (DB2) (DB2 Commands)

⯈ LOAD (DB2 Utilities)

⯈ REBUILD INDEX (DB2 Utilities)

⯈ RECOVER (DB2 Utilities)

⯈ Data sharing status descriptions (DB2 Data Sharing Planning and Administration)

⯈ SYSIBM.SYSTABLESPACE table (DB2 SQL)

Related information:

DSNT390I   * DISPLAY OF DATABASE name action*

Explanation: This message indicates the end of a set of DSNT391I and DSNT392I messages, a set of DSNT393I and DSNT394I messages, or a set of DSNT395I and DSNT396I messages.

In this message:

• The 'name' is the name of the database.

• The 'action' is one of the following:

CONTINUES
     The remaining table space and index space names within this database are continued to a new display.

TERMINATED
     The remaining table space and index space names within this database are terminated.

ENDS   The display for this database is complete.

DSNT391I   NAME TYPE STATUS PHYERRLOW PHYERRHI CATALOG PIECE

Explanation: This message indicates the beginning of a set of messages used to display information as a result of a DISPLAY DATABASE command when the keywords WEPR, LOCKS, LPL, CLAIMERS and USE are omitted. (The message appears in one continuous line across the screen, not as shown above.) This message forms the headings of the display and is always followed by multiple DSNT392I messages containing the column data. Refer to message DSNT392I for a description of the data these headings identify.

DSNT392I   *space type part status lo-page hi-page cat-name piece*

Explanation: This message contains the display information as a result of a DISPLAY DATABASE command when the keywords LOCK, CLAIMERS, LPL and USE are omitted. This message contains the column data describing the status of a table space or index space in the database listed in the preceding DSNT362I message. This message is always preceded by its column headings in message DSNT391I. The message number for this message is not displayed. However, messages DSNT391I, DSNT392I, and DSNT390I are always produced with a multiple line WTO and are never interrupted.

In this message:

- The *space* is the table space name or index space name.
- The *type* is TS for a table space, IX for an index space or LS for a LOB table space.

  If the object is involved in cloning then the object *type* will be appended with a 'B' (base object) or a 'C' (clone object) along with a data set instance number. A base object no longer involved with cloning will have a 'B2' appended to it's *type* if it's currently represented by an instance number 2 data set. A base object no longer involved with cloning, that is represented by instance number 1 data sets, will not have the 'B' and instance number suffix.

  Refer to the SYSIBM.SYSTABLESPACE catalog table CLONE column to determine if a table space, and therefore it's objects, are involved with cloning. A CLONE='Y' value indicates that cloning is active. The INSTANCE column value indicates the data set instance number for the current base objects.

- The *part* is the partition number. It is blank for a simple table space, simple index space, or segmented table space. For nonpartitioned secondary indexes, it is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value will be the character D (for example, D0001).
- See DSNT389I for a list of the possible *status* display strings.
- When a data set supporting the specified table space or index space has physical I/O errors, *lo-page* is a hexadecimal number identifying the lowest page number containing an I/O error. This column is blank if there are no physical errors.
- When a data set supporting the specified table space or index space has physical I/O errors, *hi-page* is a hexadecimal number identifying the highest page number containing an I/O error. This column is blank if there are no physical errors.
- When a data set supporting the specified table space or index space has physical I/O errors, *cat-name* identifies the catalog owning the data set with the I/O errors. This column is blank if there are no physical errors.
- When a data set supporting the specified table space or index space has physical I/O errors, *piece* is an integer number identifying the piece within the table space or index space having the error. This column is blank if there are no physical errors.

  The piece number can be used to construct the name of the data set supporting the space containing the physical error as follows:

```
cat-name.DSNDBC.databas
name.space-name.I0001.Axxxx
```

  The value xxxx is the piece number plus one and database-name is the name displayed in the preceding DSNT362I message. If the table space or index space is partitioned, the partition number should equal the piece number plus one.

DSNT393I   NAME TYPE PART STATUS CONNID CORRID USERID

Explanation: This message indicates the beginning of a set of messages used to display information as a result of a DISPLAY DATABASE command when the keyword USE is specified. This message contains the headings of the display. (The message appears in one continuous line across the screen, not as shown above.) This message is always followed by multiple DSNT394I messages containing the column data. Refer to message DSNT394I for a description of the data these headings identify.

DSNT394I   *space type part status conn-id corr-id user-id*

Explanation: This message contains the display information resulting from a DISPLAY DATABASE command when the keyword USE is specified. This message contains the column data describing the status of a table space or index space in the database listed in the preceding DSNT362I message. This message is always preceded by message DSNT393I containing the column headings. The message number for this message is not displayed. However, messages DSNT393I, DSNT394I, and DSNT390I are always produced with a multiple line WTO and are never interrupted.

In this message:

- The *type* is TS for a table space, IX for an index space or LS for a LOB table space.

  If the object is involved in cloning then the object *type* will be appended with a 'B' (base object) or a 'C' (clone object) along with a data set instance number. A base object no longer involved with cloning will have a 'B2'

appended to it's *type* if it's currently represented by an instance number 2 data set. A base object no longer involved with cloning, that is represented by instance number 1 data sets, will not have the 'B' and instance number suffix.

Refer to the SYSIBM.SYSTABLESPACE catalog table CLONE column to determine if a table space, and therefore it's objects, are involved with cloning. A CLONE='Y' value indicates that cloning is active. The INSTANCE column value indicates the data set instance number for the current base objects.

- The *type* is TS for a table space or IX for an index space.
- The *part* is the partition number. It is blank for a simple table space, simple index space, or segmented table space. For nonpartitioned secondary indexes, it is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value will be the character D (for example, D0001).
- See DSNT389I for a list of the possible *status* display strings.
- The *conn-id* is a connection identifier for the thread. It is one of the following:
  - BATCH if the thread is from a batch region.
  - TSO if the thread is from a TSO terminal.
  - UTILITY if the thread is from a utility.
  - 'IMS identifier' if the thread is from an IMS terminal.
  - 'CICS identifier' if the thread is from a CICS terminal.
  - CONSOLE if the thread is from the console.
- The *corr-id* is a correlation identifier of the thread associated with the space name. It is one of the following:
  - 'job name' if the thread is from a batch region.
  - 'TSO logon identifier' if the thread is from a TSO terminal.
  - PST#.PSBNAME if the thread is from an IMS terminal.
  - 'entry ID.thread number.tran id' if the thread is from a CICS terminal.
  - CONSOLE if the thread is from the console.
  - '0nn', where 'nn' is a number , if the thread is for a system agent.
  - For distributed database access threads, this field contains the following:
    - While the thread is being created:
      1. '028.DBAA nn', where nn ranges from 01 to 99. This indicates that the database access agent (DBAA) is in its connection processing, which consists of establishing the DBAA thread, and validating and verifying the user's ID.

         All RACF requests are serialized on a single DB2 RACF service task, which might increase the validation time when multiple DBAAs are being created concurr
    - After the thread is created:
      1. For threads using system-directed access and threads using application-direct access from a DB2 requester, this variable displays the correlation ID of the thread at the requesting location.
      2. For threads using application-directed access from a non-DB2 requester, this variable displays the first twelve characters in the DDM external name (EXTNAM) parameter of the DDM EXCSAT command received as part of the SQL CONNECT
- The *user-id* is one of the following:
  - 'TSO logon identifier' if the thread is from a TSO terminal.
  - 'sign-on id' or the LTERM if the thread is from an IMS message-drive region.
  - AXBUSER if the thread is from an IMS message-driven region and RACF is used.
  - 'PSB name' if the thread is from an IMS nonmessage-driven region and RACF is not used.
  - 'user-id', 'term id', 'tran id' or other, as specified in the resource control table (RCT), if the thread is from CICS.

Related concepts:

⮕ DB2 subsystem access control (Managing Security)

Related reference:

⮕ -DISPLAY DATABASE (DB2) (DB2 Commands)

⮕ SYSIBM.SYSTABLESPACE table (DB2 SQL)

DSNT395I   NAME TYPE PART STATUS CONNID CORRID LOCKINFO

Explanation: This message indicates the beginning of a set of messages used to display information as a result of a DISPLAY DATABASE command when the keyword LOCKS is specified. This message forms the headings of the display. (The message appears in one continuous line across the screen, not as shown above.) This message is always followed by multiple DSNT396I messages containing the column data. Refer to message DSNT396I for a description of the data these headings identify.

DSNT396I   *space type part status conn-id corr-id lock*

Explanation: This message contains the display information that results from a DISPLAY DATABASE command when the keyword LOCKS is specified. This message contains the column data describing the status of a table space or index space in the database listed in the preceding DSNT362I message. This message is always preceded by its column headings in message DSNT395I. The message number for this message is not displayed. However, the messages DSNT395I, DSNT396I, and DSNT390I are always produced with a multiple line WTO and are never interrupted.

In this message:

- The *space* is the table space name, index space name, or table OBID.
- The *type* is TS for a table space, IX for an index space, LS for a LOB table space or TB for a table.

  If the object is involved in cloning then the object *type* will be appended with a 'B' (base object) or a 'C' (clone object) along with a data set instance number. A base object no longer involved with cloning will have a 'B2' appended to it's *type* if it's currently represented by an instance number 2 data set. A base object no longer involved with cloning, that is represented by instance number 1 data sets, will not have the 'B' and instance number suffix.

  Refer to the SYSIBM.SYSTABLESPACE catalog table CLONE column to determine if a table space, and therefore it's objects, are involved with cloning. A CLONE='Y' value indicates that cloning is active. The INSTANCE column value indicates the data set instance number for the current base objects.

- The *part* is the partition number. It is blank for a simple table space, simple index space, or segmented table space. For nonpartitioned secondary indexes, it is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value will be the character D (for example, D0001).
- See DSNT389I for a list of the possible *status* display strings. If the *type* is TB, the *status* field is blank.
- The *conn-id* is a connection identifier for the thread. It is one of the following:
  - BATCH if the thread is from a batch region.
  - TSO if the thread is from a TSO terminal.
  - UTILITY if the thread is from a utility.
  - 'IMS identifier' if the thread is from an IMS terminal.
  - 'CICS identifier' if the thread is from a CICS terminal.
  - CONSOLE if the thread is from the console.

  When the displayed lock is a P-lock or a retained lock, *conn-id* is not specified.
- The *corr-id* is a correlation identifier of the thread associated with the space name. It is one of the following:
  - 'job-name' if the thread is from a batch region.
  - 'TSO logon identifier' if the thread is from a TSO terminal.
  - PST#.PSBNAME if the thread is from an IMS terminal.
  - 'entry ID.thread number.tran id' if the thread is from a CICS terminal.
  - CONSOLE if the thread is from the console.

DSNT396I

- For distributed database access threads, this field contains the following:
  - While the thread is being created:
    1. '028.DBAA nn', where nn ranges from 01 to 99. This indicates that the database access agent (DBAA) is in its connection processing, which consists of establishing the DBAA thread, and validating and verifying the user's ID.

       All RACF requests are serialized on a single DB2 RACF service task, which might increase the validation time when multiple DBAAs are being created concurr
  - After the thread is created:
    1. For threads using system-directed access and threads using application-direc access from a DB2 requester, this variable displays the correlation ID of the thread at the requesting location.
    2. For threads using application-directed access from a non-DB2 requester, this variable displays the first twelve characters in the DDM external name (EXTNAM) parameter of the DDM EXCSAT command received as part of the SQL CONNECT

corr-id is not specified when the displayed lock is a P-lock or a retained lock.

- The *lock* describes the type and duration of locks:
  - Transaction locks (also called L-locks) are identified by their correspondin agents' connection and correlation IDs

    Multiple locks held by an agent are listed on multiple lines of the report followed by any locks the agent is currently awaiting.
  - Physical locks (P-locks) and retained locks are identified by their DB2 data sharing member names.

    Multiple P-locks or retained locks held by a DB2 data sharing member are listed on multiple lines of the report followed by any locks the member is currently awaiting.

The *lock* is composed of a 'lock status' and followed by a dash '-', a 'lock state' followed by a 'lock type', and a 'lock duration' delimited by commas.

- The 'lock status' is a value of:

  H   The lock is held by the agent or the DB2 member.

  W   The agent or the member is awaiting the lock.

  R   The lock is retained.

- The 'lock state' describes themode of the lock as follows:
  - For L-locks and drain locks,

    IS   A lock with read intentions.

    IX   A lock with update intentions.

    S   A lock with read-only capability.

    U   A lock with update capability.

    SIX   A table space or table lock with a protocol that does not lock a page while reading, but locks the page with update intention while updating.

    X   An exclusive lock.
  - For L-locks on LOB table spaces,

    IS   A lock with intent to reserve space within the LOB table space

    IX   A lock with intent to insert LOBs in the LOB table space.

    S   A lock which reserves space within the LOB table space.

    SIX   A lock which allows the owner to read and update LOBs.

    X   An exclusive lock.
  - For P-locks,

    IS   This DB2 member has R/O interest. The page set or partition is group buffer pool (GBP) dependent.

    IX   This DB2 member has R/W interest. The page set or partition is GBP-depende

    S   This DB2 member has R/O interest. The page set or partition is not GBP-dep

    SIX   This DB2 member has R/W interest. The page set or partition is GBP-depende

    NSU   This DB2 member has R/W interest. The page set or partition is GBP-depende

    X   This DB2 member has R/W interest. The page set or partition is not GBP-dep

– The 'lock type' describes the type of locking as follows:

    S   A table space L-lock

    T   A table L-lock

    C   A cursor-stability read drain lock

    R   A repeatable-read drain lock

    W   A write drain lock

    P   A partition L-lock

    PP   A page set or partition P-lock

– If the 'lock status' is H, the 'lock duration' describes the duration of the lock:

    A   The lock is freed at deallocation.

    C   The lock is freed at commit.

    H   The lock is freed when all cursors are closed.

    M   The lock is freed by the system.

    P   The lock is freed when the plan is complete.

    I   The lock is a page set or partition P-lock. It is released when the pageset or partition data set is closed.

– If the 'lock status' is W, the 'lock duration' is a 2-digit number indicating the waiter's position in the local DB2 system's lock waiting queue. When this lock is acquired, its duration is the same as all locks held by the user.

– For a P-lock, the lock is owned by the system, not the user, and its duratio is not related to the duration of the locks that are owned by the user.

– Retained locks do not have a lock duration.

All index spaces on tables are implicitly locked with transaction locks at the same level as the table spaces containing those tables. P-locks are acquired by DB2 on index spaces as well as on table spaces.

Related concepts:

▶ DB2 subsystem access control (Managing Security)

Related reference:

▶ -DISPLAY DATABASE (DB2) (DB2 Commands)

▶ SYSIBM.SYSTABLESPACE table (DB2 SQL)

DSNT397I

Explanation: This message indicates the beginning of a multiple line display that is generated as a result of a DISPLAY DATABASE command. The format of this multiple line display has the following three parts:

- The column headers, which are dependent on the keywords issued. The headers are underlined.
- The space information for all requested spaces and partitions. (For partitioned objects, if the partition column is blank or contains L*, the line applies to the entire table space or index object and not to an individual partition.)
- An ending message to state whether the spaces were displayed or not.

Which headers are included in the DISPLAY DATABASE output is dependent on which of the following DISPLAY DATABASE keywords you specify:

- CLAIMERS
- LOCKS
- USE
- LPL

DSNT397I

- no keyword

The following list describes the five different DISPLAY DATABASE command output formats:

- DISPLAY DATABASE CLAIMERS

  This DISPLAY DATABASE command displays the claims that are on certain table spaces, index spaces, and partitions.

  The output for DISPLAY DATABASE CLAIMERS includes the following column headers:

  NAME The table space name or index space name.

  TYPE The type of object. TYPE contains one of the following values:
  - TS for a table space
  - IX for an index space
  - LS for a LOB table space.
  - XS for an XML table space

  If the object is involved in cloning, the TYPE value is appended with the value 'B' (base object) or 'C' (clone object) and a data set instance number. For a base object that is no longer involved with cloning and is currently represented by an instance number 2 data set, the TYPE value is appended with the value 'B2'. For a base object that is no longer involved with cloning and is represented by an instance number 1 data set, the TYPE value is not appended with a 'B' and instance number suffix.

  Look at the CLONE column in the SYSIBM.SYSTABLESPACE catalog table to determine if a table space and its objects are involved in cloning. A value of Y in the CLONE column indicates that cloning is active. The value in the INSTANCE column indicates the data set instance number for the current base objects.

  PART The partition number. PART is blank for a simple table space or simple index space. For nonpartitioned indexes on a partitioned table space, the PART value is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value is the character D (for example, D0001). If this column is blank or contains L* for a partitioned object, the data that is reported in that line of output applies to the entire table space or index object.

  STATUS
  See DSNT389I for a list of the possible display strings for STATUS.

  If the TYPE value is TB, the STATUS field is blank.

  If the status for an index is blank, the status information is displayed at the logical partition level.

  For L-locks on LOB table spaces,

  IS   A lock with intent to reserve space within the LOB table space

  IX   A lock with intent to insert LOBs in the LOB table space.

  S   A lock which reserves space within the LOB table space.

  SIX   A lock which allows the owner to read and update LOBs.

  X   An exclusive lock.

  CONNID
  A connection identifier for the thread. CONNID has one of the following values:
  - BATCH if the thread is from a batch region.
  - TSO if the thread is from a TSO terminal.
  - UTILITY if the thread is from a utility.
  - 'IMS identifier' if the thread is from an IMS terminal.
  - 'CICS identifier' if the thread is from a CICS terminal.
  - CONSOLE if the thread is from the console.
  - For distributed database access threads, this field contains the following information:
    - For threads using system-directed access and for threads using application-directed access from a DB2 Version 3 or later requester, this field displays the connection name of the thread at the requesting location.

- For threads using application-directed access from a non-DB2 requester or from a DB2 Version 2 Release 3 requester, this field displays the constant SERVER.

CORRID

A correlation identifier of the thread that is associated with the space name. CORRID has one of the following values:

– 'job name' if the thread is from a batch region.

– 'TSO logon identifier' if the thread is from a TSO terminal.

– PST#.PSBNAME if the thread is from an IMS terminal.

– 'entry ID.thread number.tran id' if the thread is from a CICS terminal.

– CONSOLE if the thread is from the console.

– '0nn', where 'nn' is a number, if the thread is for a system agent.

– For distributed database access threads, this field contains the following information:

- While the thread is being created, this field contains '028.DBAA nn', where *nn* ranges from 01 to 99. This value indicates that the database access agent (DBAA) is in its connection processing. This processing consists of establishing the DBAA thread and validating and verifying the user's ID.

All RACF requests are serialized on a single DB2 RACF service task, which might increase the validation time when multiple DBAAs are being created concurrently.

- After the thread is created, this field contains one of the following values:

1. For threads using system-directed access and threads using application-direct access from a DB2 requester, this field displays the correlation ID of the thread at the requesting location.

2. For threads using application-directed access from a non-DB2 requester, this field displays the first twelve characters in the DDM external name (EXTNAM) parameter of the DDM EXCSAT command that is received as part of the SQL CONNECT.

CLAIMINFO

The type and duration of currently held claims. The claim is composed of a claim class descriptor and a claim duration descriptor.

The claim class descriptor has one of the following values:

CS Cursor stability claim class

RR Repeatable-read claim class

WR Write claim class

The claim duration descriptor describes the duration of the claims and has one of the following values:

A The claim is held until deallocation.

C The claim is held until the next commit point.

H The claim is held across commit.

- DISPLAY DATABASE LOCKS

This DISPLAY DATABASE command displays the locks that are on certain table spaces, index spaces and partitions.

The output for DISPLAY DATABASE LOCKS includes the following column headers:

NAME The table space name, index space name, or table object ID.

TYPE The type of object. TYPE contains one of the following values:

– TS for a table space

– IX for an index space

– TB for table

– XS for an XML table space

PART See the description in the DISPLAY DATABASE CLAIMERS section.

STATUS

See the description in the DISPLAY DATABASE CLAIMERS section.

DSNT397I

CONNID
> See the description in the DISPLAY DATABASE CLAIMERS section.
>
> The CONNID value is not specified when the display lock is a P-lock or a retained lock (either L-lock or P-lock). For page set P-locks, the DB2 member name is used to identify the lock.

CORRID
> See the description in the DISPLAY DATABASE CLAIMERS section.
>
> The CORRID value is not specified when the display lock is a P-lock or a retained lock (either L-lock or P-lock). For page set P-locks, the DB2 member name is used to identify the lock.

LOCKINFO
> Values that describe the type and duration of the locks that are used by corresponding agents identified by their connection and correlation IDs (for transaction locks, which are also called L-locks) or used by corresponding subsystems identified by their DB2 data sharing member names (for physical locks, which are called P-locks, and retained locks). Multiple locks held by an agent are listed on multiple lines of the report followed by any locks the agent is currently awaiting. Multiple P-locks or retained locks that are held by a DB2 data sharing member are listed on multiple lines of the report followed by any locks the member is currently awaiting.
>
> The LOCKINFO field consists of the following information:
>
> ```
> lock status-lock state,lock type,
> lock duration
> ```
>
> – *lock status* has one of the following values:
>
> | | |
> |---|---|
> | **H** | The lock is held by the agent or the DB2 member. |
> | **W** | The agent or the member is awaiting the lock. |
> | **R** | The lock is retained. |
>
> – *lock state* describes the mode of lock as follows:
>
> - For L-locks and drain locks, *lock state* has one of the following values:
>
> | | |
> |---|---|
> | **IS** | A lock with read intentions. |
> | **IX** | A lock with update intentions. |
> | **S** | A lock with read-only capability. |
> | **U** | A lock with update capability. |
> | **SIX** | A lock with a protocol that does not lock a page while reading, but locks the page with update intention while updating. |
> | **X** | An exclusive lock. |
>
> - For P-locks, *lock state* has one of the following values:
>
> | | |
> |---|---|
> | **IS** | This DB2 member has R/O interest. The page set or partition is group buffer pool (GBP) dependent. |
> | **IX** | This DB2 member has R/W interest. The page set or partition is GBP-dependent. |
> | **S** | This DB2 member has R/O interest. The page set or partition is not GBP-dependent. |
> | **SIX** | This DB2 member has R/W interest. The page set or partition is GBP-dependent. |
> | **NSU** | This DB2 member has R/W interest. The page set or partition is GBP-dependent. |
> | **X** | This DB2 member has R/W interest. The page set or partition is not GBP-dependent. |
>
> – *lock type* describes the type of locking and has one of the following values:
>
> | | |
> |---|---|
> | **S** | A table space L-lock |
> | **T** | A table L-lock |
> | **C** | A cursor-stability read drain lock |
> | **R** | A repeatable-read drain lock |
> | **W** | A write drain lock |

> **P** A partition L-lock
>
> **PP** A page set or partition P-lock
>
> – *lock duration* describes the duration of the lock as follows:
>
> - If *lock status* has a value of H, *lock duration* has one of the following values
>
>   **A** The lock is freed at deallocation.
>
>   **C** The lock is freed at commit.
>
>   **H** The lock is freed when all cursors are closed.
>
>   **M** The lock is freed by the system.
>
>   **P** The lock is freed when the plan is complete.
>
>   **I** The lock is a page set or partition P-lock. It is released when the page set or partition data set is closed.
>
> - If the *lock status* has a value of W, *lock duration* is a 2-digit number that indicates the waiter's position in the local DB2 system's lock waiting queue.
>
>   After the agent acquires the lock, the *lock status* value becomes H, and the *lock duration* value becomes one of the lock duration values that are listed for a lock status of H.
>
> - For a P-lock, the lock is owned by the system, not the user, and its duration is not related to the duration of the locks that are owned by the user.
>
> - Retained locks do not have a lock duration.
>
> There are no explicit index space locks. However, all index spaces on tables are implicitly locked at the same level as the table spaces containing those tables. P-locks are acquired by DB2 on index spaces as well as on table spaces.

- DISPLAY DATABASE USE

  This DISPLAY DATABASE command displays the user IDs that are accessing certain table spaces, index spaces, and partitions.

  The output for DISPLAY DATABASE USE includes the following column headers:

  NAME See the description in the DISPLAY DATABASE CLAIMERS section.

  TYPE See the description in the DISPLAY DATABASE CLAIMERS section.

  PART See the description in the DISPLAY DATABASE CLAIMERS section.

  STATUS
  > See the description in the DISPLAY DATABASE CLAIMERS section.

  CONNID
  > See the description in the DISPLAY DATABASE CLAIMERS section.

  CORRID
  > See the description in the DISPLAY DATABASE CLAIMERS section.

  USERID
  > A field that contains one of the following values:
  >
  > – 'TSO logon identifier' if the thread is from a TSO terminal.
  >
  > – 'sign-on id' or the LTERM if the thread is from an IMS message-drive region.
  >
  > – AXBUSER if the thread is from an IMS message-driven region and RACF is used.
  >
  > – 'PSB name' if the thread is from an IMS nonmessage-driven region and RACF is not used.
  >
  > – 'userid', 'term id', 'tran id' or other, as specifie in the resource control table (RCT), if the thread is from CICS.

- DISPLAY DATABASE LPL

  This DISPLAY DATABASE command displays information about LPL pages.

  The output contains the column data that is described in the status of a table space or index space in the database listed in the preceding DSNT362I message. This message is always preceded by its column headings message DSNT805I. The message number for this message is not displayed.

The output for DISPLAY DATABASE USE includes the following column headers:

NAME See the description in the DISPLAY DATABASE CLAIMERS section.

TYPE See the description in the DISPLAY DATABASE CLAIMERS section.

PART See the description in the DISPLAY DATABASE CLAIMERS section.

STATUS
> See the description in the DISPLAY DATABASE CLAIMERS section.

LPL PAGES
> Lists the pages in the specified *space* that are unavailable because of logical or physical damage.

- DISPLAY DATABASE (with no LOCKS, USE, LPL or CLAIMERS keyword).

This DISPLAY DATABASE command displays the error page range information for certain table spaces, index spaces, and partitions.

The output for DISPLAY DATABASE includes the following column headers:

NAME See the description in the DISPLAY DATABASE CLAIMERS section.

TYPE See the description in the DISPLAY DATABASE CLAIMERS section.

PART See the description in the DISPLAY DATABASE CLAIMERS section.

STATUS
> See the description in the DISPLAY DATABASE CLAIMERS section.

PHYERRLO
> A hexadecimal number that identifies the lowest page number that contains a physical I/O error in a data set that supports the specified table space or index space. This column is blank if no physical errors exist.

PHYERRHI
> A hexadecimal number that identifies the highest page number that contains a physical I/O error in a data set that supports the specified table space or index space. This column is blank if no physical errors exist.

CATALOG
> When a data set that supports the specified table space or index space has physical I/O errors, CATALOG is the catalog that owns the data set with the I/O errors. This column is blank if no physical errors exist.

PIECE When a data set that supports the specified table space or index space has physical I/O errors, PIECE is an integer that identifies the piece within the table space or index space that has the error. This column is blank if no physical errors exist.

> The piece number can be used to construct the name of the data set that supports the space that contains the physical error as follows:

```
cat-name.DSNDBC.database-name.
space-name.I0001.Axxxx
```

> The value *xxxx* is the piece number plus one. *database-name* is the name that is displayed in the preceding DSNT362I message. If the table space or index space is partitioned, the partition number should equal the piece number plus one.

The following informational messages that are not related to a specific DISPLAY DATABASE keyword can also be displayed in the output:

- -luw-id ACCESSING DATA FOR location

This message is issued for each thread reported that is participating in distributed data access (database access thread). The thread that corresponds to this message is the thread that is named by the correlation id and connection id of the data line that precedes this message. This additional line reports the distributed identification as follows:

luw-id=token
> The logical-unit-of-work identifier (LUW-ID) is a unique identifier that is assigned to the thread followed by an '=' sign and a token. The token can be used in place of luw-id in any DB2 command that accepts luw-id as input.

location

> The LOCATION name assigned to the requesting site.

- -MEMBER NAME *member-name*

This message is issued with messages DSNT389I, DSNT394I or DSNT396I for each thread in a data sharing environment. This additional line reports the member name of the DB2 subsystem accessing this object.

If the characters '(CO)' follow the member name, then this member is the castout owner for the page set or partition. The castout owner for a page set or partition is the DB2 member responsible for doing all of the castout I/O for that page set or partition.

The message number and subsystem recognition character are not included in this message.

- INVALID space-name

The space-name specified in the SPACENAM keyword on the -DISPLAY DATABASE command does not exist. The name of the table space or index space is 'space-name'. The request is rejected.

- NO SPACES FOUND

This message is produced when a DISPLAY DB(name) SPACENAM is entered and no spaces were found to satisfy the request. SPACENAM(...) might have been entered as a range, a subset, or a list of spaces.

- spacetype name NOT type

This message is produced when no table space or index space listed in the DISPLAY DB(...) SPACENAM(...) satisfies your request.

spacetype

> TABLESPACE if the space is a table space or INDEXSPACE if the space is an index space.

name   The name of the space.

type   A character string that indicates the type of request specified. The string is a list of the following, delimited by 'OR':

- ACTIVE if the keyword ACTIVE is specified
- RESTRICTED if the keyword RESTRICT is specified.

- NO INFORMATION AVAILABLE

This message is produced when there is no LOCK or USE information available. When DISPLAY DB(...) is entered, and DB(...) is a list of databases or a list of names, this message indicates that no listed database has LOCK or USE information available.

If DISPLAY DB(...) or DISPLAY DB(...) SPACENAM(*) is entered, and DB(...) is a range or subset of databases, or all databases ('*'), this message indicates that no databases, for which the invoker has authority to display, have LOCK or USE information available. This message is produced only when the keywords USE and LOCKS are specified.

The ending message looks like the following line:

```
* DISPLAY OF DATABASE name action *
```

The variables in this ending message have the following values:

name   The name of the database

action   One of the following actions:

CONTINUES

> The remaining table space and index space names within this database are continued to a new display.

TERMINATED

> The remaining table space and index space names within this database are terminated.

ENDS   The display for this database is complete.

Related reference:

↪ SYSIBM.SYSTABLESPACE table (DB2 SQL)

Related information:

"DSNT389I" on page 519

DSNT398I

Explanation: This message appears within a multiple line WTO message set and precedes DSNT391I, DSNT393I, and DSNT395I. It is used to underscore the column headings.

This message is issued by CSECT: DSNTDMUL

DSNT399I   INVALID space-name

Explanation: The space-name specified in the SPACENAM keyword on the -DISPLAY DATABASE command does not exist. The name of the table space or index space is 'space-name'. The request is rejected.

DSNT400I   SQLCODE = 0, SUCCESSFUL EXECUTION

Explanation: The SQLCODE was 0, indicating successful execution of the function. SQLWARN indicators might note some warnings. .

System action: Processing continues.

Operator response: No action is required.

Related concepts:

↪ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

↪ Description of SQLCA fields (DB2 SQL)

DSNT402I   ONLINE BOOK DATA SET NAMES MUST END IN '.BOOK'

Explanation: All book names entered on this panel must end in ".BOOK".

This message is issued by panel: DSNTIPA0

System action: The panel is redisplayed with the error message.

User response: Replace the book name or names with a value that ends in ".BOOK".

DSNT404I   SQLCODE = *sql-code*, *explanation*

Explanation: This message contains an SQL return code and a brief explanation, with text inserted from the SQLERRM field. The SQL return code is positive, indicating a warning condition.

System programmer response: Determine the cause for the SQL warning code, and correct the situation if needed. Rerun the application program or SQL statement.

Related concepts:

↪ Successful execution SQL codes (DB2 Codes)

Related reference:

↪ Description of SQLCA fields (DB2 SQL)

DSNT405E   csect-name DISPATCH PRIORITIES NOT IN SYNC: ccccccc1: hhh1 COMPARED TO ccccccc2: hhh2

Explanation: The dispatch priorities of task 'ccccccc1' (dispatch priority 'hhh1') and task 'ccccccc2' (dispatch priority 'hhh2') are found to be incorrect in relation to each other. This can cause severe performance problems and possible failure of either task 'ccccccc1', task 'ccccccc2', or other tasks.

System action: DB2 continues with possible problems described above.

Operator response: Notify the system programmer.

System programmer response: Modify the dispatch priority algorithms for the two tasks to meet documented requirements. For instance, IRLM must have a higher dispatch priority than DB2.

DSNT406I   THE ART/ORT ESCAPE CHARACTER CANNOT BE % OR _

Explanation: You specified the percent sign (%) or the underscore character (_) as the application registration table/object registration table (ART/ORT) escape character. The ART/ORT escape character can be any other character.

System action: The panel is redisplayed and the cursor is placed on the field that produced the error.

User response: Change the ART/ORT escape character value.

DSNT407I   THIS FIELD HAS AN INVALID FORMAT

Explanation: The field must be NOLIMIT, 1440, 0, or minutes and seconds separated by a comma.

This message is issued by the following panel: DSNTIPA

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Correct the input.

DSNT408I   SQLCODE = -*sql-code*, *explanation*

Explanation: This message contains an SQL return code and a brief explanation, with text inserted from the SQLERRM field. The SQL return code is negative, indicating that an error has occurred.

System programmer response: Determine the cause for the SQL error by using information about that specific SQLCODE. Correct the error. Rerun the application program or SQL statement.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of the results produced by the SQL statements.
- Source listing of the failing application program.

Related concepts:

➦ SQL error codes (DB2 Codes)

Related reference:

➦ Description of SQLCA fields (DB2 SQL)

DSNT409I   BUFFERPOOL SIZES SHOWN MAY BE OBSOLETE

Explanation: The buffer pool sizes shown here were set at installation or migration time and might not be the most current. Updates might have occurred using the ALTER BUFFERPOOL command.

System action: Processing continues.

DSNT410I   THE SUBSYSTEM COMMAND PREFIX ENTERED IS INCORRECT

Explanation: An incorrect command prefix was entered. The first character of the command prefix must be one of the following characters:

Character	Description
@	at sign
$	dollar sign

Character	Description
#	number sign
.	period
/	forward slash
'	apostrophe
?	question mark
"	quotation mark
%	percent sign
!	exclamation point
<	less than symbol
)	closing parenthesis
*	asterisk
+	plus sign
-	minus sign
=	equal sign
¢	cent sign
:	colon
_	underscore
;	semicolon
|	vertical bar

For the other characters in the command prefix, you can use any character from the preceding set, uppercase letters (A-Z), and digits (0-9).

This message is issued by the following panel: DSNTIPM

System action: The panel is redisplayed and the cursor is placed on the field in error

User response: Enter a valid command prefix

DSNT411I   SQLCAID = xxxx, SQLCA IDENTIFIER

Explanation: The SQLCA ID field (SQLCAID) is not the expected value, C'SQLCA'.

System programmer response: The SQLCA area is invalid. This type of error is usually the result of one of the following:

- Calling the DSNTIAR formatting routine before an SQL statement initializes the area
- Overlaying the SQLCA area
- Passing the wrong address to the DSNTIAR routine.

Correct the error, and rerun the application program.

DSNT412I   SQLCABC = xxxx, SQLCA BYTE COUNT

Explanation: The size of the SQLCA is not the expected value (136 bytes).

System programmer response: The SQLCA area is invalid. Correct the error. It is probably an overlay or an incorrect SQLCA address. Rerun the application program.

DSNT413I   **SQLCODE = xxxx, INVALID SQLCODE**

Explanation: The SQL return code is not a valid value.

System programmer response: Either the SQLCA area is invalid or the maintenance was not coordinated correctly. As a result, the SQL return code was added to the DB2 subsystem but not to the formatting routine.

An invalid SQLCA area is usually the result of one of the following:

* Calling the DSNTIAR formatting routine before an SQL statement initializes the area
* Overlaying the SQLCA area
* Passing the wrong address to the DSNTIAR routine.

If the maintenance was not coordinated correctly, the DSNTIAR and DSNTIAM CSECTs were probably not replaced in the code that is now running. Link-edit the load module, and replace DSNTIAR and DSNTIAM. Correct the error, and rerun the application program.

DSNT414I   **SQLERRM = xxxx, SQLCA ERROR MESSAGE INSERTION TEXT**

Explanation: This message results from an SQL return code that is not recognized by DSNTIAR. SQLERRM contains error message text.

System programmer response: Either the SQLCA area is invalid or the maintenance was not coordinated correctly. As a result, the SQL return code was added to the DB2 subsystem but not to the formatting routine.

An invalid SQLCA area is usually the result of one of the following:

* Calling the DSNTIAR formatting routine before an SQL statement initializes the area
* Overlaying the SQLCA area
* Passing the wrong address to the DSNTIAR routine.

If the maintenance was not coordinated correctly, the DSNTIAR and DSNTIAM CSECTs were probably not replaced in the code that is now running. Link-edit the load module, and replace DSNTIAR and DSNTIAM. Correct the error, and rerun the application program.

DSNT415I   **SQLERRP = xxxx, SQL PROCEDURE DETECTING ERROR**

Explanation: The name of the procedure detecting the error is provided in the SQLERRP.

System programmer response: (The SQL procedure detecting the error may be of help in diagnosing an error or warning.) Correct the error if there is one noted in other messages. Rerun the program or SQL statement.

DSNT416I   **SQLERRD =** *field-1*, *field-2*, *field-3*, *field-4*, *field-5*, *field-6* **SQL DIAGNOSTIC INFORMATION**

Explanation: One or more of the six diagnostic fields do not have the expected value. The diagnostic fields should contain one of the following expected sets of values:

* 00, 00, 00, 00, 00, 00
* 00, 00, 00, -1, 00, 00

field-1   DB2 unique internal relational data system code.

field-2   DB2 internal data manager error code.

field-3   One of the following conditions:

* Number of rows inserted, updated, or deleted following an INSERT, UPDATE, MERGE or DELETE statement.

    Attention: A **-1** in this field indicates that a mass delete of a table in a segmented table space was performed, wherein DB2 does not access nor count individual rows.

* For SQLCODE -911 or -913, a reason code that indicates whether a deadlock or timeout occurred.

    Tip: The most common reason codes are:

    – 00C90088 - deadlock
    – 00C9009E - timeout

DSNT417I

field-4   Reserved.

field-5   Position or column of a syntax error for PREPARE or EXECUTE IMMEDIATE.

field-6   DB2 internal buffer manager error code.

System action: Processing continues.

System programmer response: The SQL diagnostic information might help in diagnosing an error or in indicating the results of successful execution. If this error or warning should not occur, correct the error. Rerun the program or SQL statement.

Related reference:

➥ GET DIAGNOSTICS (DB2 SQL)

➥ Description of SQLCA fields (DB2 SQL)

Related information:

➥ -911 (DB2 Codes)

➥ -913 (DB2 Codes)

➥ 00C90088 (DB2 Codes)

➥ 00C9009E (DB2 Codes)

DSNT417I   **SQLWARNn-n SQL WARNINGS**

Explanation: At least one of the 11 warning values is not the expected value (a blank). A nonblank value in one of the 11 warning fields has the following meaning:

SQLWARN0
>   Any other warning code is set.

SQLWARN1
>   String truncation.

SQLWARN2
>   Null value(s) eliminated.

SQLWARN3
>   Number of host variables provided did not equal degree of the table.

SQLWARN4
>   No WHERE clause on UPDATE or DELETE.

SQLWARN5
>   Not a DB2 SQL statement.

SQLWARN6
>   Date or timestamp value adjusted.

SQLWARN7
>   Conversion of decimal number from precision 31 to precision 15 caused truncation of fractional digits.

SQLWARN8
>   Character that does not exist on the target code page was replaced with a substitution character.

SQLWARN9
>   Arithmetic exceptions have been ignored during processing of COUNT DISTINCT.

SQLWARNA
>   At least one character field of the SQLCA is invalid due to a string conversion error.

User response: The SQL warning information may be of help in diagnosing an error or in indicating the results of successful execution. If this warning should occur, correct the error and rerun the program or SQL statement.

DSNT418I   SQLSTATE = *sqlstate*, SQLSTATE RETURN CODE

Explanation: The SQLSTATE is a return code that indicates the outcome of the most recently executed SQL statement. The execution of every SQL statement sets SQLSTATE to a five-digit code in the range of 00000 to 65535. This has no effect on the existing use of any other field in the SQLCA.

User response: Examine the SQLSTATE value. If necessary, correct the SQL statement and reissue it.

Related concepts:

↪ SQL communication area (SQLCA) (DB2 SQL)

Related tasks:

↪ Checking the execution of SQL statements by using SQLCODE and SQLSTATE (DB2 Application programming and SQL)

Related reference:

↪ SQLSTATE values and common error codes (DB2 Codes)

↪ SQLSTATE (DB2 SQL)

DSNT419I   WARNING - MAX REMOTE ACTIVE PREVIOUSLY SPECIFIED AS 0.

Explanation: You specified a value of 0 for the number of active database access threads on a previous panel. This value might be inconsistent with a DDF STARTUP OPTION of AUTO or COMMAND.

This message is issued by the following panel: DSNTIPR

System action: The panel is redisplayed and the cursor is placed on the DDF STARTUP OPTION field.

User response: No response is necessary if your intent is to have DDF started with no active database access threads allocated and DDF is used as a requester only. Press ENTER to continue. Otherwise, either change the DDF STARTUP OPTION to NO or MAX REMOTE ACTIVE to the number of threads you want.

Severity: 4 (warning)

DSNT422I   THE ART TABLE NAME MUST BE DIFFERENT FROM THE CRT TABLE NAME

Explanation: You entered the same name for the Application Registration Table and the Object Registration Table on the Data Definition Control Support panel.

This message is issued by the following panel: DSNTIPZ

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change either the APPL REGISTRATION TABLE field or the OBJT REGISTRATION TABLE field to make these two fields unique with respect to each other.

DSNT424I   OUTPUT DATA SETS MUST BE DIFFERENT FROM THE INPUT DATA SETS

Explanation: You specified an output data set name that is the same as one of the input data set names built from the PREFIX and SUFFIX fields on the main panel DSNTIPA1.

This message is issued by the following panel: DSNTIPT

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Either change the data set name under the cursor or change the PREFIX or SUFFIX fields on panel DSNTIPA1.

DSNT425I   ALL THREE OUTPUT DATA SETS MUST BE UNIQUE

Explanation: You specified the same name for two or more of the output data sets.

This message is issued by the following panel: DSNTIPV

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change the data set name under the cursor or the data set field(s) that are the same as the one under the cursor.

DSNT426I   ALL THREE DATA SETS ARE REQUIRED FOR OUTPUT

Explanation: You specified one or more output data sets and left one or more output data set fields blank.

This message is issued by the following panel: DSNTIPT

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: You must specify all or none of the output data set fields. Leave blank all output data set fields if you do not want the installation CLIST to edit any CLISTs or JCL.

DSNT427I   THE DATA SET OR MEMBER SPECIFIED COULD NOT BE LOCATED

Explanation: The input parameter member for the INPUT MEMBER NAME field or the DATA SET NAME(MEMBER) field for migration could not be found.

This message is issued by the following panel: DSNTIPA1

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Verify that you entered an existing data set and member in that data set. Also verify that the data set is cataloged.

DSNT428I   WHEN USING TIMESTAMPS, THE PREFIX MUST BE <= *preflen* CHARACTERS LONG

Explanation: YOU HAVE ENTERED AN ARCHIVE LOG PREFIX LONGER THAN ALLOWED. The archive log prefix names must not exceed 19 characters when the TIMESTAMP ARCHIVES value is YES and must not exceed 17 characters when the TIMESTAMP ARCHIVES value is EXT.

This message is issued by the following panel: DSNTIPH

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Reduce the length of the prefix or change the TIMESTAMP ARCHIVES value to NO.

DSNT429I   THE SUM OF MAX USERS AND MAX REMOTE ACTIVE MUST BE <= TO 2000

Explanation: The total number of threads determined by MAX USERS and MAX REMOTE fields is too large.

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Reduce the MAX USERS field, or the MAX REMOTE field, or both.

DSNT430I   THIS FIELD MAY ONLY BE FILLED IN FOR MIGRATE MODE

Explanation: You can only specify the FROM RELEASE or DATA SET NAME(MEMBER) fields for INSTALL TYPE of MIGRATE.

This message is issued by the following panel: DSNTIPA1

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change the INSTALL TYPE to MIGRATE or blank out the field under the cursor.

DSNT431I   DDF, MIXED DATA, OR DEF ENCODING SCHEME CONFLICTS WITH CODED CHAR SET

Explanation: An invalid EBCDIC or ASCII coded character set (CCSID) value is not valid for the specified fDDF, MIXED DATA, or DEF ENCODING SCHEME. The following restrictions apply to these values:

- With a MIXED DATA value of 'YES', you must choose a valid mixed CCSID value for the EBCDIC CCSID. The ASCII CCSID must be a valid mixed value or zero.
- With a MIXED DATA value of 'NO', you must choose a valid SBCS CCSID value or zero for EBCDIC CCSID or ASCII CCSID.
- With a DDF value of 'AUTO' or 'COMMAND', you must choose a non-zero value for the EBCDIC CCSID.

- With a DEF ENCODING SCHEME value of 'ASCII', you must choose non-zero values for EBCDIC CCSID and ASCII CCSID.

This message is issued by the following panels: DSNTIPF, DSNTIPR

System action: The panel is re-displayed and the cursor is placed on the field in error.

User response: Correct either the DDF, MIXED DATA, or DEF ENCODING SCHEME value or the EBCDIC CCSID or ASCII CCSID value to correct the incompatibility.

Related reference:

➡️ EBCDIC and ASCII support (DB2 Installation and Migration)

➡️ Application programming defaults panel 1: DSNTIPF (DB2 Installation and Migration)

➡️ Distributed data facility panel 1: DSNTIPR (DB2 Installation and Migration)

DSNT433I   MAX REMOTE ACTIVE MUST BE <= MAX REMOTE CONNECTED.

Explanation: The number of active database access threads cannot be larger than the number of concurrent database access threads.

System action: The panel is displayed again, and the cursor is placed on the field in error.

User response: Take either or both of the following actions:
- Reduce the value of the MAXDBAT subsystem parameter
- Increase the value CONDBAT subsystem parameter

Severity: 8 (error)

Related reference:

➡️ MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

➡️ MAX REMOTE CONNECTED field (CONDBAT subsystem parameter) (DB2 Installation and Migration)

DSNT434I   WARNING, DATA SETS MARKED WITH ASTERISKS EXIST AND WILL BE OVERWRITTEN

Explanation: The data sets marked with asterisks already exist. If you continue, the installation process will delete the existing data set and allocate a new one for tailored job output.

This message is issued by the following panel: DSNTIPT

User response: You should:
1. Press ENTER to continue; the data set(s) will be deleted and reallocated OR
2. Press RETURN to leave the installation process; the data set(s) will not be deleted OR
3. Change the data set name(s); the new name will be checked for existence.

Severity: 4

DSNT435I   DSNT435I OVERRIDE VALUES ARE IN EFFECT

Explanation: Calculated values for EDMPOOL, SORTPOOL, or RIDPOOL on panel DSNTIPC were previously overridden. Installation uses any override values displayed.

System action: The display shows the override values instead of the calculated values.

User response: No response is necessary if you want to use the override values. If you want to use the calculated values, erase the values in the override column.

Severity: 4

DSNT436I   REGION SIZE MAY BE TOO LARGE

Explanation: The numbers you specified produced a region size estimate of more than 5000KB. This region size estimate does not include the storage required for SWA because it is assumed that SWA has been placed above the line. If a larger private area is permitted, you should increase the region size on the DB2 start procedures.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST continues.

User response: Check the values entered for numbers of concurrent users and databases, buffers, and plan sizes. Check the available private region size. If this region size is not permitted, these numbers need to be reduced. If the size is permitted, this warning can be ignored.

Problem determination: The region size calculation is shown in the DSNT485I through DSNT487I messages. If you suspect an error in the process, set CONTROL(CONLIST) and invoke the CLIST. This produces tracing for error documentation.

For more information about activating the CLIST trace facility, see the DB2 diagnosis information.

Severity: 4 (warning)

Related concepts:

▷ Virtual storage requirements for storage pools and working storage (DB2 Installation and Migration)

Related tasks:

▷ Invoking the CLIST (DB2 Installation and Migration)

Related information:

▷ DB2 Diagnosis Guide and Reference

DSNT437I   REGION SIZE IS TOO LARGE

Explanation: The numbers you specified produced a region size estimate of more than 8000KB. This estimate does not include the storage required for SWA, because it is assumed that you placed SWA above the 16MB line.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST continues, but does not produce edited jobs.

User response: Check the numbers entered for numbers of concurrent users and databases, buffers, and plan sizes. One or more of these numbers needs to be reduced.

Problem determination: If you suspect an error in the process, set CONTROL(CONLIST) and invoke the CLIST. This produces tracing for error documentation.

For more information about activating the CLIST trace facility, see the DB2 diagnosis information.

Severity: 8 (error)

Related concepts:

▷ Virtual storage requirements for storage pools and working storage (DB2 Installation and Migration)

Related tasks:

▷ Invoking the CLIST (DB2 Installation and Migration)

Related information:

▷ DB2 Diagnosis Guide and Reference

DSNT438I   WARNING, IRLM LOCK MAXIMUM SPACE = *specified-max* **MB, AVAILABLE SPACE =** *available-space* **MB**

Explanation: The maximum number of locks allowed for each user could exceed what is available to the IRLM.

specified-max
        The estimated maximum amount of space specified as IRLM LOCK MAXIMUM SPACE on panel DSNTIPC.

available-space
> The amount of space that is available for IRLM locks. The available space is specified in the MAX STORAGE FOR LOCKS field of the DSNTIPJ installation panel, and is never less than 2 GB.

All threads are not likely to have the maximum number of locks. However, if the *specified-max* value exceeds the amount of available space by a significant factor, the possibility of abends in the IRLM increases.

System action: The CLIST continues.

User response: Choose one of the following responses:
- Ensure that you can run with the overcommitment of lock address space.
- Reduce the values of the MAX USERS, MAX REMOTE ACTIVE, or LOCKS PER USER fields.
- Increase the value of the MAX STORAGE FOR LOCKS field.

Severity: 4 (warning)

Related tasks:

Estimating the storage needed for locks (DB2 Performance)

Related reference:

CLIST calculations panel 1: DSNTIPC (DB2 Installation and Migration)

IRLM panel 2: DSNTIPJ (DB2 Installation and Migration)

MAX USERS field (CTHREAD subsystem parameter) (DB2 Installation and Migration)

MAX REMOTE ACTIVE field (MAXDBAT subsystem parameter) (DB2 Installation and Migration)

LOCKS PER USER field (NUMLKUS subsystem parameter) (DB2 Installation and Migration)

MAX STORAGE FOR LOCKS field (DB2 Installation and Migration)

DSNT439I   IRLM NO LOCK ESCALATION OR NO USER MAXIMUM

Explanation: You specified the number 0 for:
- The number of table or table space locks that will cause lock escalation, or
- The number of locks that will terminate a DB2 user.

A lock escalation value of 0 turns off lock escalation and means that DB2 cannot use this means of conserving lock space. A user maximum of 0 means that the calculation noted in the DSNT438 message cannot be performed to see if the IRLM has sufficient address space for all users, since there is no maximum per user.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue. This is only a warning.

User response: Turn off lock escalation and/or user lock limiting.

Severity: 4 (warning)

DSNT440I   TOTAL BP SIZE TOO LARGE

Explanation: The total size of a buffer pool is too large. The summation of the *ssnm*DBM1 address space storage requirements for all buffer pools cannot exceed 1TB. When calculating the storage requirements for the buffer pools, DB2 uses the page size (4K, 8K, 16K, or 32K) for each buffer.

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Reduce one or more of the buffer pool sizes.

DSNT442E   ERROR, VIRTUAL STORAGE OF main EXCEEDS THE SYSTEM LIMIT OF 2 GIGABYTES

Explanation: The total of all virtual storage calculations exceeds the maximum amount possible for an address space.

This message is issued by the following CLIST: DSNTINST

System action: The installation CLIST returns to the initial panel without editing the installation jobs.

User response: Respecify installation values that determine storage sizes so that the 2 GB limit is not exceeded.

Severity: 8 (error)

DSNT443I   VALUES MARKED WITH AN ASTERISK HAVE BEEN UPDATED

Explanation: An asterisk is placed before the description of fields which were generated by the DSNTINST CLIST. This message only appears when you update field(s) from a prior panel used to generate the fields on the current panel.

This message is issued by the following panels: DSNTIPA, DSNITPL

System action: The panel is displayed.

User response: Verify that the current value is what you want. If it is not, enter the correct value.

DSNT444I   SCROLLING BACKWARD MAY CHANGE FIELDS MARKED WITH ASTERISKS

Explanation: The marked fields on the current panel are generated from fields on other panels. If you scroll backward, these fields may be updated to reflect new values from other panels. For example, the ARCHIVE PREFIX field is generated from the CATALOG ALIAS field on DSNTIPA2. Message DSNT443I will be displayed to inform you when fields are changed by the DSNTINST CLIST.

This message is issued by the following panels: DSNTIPA, DSNITPL

System action: The panel is displayed.

User response: No action required.

DSNT445I   REMOTE LOCATION NAME MUST NOT BE THE SAME AS LOCAL LOCATION

Explanation: The name entered for the remote sample location was the same as the local location name.

This message is issued by the following installation panel: DSNTIPY

System action: The panel is displayed.

User response: Enter a remote location name or leave the field blank to indicate that the organization sample will only be executed locally.

DSNT446I   SAMPLE LOCATION MUST BE BLANK IF LOCAL LOCATION IS BLANK

Explanation: In order to edit the remote sample jobs, both the DB2 LOCATION NAME (local location) on panel DSNTIPR and the REMOTE LOCATION must have valid names. If the DB2 LOCATION NAME is unknown, leave both fields blank and the remote sample jobs will not be edited.

This message is issued by the following panel: DSNTIPY

System action: The panel is displayed.

User response: Either scroll back to the DISTRIBUTED DATA FACILITY panel and enter a value for the DB2 LOCATION NAME, or enter a blank in the REMOTE LOCATION field on the job editing panel.

DSNT448I   STG AVAILABLE ABOVE 2GB = size T

Explanation: This is the estimated amount of real storage recommended to support the virtual storage sizes chosen.

The amount of real storage needed varies greatly, and you might find that you need several times more storage than the estimate that is shown.

This message is issued by the following CLIST: DSNTINST

System action: The CLIST continues.

User response: Compare this recommended value with the actual available real storage.

Severity: 0 (successful execution)

Related concepts:

▶ Virtual storage requirements for storage pools and working storage (DB2 Installation and Migration)

Related tasks:

▶ Invoking the CLIST (DB2 Installation and Migration)

Related information:

▶ DB2 Diagnosis Guide and Reference

DSNT449I   NUMBER OF LOGS REDUCED FROM old-num TO MAXIMUM OF 31

Explanation: The number of active log data sets used in the DB2 release you are migrating from is greater than the new maximum number of data sets allowed for the current release of DB2. Installation job DSNTIJUZ will update your bootstrap data sets to contain only the last 31 data sets.

This message is issued by the following panel: DSNITPL

System action: The value is reduced to the maximum of 31 and the panel is displayed.

User response: Update any other fields on the panel and continue.

DSNT450I   type ENTRY MUST BE EXACTLY TWO CHARACTERS

Explanation: These entries are used to complete SYS1.PARMLIB member names and must be exactly two characters. The type is:

APF AUTH
>   For the APF-authorized library list IEAAPFxx

LINK LIST
>   For the link list library concatenation LNKLSTxx

MVS SUBSYS
>   For the MVS subsystem list IEFSSNxx

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

User response: Check your entries for these fields on the MVS panel and make them exactly two characters. See job DSNTIJMV that was produced and correct the IEBUPDTE CHANGE statement.

Severity: 8 (error)

DSNT451I   WARNING-THE LOCAL DATE LENGTH OR LOCAL TIME LENGTH HAS BEEN INCREASED

Explanation: The LOCAL DATE LENGTH or LOCAL TIME LENGTH has been increased and a REBIND command must be issued for any dependent plans.

This message is issued by the following installation CLIST: DSNTINST

System action: Processing continues.

User response: Rebind all application plans that depend upon the length of the local format that was changed.

Severity: 4 (warning)

DSNT454I   ERROR CODE = errcode, CNT = count

Explanation: An error was encountered during CLIST processing for reading the parameter member. Most of these errors are caused by having a mismatch between the level of the CLIST and that of the parameter member being read.

The 'errcode' is an error code defined for the CLIST. The error code values and associated meanings are found in *TSO/E CLISTS*.

The 'count' value is the number of parameter values already read.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will terminate.

User response: Be sure the parameter member and the CLIST are at the same level.

Problem determination: If you suspect an error in the process, set CONTROL(CONLIST) and invoke the CLIST in batch, so that the tracing can be sent with the error documentation. You will also need the listing of the CLIST and the parameter member being read.

DSNT456I   ERROR IN VARIABLE varname, VALUE IS value

Explanation: The variable named 'varname' does not have a valid value. Its current value is 'value'. Subsequent messages will describe the reason the value is invalid and the action taken by the CLIST.

If the error is in the CONTROL value, the value was specified when the CLIST was invoked. The value is not valid, and no additional tracing will be done. The permitted values are NONE, LIST, CONLIST, and SYMLIST. Each value produces more CLIST tracing information than those which precede it. If the error is in other values, the value was specified when the CLIST was invoked on the ISPF panels.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce jobs or parameter data.

User response: Check the subsequent messages. Exit, then invoke the CLIST with a valid value for the variable.

Problem determination: If you suspect an error in the process, set CONTROL(CONLIST) and invoke the CLIST in batch, so that the tracing can be sent with the error documentation. The CLIST itself and the parameter member read in should be sent as well.

Severity: 8 (error)

DSNT457I   The SUBSYSTEM NAME MUST BE DIFFERENT THAN THE PROC NAME

Explanation: Internal MVS restrictions require that the IRLM subsystem name be different than the procedure name.

This message is issued by the following panel: DSNTIPI

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change either the SUBSYSTEM NAME or the PROC NAME so these fields are unique with respect to each other.

DSNT458I   VALUE MUST BE NUMERIC

Explanation: The variable named in message DSNT456I must be an integer. Only digits are allowed, with no decimal point or fractions. If the value is a number suffixed by K or M, it is interpreted as the indicated number multiplied by 1024 or 1048676. If any other non-numeric character is present in the value, or if the K or M is not the final character, this message is produced.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs or parameter data.

User response: This CLIST will continue, but will not produce edited jobs. The value of the variable has been set to the value shown in message DSNT465I. You will need to reenter the output parameter and output data set prefix. Examine the value and reset it if needed.

Severity: 8 (error)

DSNT459I   **VALUE IS TOO LOW, MINIMUM IS min**

Explanation: The variable named in message DSNT456I must be larger than its current value. The minimum value allowed is 'min'.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs. The CLIST resets the incorrect value to 'min'.

User response: See message DSNT465I for the CLIST action.

Severity: 8 (error)

DSNT460I   **VALUE IS TOO HIGH, MAXIMUM IS max**

Explanation: The variable named in message DSNT456I must be smaller than its current value. The maximum value allowed is 'max'.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs. The CLIST resets the incorrect value to 'max'.

User response: See message DSNT465I for the CLIST action.

Severity: 8 (error)

DSNT462I   **VALUE IS TOO LONG, MAXIMUM IS length CHARACTERS**

Explanation: The variable named in message DSNT456I must be shorter than its current value. The maximum length allowed is 'length'.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs. The CLIST resets the value to the default value.

User response: See message DSNT465I for the CLIST action.

Severity: 8 (error)

DSNT463I   **DATA SHARING FIELD MUST BE BLANK IN UPDATE MODE.**

Explanation: This field is not relevant in Update mode and must be blank This message is issued by panel: DSNTIPA1

System action: The panel is redisplayed and the cursor is placed on t field in error.

User response: Replace the value with a blank.

DSNT464I   **VALUES PERMITTED ARE values**

Explanation: The variable named in message DSNT456I must be one of the 'values' listed.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs.

User response: See message DSNT465I for the CLIST action.

Severity: 8 (error)

DSNT465I   VARIABLE varname SET TO value

Explanation: The variable, 'varname', named in message DSNT456I is in error. Possible reasons for the error are listed in messages: DSNT458I, DSNT459I, DSNT460I, DSNT462I, DSNT463I, DSNT464I.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs. The value has been set to the value shown.

User response: Specify the output data again on the primary installation panel. You will be able to see and change the value. If this value is not acceptable, change it, noting the limits.

DSNT466I   Output member cannot be DSNTIDSA, DSNTIDVB or DSNTIDXA.

Explanation: The OUTPUT MEMBER NAME field value must not be DSNTIDSA, DSNTIDVB, or DSNTIDXA. Members with these names are shipped with DB2 and must not be overwritten.

This message is issued by the following panel: DSNTIPA1

System action: The panel is re-displayed and the cursor is placed on the field in error.

User response: Change the value of the field under the cursor.

Important: If the member with the name that you specify already exists, it is overwritten.

DSNT467I   OUTPUT MEMBER SHOULD BEGIN WITH DSNTID

Explanation: The output parameter member will be placed into prefix.SDSNSAMP. This data set already has a large number of members. To avoid replacing a member that you need, a naming convention with prefix 'DSNTID' is strongly suggested. The suffixes 'VB' and 'XA' are not allowed (see message DSNT466I).

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

User response: Specify a different name for the output member.

Severity: 4 (warning)

DSNT468I   ERROR(S) ENCOUNTERED, OUTPUT IS BYPASSED

Explanation: An error was detected earlier in this run. Because of the error, data set editing was not performed.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue, but will not produce edited jobs.

User response: Correct the errors noted earlier, and specify the output prefix name again to produce the edited data.

Severity: 8 (error)

DSNT469I   INPUT PARAMETERS MUST BE READ BEFORE MIGRATION

Explanation: The INPUT MEMBER NAME on panel DSNTIPA1 must have a value to allow migration values to be read. This is to permit the new values to be initialized.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will return to the main installation panel, DSNTIPA1.

User response: Specify DSNTIDXA as the INPUT MEMBER NAME and specify the current value for the migration member.

Severity: 8 (error)

DSNT470I   WARNING, ONLY ONE VOLUME WAS SPECIFIED

Explanation: Only one volume serial number was specified on the installation panel or CLIST invocation. This does not allow adequate recovery or performance unless it is intended to be a single volume test system.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

User response: If this subsystem is intended to be recoverable, more than one volume is needed. Rerun the installation tailoring process, specifying at least one more volume on the DSNTIPA2 panel. If this is just a single volume test system, be sure that you specify a single copy of the log data sets. The boot strap data sets are small and not active, so that two are recommended even on a single volume system.

Severity: 4 (warning)

DSNT471I   BEGINNING PARAMETER INPUT MEMBER inmem

Explanation: The input parameter member, 'inmem', is being read. This data includes values for all of the installation parameters.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT472I   WORK FILE DATABASE NAME CANNOT START WITH DSNDB EXCEPT FOR DSNDB07

Explanation: The work file database name cannot start with the characters DSNDB unless the name is DSNDB07.

This message is issued by the following panel: DSNTIPK

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change the name so that it does not start with DSNDB.

DSNT473I   BEGINNING CHECK PHASE

Explanation: This message indicates the following:
- The values entered on the panels are being checked.
- Values which use the K or M suffix are being translated to numbers.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT474I   READING count, varname, datatype, data sharing scope, minval, maxval, currentval

Explanation: The values are being read from the parameter member. The count, the variable name, the data type, the data sharing scope, the minimum value, the maximum value, and the current value are shown. This message occurs when the CONTROL keyword requests a trace.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT475I   CHECK count, varname, type, minval, maxval, currentval

Explanation: This message indicates the following:
- The values read in or entered on the panels are being checked.
- Values which use the K or M suffix are being translated to numbers.

The count, the variable name, the data type, the minimum value, the maximum value, and the current value are shown. This message occurs when the CONTROL keyword requests a trace.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT476I   BEGINNING PARAMETER OUTPUT MEMBER outmem

Explanation: The output parameter member named 'outmem' is being written. This data includes values for all the installation parameters.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT477I   OUTPUT count, varname, type, minval, maxval, currentval

Explanation: The values entered on the panels are being written to the output parameter member. The count, the variable name, the data type, the minimum value, the maximum value, and the current value are shown. This message occurs when the CONTROL keyword requests a trace.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT478I   BEGINNING EDITED DATA SET OUTPUT

Explanation: The output jobs and data are being edited to tailor them. Edited JCL output was requested and no severe errors were encountered.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST will continue.

Severity: 0 (successful execution)

DSNT479I   THE IRLM GROUP NAME AND DB2 GROUP NAME MUST BE DIFFERENT

Explanation: You entered an IRLM group name on panel DSNTIPJ that is the same as the DB2 group name entered on panel DSNTIPK. The names must be different.

This message is issued by the following panel: DSNTIPJ

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Either change the DB2 group name entered on panel DSNTIPK or the IRLM group name on panel DSNTIPJ.

DSNT480I   NAME MUST BE 1-16 CHARS, BEGIN WITH A-Z, AND NOT INCLUDE #, @, OR $.

Explanation: The location name specified on panel DSNTIPR is invalid. The location name must be 1-16 alphanumeric characters, excluding the alphabetic extenders #, @ or $, and must begin with a letter.

This message is issued by panel: DSNTIPR

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change the location name to meet the allowable criteria.

DSNT481I   **DSNTINST** *data set* **DATA SET SIZE MAY BE TOO SMALL**

Explanation: The DSNTINST CLIST encountered an arithmetic overflow during calculation of this data set's size.

System action: DSNTINST continues processing and uses 2147483647 for the result of the calculation that caused the overflow. Thus, the size of the data set might be too small.

User response: Check the data set size in installation job DSNTIJIN. If it is too small, increase it.

DSNT482I   **TOO MANY CONCURRENTLY ALLOCATED DATA SETS,** *opnds*

Explanation: The number (*opnds*) calculated for the number of concurrent databases, tables per database, and table spaces per database, is too large. The maximum number of data sets that are allowed is 100,000. See DSNT436I and DSNT437I for more information.

The number of concurrently allocated data sets is calculated using the formula:

```
OPNDS = (NUMCONDB X NUMTABLE X NUMINDEX) +
(NUMCONDB X NUMTABSP)
```

Where   Represents

NUMCONDB
> The maximum number of concurrently active databases (DATABASES on panel DSNTIPE)

NUMTABLE
> The estimated average number of tables per database (TABLES on panel DSNTIPD)

NUMINDEX
> The estimated average number of indexes per table (set to 2)

NUMTABSP
> The estimated average number of table spaces per database (TABLE SPACES on panel DSNTIPD)

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST reduces the value of NUMCONDB so that the calculated value for OPNDS is within the allowed range. CLIST processing continues.

User response: Do one of the following:

• Accept the reduced value for NUMCONDB calculated by the CLIST.
• Reenter smaller values for NUMCONDB, NUMTABLE, or NUMTABSP on the SIZES and STORAGE SIZES panels. You can do this by either using the END key to go back to those panels or by running the CLIST again.

Severity: 4 (warning)

DSNT483I   **NAME MUST BE 1-8 CHARS AND MUST BEGIN WITH A-Z**

Explanation: The DB2 data sharing group name specified on panel DSNTIPK or the IRLM data sharing group name specified on panel DSNTIPJ is invalid. The DB2 or IRLM group name must be 1-8 characters, and must begin with a letter.

This message is issued by panels: DSNTIPJ, DSNTIPK

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Change the group name to meet the allowable criteria.

DSNT485I   *type* **SIZE =** *size* **K**

Explanation: This is the estimated main storage size for part of the DB2 database address space. This is the most constrained address space. The *type* can be BUFFER POOL, SORT POOL SIZE, RID POOL SIZE, EDMPOOL STORAGE, DATA SET STORAGE, CODE STORAGE, or WORKING STORAGE.

The storage for buffer pools, sort pools, rid pools, EDMPOOL, and code is above the 16MB line. Most of the main storage for data set control blocks is below the 16MB line and is included in the region size. Working storage includes storage above and below the 16MB line, but storage above the line is not included in the region.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST continues.

Severity: 0 (successful execution)

Related tasks:

Planning storage for DB2 (DB2 Installation and Migration)

Invoking the CLIST (DB2 Installation and Migration)

DSNT486I   TOTAL MAIN STORAGE = *totregion* **K**

Explanation: This is the estimated total region size. It includes storage above the 16MB line.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST continues.

Severity: 0 (successful execution)

Related tasks:

Planning storage for DB2 (DB2 Installation and Migration)

Invoking the CLIST (DB2 Installation and Migration)

DSNT487I   TOTAL STORAGE BELOW 16MB = *region* **K (ASSUMING SWA IS ABOVE THE LINE)**

Explanation: This message provides an estimate of the virtual storage required below the 16MB line to check for virtual storage constraints. This calculation presumes that you have moved the SWA for DB2 above the 16MB line.

This message is issued by the following installation CLIST: DSNTINST

System action: The CLIST continues.

User response: Check for virtual storage constraints. Your MVS should permit at least this amount of virtual storage below the 16MB line.

Severity: 0 (successful execution)

Related tasks:

Calculating virtual storage below the 16-MB line (DB2 Installation and Migration)

Invoking the CLIST (DB2 Installation and Migration)

DSNT488I   VOLUME voln WILL REQUIRE AT LEAST nn 4K BLOCKS

Explanation: This message states the number of 4KB blocks of space that will be required for the specified volume in order to install or migrate the current release of DB2.

Volume 'voln' is specified in the VOLUME SERIAL fields on installation panel DSNTIPA2, or with the VOLSDATn input parameters to the Install CLIST. The message indicates how much space will be needed on the selected volume to install or migrate this release of DB2.

This message is issued by the following installation CLIST: DSNTINST

System action: Processing continues.

User response: Ensure that the volume has at least as much available storage as the message indicates it should have. If it does not, rerun the INSTALL CLIST specifying different volumes, or create more available space on the originally specified volume.

In order to convert the number of 4KB blocks required to the number of cylinders required, use the table below. Divide the number of 4KB blocks (in the DSNT488I message) by the number of 4KB blocks per cylinder (in the table) to get the number of cylinders required.

Device Type	4KB Blocks per Cylinder	4KB Blocks per Track
3330	57	3
3340	24	2
3350	120	4
3375	96	8
3380	150	10

These size estimates are for system data only. They do not include: user data, archive logs, image copies, or temporary work data sets other than the temporary database.

Severity: 0 (successful execution)

DSNT489I   CLIST EDITING *dsname(member), explanation*

Explanation: The CLIST is editing the data set and member *dsname(member)*. The *explanation* is the type of data being edited and has one of these values:

INSTALL JCL
     Jobs that are edited and run as directed during the installation process.

MIGRATE JCL
     Jobs that are edited and run as directed only during the migration process.

FALL BACK JCL
     A job that is only run as directed when falling back to the previous release.

RECOVERY JCL
     A job that is only run as directed during a recovery process. It deletes the DB2 data sets.

SAMPLE JCL
     Jobs that are edited and run during installation verification to add the sample application.

SAMPLE DATA
     Edited data that installation verification jobs require.

CLIST   Customized CLISTs to reflect your names used during the installation process.

This message is issued by the following installation CLISTs:

DSNTINST   DSNTINS1   DSNTINS2

System action: The CLIST continues.

Severity: 0 (successful execution)

DSNT490I   SAMPLE DATA UNLOAD PROGRAM

Explanation: This is the header, indicating a normal start for the application program.

DSNT491I   TOO MANY TABLES ENTERED (MAXIMUM OF 100 TABLES)

Explanation: The number of tables entered for unloading exceeded the maximum of 100 tables. The first 100 tables in the input should have been processed.

System action: The application program is terminated.

User response: Split the run into multiple runs, none with more than 100 tables.

Problem determination: Check the application program input. Make sure there are no more than 100 tables.

DSNT492I   **SQL WARNING DURING SQL STATEMENT** *statement-type*, **TABLE** *table-name*

Explanation: The SQL statement might not have executed correctly.

statement-type
          The type of SQL statement that encountered the error: PREPARE, OPEN, FETCH, or CLOSE,

table-name
          The name of the table or view that is being unloaded.

System action: The request might not be executed correctly.

User response: Take the action that is indicated by the return code for the SQL statement.

Problem determination: Determine which SQL statement in the application program caused the warning and see if it is reproducible. Follow the diagnosis approach suggested for the warning.

Related concepts:

▶ SQL codes (DB2 Codes)

DSNT493I   **SQL ERROR DURING SQL STATEMENT** *statement-type*, **TABLE** *table-name*

Explanation: The SQL statement may not be executed correctly.

statement-type
          The type of SQL statement that encountered the error: PREPARE, OPEN, FETCH, or CLOSE,

table-name
          The name of the table or view that is being unloaded.

System action: The request was not executed correctly.

User response: Take the action that is indicated by the return code for the SQL statement.

Problem determination: Determine which SQL statement in the application program caused the error and try to reproduce the problem. Follow the diagnosis approach suggested for the SQL return code for that statement.

Related concepts:

▶ SQL codes (DB2 Codes)

DSNT494I   **ERROR DURING OPEN OF DDNAME SYSRECnn**

Explanation: The ddname listed, where 'nn' ranges from 00 to 99, could not be successfully opened. A possible reason is that the DD statement to the data set was not present in the JCL.

System action: The application program is terminated.

User response: Check the JCL to ensure that this DD statement is provided and is able to be used for output.

Problem determination: Make sure that the same number of DD statements and table names were provided.

DSNT495I   **SUCCESSFUL UNLOAD n ROWS OF TABLE tttttttt**

Explanation: The message indicates that 'n' rows of table or view 'tttttttt' were unloaded. The 'n' rows were counted during the unload processing.

System action: The request was executed successfully.

DSNT496I   **UNRECOGNIZED DATA TYPE CODE OF nnnnn**

Explanation: The PREPARE returned an invalid data type code of decimal 'nnnnn'.

System action: The next table is processed.

System programmer response: See the Problem Determination section.

User response: Notify the system programmer.

Problem determination: There is an error in the unload program or a new data type has been added. The

recognized data types are fullword integer, halfword integer, fixed-length character, variable length character, packed decimal, floating-point, fixed-length graphic, variable length graphic, and long, variable length graphic.

DSNT497I   RETURN CODE FROM MESSAGE ROUTINE DSNTIAR

Explanation: The message formatting routine detected an error.

System action: The application program is terminated.

System programmer response: See the Problem Determination section.

User response: Notify the system programmer.

Problem determination: There is an error in the unload program. The return code from DSNTIAR is given in the MVS output of your job. When reporting this problem, have the return code, the application program listing, and JCL available.

DSNT498I   ERROR, NO VALID COLUMNS FOUND

Explanation: The PREPARE returned data which did not produce a valid output record. The data was not recognized as valid, so the columns were not unloaded.

System action: The next table is processed.

System programmer response: See the Problem Determination section.

User response: Notify the system programmer.

Problem determination: There is an error in the unload program. When reporting this problem, have the application program listing and its JCL available.

DSNT499I   ERROR IN INPUT PARAMETER LIST

Explanation: The parameter list specified when DSNTIAUL or DSNTEP2 was invoked contains one or more invalid values.

System action: The application program is terminated.

System programmer response: Check the parameter values entered when DSNTIAUL or DSNTEP2 was invoked. For DSNTIAUL, the correct parameter value is 'SQL'. This parameter indicates that DSNTIAUL input is in complete SQL statements terminated by a semicolon. For DSNTEP2, the correct parameter values are either 'ALIGN(LHS)', 'ALIGN(MID)', 'MIXED', or 'NOMIXED'. The first two parameter values represent the two output alignment options available in DSNTEP2 where 'ALIGN(LHS)' will align the output on the left margin and 'ALIGN(MID)' will center the output. The remaining two parameter values designate whether or not DSNTEP2 will recognize mixed character data where 'MIXED' means that it will and 'NOMIXED' means that it will not recognize mixed character data.

User response: Notify the system programmer.

DSNT500I   *csect-name* RESOURCE UNAVAILABLE REASON *reason-code* TYPE *resource-type* NAME *resource-name*

Explanation: The requested function requires a resource that is currently unavailable.

csect-name
        The name of the control section that issued the message.

reason-code
        A numeric value that indicates the reason for the failure.

resource-type
        The type, format, and content of the resource that is identified by *resource-name*.

resource-name
        The name of the resource that is currently unavailable.

System action: Additional messages and codes are issued to identify the system action and the action that should be taken by the installation or operator.

DSNT501I

Operator response: Use the *reason-code, resource-type,* and *resource-name* values provided in the message to identify the source of the problem.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

If *resource-type* is X'D00' or X'D01', use the following SQL statement to determine the table name:

```
SELECT CREATOR, NAME
   FROM SYSIBM.SYSTABLES
   WHERE DBID = dbid AND OBID = obid;
```

This statement can either fail or return an incorrect name if the table has been dropped.

Related concepts:

⏩ DB2 reason codes (DB2 Codes)

Related information:

Appendix C, "Resource types," on page 1153

DSNT501I   *csect-name* **RESOURCE UNAVAILABLE**

```
CORRELATION-ID = correlation-id
CONNECTION-ID  = connection-id
LUW-ID         = luw-identifier
REASON         = reason-code
TYPE           = resource-type
NAME           = resource-name
```

Explanation: The requested function requires a resource that is currently unavailable. This message is issued with other messages that identify the system action and the action the installer or operator should take.

DB2 can recover from a resource not available in some cases and can complete successfully. One example is during setup of a performance enhancement during bind time for execution time. If DB2 recovers successfully, then no action is necessary.

For TYPE=905, REASON=00C90092, and NAME=IRLM, message DSNT501I is usually issued once. It is suppressed at the DB2 subsystem level until the IRLM out-of-storage condition is relieved.

correlation-id
> The correlation identifier of the thread associated with the requester of the resource.

connection-id
> The connection identifier of the requester.

luw-id   The logical-unit-of-work identifier (LUW-ID) of the thread associated with the requester of the resource. An asterisk (*) indicates that the thread does not have a LUW-ID. If the LUW-ID is not an asterisk, the value is a logical-unit-of-work-id assigned to the thread. This is followed by an equal sign (=) and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

resource-type
> The type of resource that is currently unavailable. The 'type' identifies the type, as well as the format and content, of the resource 'name'. Refer to Appendix C, "Resource types," on page 1153 to find the type of resource identified in the message.

resource-name
> The name of the resource that is currently unavailable.

reason-code
> The reason code that tells why the resource is unavailable.

Operator response: Note the REASON, TYPE, NAME values.

System programmer response: To perform the operation on the object, you must migrate the catalog to the current release. To do this, run DSNTIJTC.

Problem determination: A type X'401' indicates that a cascading revoke of SYSADM or SYSCTRL in a lower-level

release attempted to revoke privileges or grants on objects that are available only in a higher-level release. Because the cascading revoke affects objects that are unknown to the lower-level release, you can only revoke SYSADM or SYSCTRL from the higher-lever release.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

Related tasks:

➡ Installation step 15: Tailor the DB2 catalog: DSNTIJTC (DB2 Installation and Migration)

➡ Migration step 18: Tailor DB2 Version 11 catalog: DSNTIJTC (DB2 Installation and Migration)

➡ Populating the communications database for use with TCP/IP (DB2 Installation and Migration)

➡ Populating the communications database (DB2 Installation and Migration)

➡ Connecting systems with VTAM (DB2 Installation and Migration)

DSNT502I   ERROR IN DSNTIAUL SQL INPUT: statement

Explanation: The DSNTIAUL input SQL statement could not be prepared.

System action: The application program terminates.

System programmer response: Check that the DSNTIAUL input statement is a complete SQL statement, ending with a semicolon.

User response: Notify the system programmer.

DSNT503I   UNLOAD DATA SET *ddname* RECORD LENGTH SET TO *nnnnn*.

Explanation: If the ddname is SYSRECnn, the record length specified in the JCL for the data set is not large enough to contain an entire row of unloaded data, or the record length was not specified in the JCL.

If the ddname is SYSPUNCH, the record length specified in the JCL is not between 80 and 255, or the record length was not specified in the JCL.

System action: DB2 sets the record length of the unload data set to the value specified in the message text.

DSNT504I   UNLOAD DATA SET *ddname* BLOCKSIZE SET TO *nnnnn*.

Explanation: The block size for the SYSRECnn or SYSPUNCH data set was one of the following:

- Not specified in the JCL
- Set to zero in the JCL
- Less than the required record length
- Not an even multiple of the required record length

System action: DB2 sets the block size of the unload data set to the value specified in the message text.

DSNT505I   DSNTIAUL OPTIONS USED: 'nnn'.

Explanation: You specified these options when you invoked DSNTIAUL. 'SQL' is the only valid option.

Severity: 0 (informational)

DSNT506I   INPUT STATEMENT WAS NOT A FULL SELECT ON A SINGLE TABLE. LOAD STATEMENT WILL NEED MODIFICATION.

Explanation: The SELECT statement used to request a table unload contained one of the following:

- A SELECT clause other than "SELECT * FROM"
- A FROM clause that does not contain exactly one table name

The table to be loaded is therefore not the same as the table from which the data is unloaded. DSNTIAUL is unable to determine the table name and all column names for the table to be loaded.

System action: DSNTIAUL generates a LOAD statement containing the string 'TBLNAME' for the table name. Some column names may be blank.

User response: Change the LOAD statement table name to the name of a table with the indicated column descriptions.

Severity: 4 (warning)

DSNT507I *csect-name* **RECORD LENGTH NEEDED IS GREATER THAN MAXIMUM ALLOWED FOR TABLE** *table-name*

Explanation: In order to unload table *table-name*, DB2 needs an output data set with a record length greater than the system maximum of 32760 bytes. This occurs when the table to be unloaded is in a 32KB page table space and contains a combination of long columns and a number of date, time, or timestamp columns.

System action: The application program is terminated.

Severity: 8 (error)

DSNT508I **WARNING - NONSTANDARD JOB CARD (***cardstat***), NO JOB CARDS WILL BE ADDED TO THE EDITED JOBS**

Explanation: The job card you entered in field 3 of panel DSNTIPY contains one or more of the following errors:
- Less than 7 characters ('cardstat' = TOO SHORT)
- "//" missing as the first 2 characters ('cardstat' = MISSING //)
- Missing the word "JOB" ('cardstat' = MISSING "JOB"TOKEN)

This message is issued by the following installation CLIST: DSNTINST

System action: The message is written to the screen, after all panels are displayed and before any job editing. No JCL is added to the edited jobs.

User response: You can continue with job editing and manually add JCL to the edited jobs. Alternatively, you can repeat the installation process and make corrections to the job card entered on panel DSNTIPY.

DSNT509I **WARNING - MIGRATION INPUT MEMBER LEVEL IS** *rel_ind***. LEVEL** *prev_rel* **IS REQUIRED. RETURN TO PANEL DSNTIPA1 TO CHANGE MIGRATION INPUT MEMBER**

Explanation: The migration input member you entered on panel DSNTIPA1 contains incorrect release indicator 'rel_ind'. It should contain the indicator for the prior release ('prev_rel'). If your input member does not contain a release indicator, 'rel_ind' will be <null>. This probably indicates that you are using a down-level member.

This message is issued by the following installation CLIST: DSNTINST

System action: The message is written to the screen, after display of panel DSNTIPA1.

User response: Press ENTER to proceed to the next panel, and then PF3 to return to panel DSNTIPA1 in order to enter an input member with the correct release indicator.

DSNT510I **WARNING - INPUT MEMBER** *inmem* **LEVEL IS** *rel_ind***. LEVEL** *curr_rel* **IS REQUIRED. RETURN TO PANEL DSNTIPA1 TO CHANGE INPUT MEMBER**

Explanation: The input member 'inmem' you entered in field 6 on panel DSNTIPA1 contains incorrect release indicator 'rel_ind'. It should contain the indicator for the current release ('curr_rel'). If your input member does not contain a release indicator, 'rel_ind' will be <null>. This probably indicates that you are using a down-level member.

This message is issued by the following installation CLIST: DSNTINST

System action: The message is written to the screen, after display of panel DSNTIPA1.

User response: Press ENTER to proceed to the next panel, and then PF3 to return to panel DSNTIPA1 in order to enter an input member with the correct release indicator.

DSNT511I   FIELDS 2 AND 5 MUST BE UNIQUE FOR EACH NEW MEMBER INSTALLED

Explanation: This warning message is issued to remind you that the IRLM SUBSYSTEM and IRLM PROC names must be unique for each new member installed within a DB2 data sharing group. The installation process does not check for uniqueness of these names within a group.

This message is issued by the following panel: DSNTIPI

User response: Enter unique names in these fields for each member of a DB2 data sharing group.

DSNT512I   DSNT512I WARNING: ENTER UNIQUE NAMES FOR LUNAME AND LOCATION NAME

Explanation: This warning message is issued to remind you that you must enter values for DB2 LOCATION NAME and DB2 NETWORK LUNAME fields.

LUNAME is used to uniquely identify this DB2 subsystem within the network and to uniquely identify trace records for correlation of accounting records. The NETWORK LUNAME must be unique within the network. The installation process does not check for uniqueness of this name within a data sharing group.

The LOCATION NAME must be unique within the network, but must be the same on all members within a DB2 data sharing group.

This message is issued by the following panel: DSNTIPR

User response: Enter unique names in these fields.

DSNT513I   FIELD 2 MUST BE UNIQUE FOR EACH NEW MEMBER INSTALLED

Explanation: This warning message is issued to remind you that the DB2 PROC NAME field on panel DSNTIPX or the RESYNC PORT fieldon panel DSNTIP5 must be unique for each new member installed within a DB2 data sharing group. The installation process does not check for uniqueness of these values within a group.

This message is issued by the following panels: DSNTIPX and DSNTIP5

User response: Enter unique values in these fields for each member of a DB2 data sharing group.

DSNT514I   FIELDS 2 AND 3 MUST BE UNIQUE FOR EACH NEW MEMBER INSTALLED

Explanation: This warning message is issued to remind you that the MEMBER NAME and WORK FILE DB NAME must be unique for each new member installed within a DB2 data sharing group. The installation process does not check for uniqueness of these names within a group.

This message is issued by the following panel: DSNTIPK

User response: Enter unique names in these fields for each member of a DB2 data sharing group.

DSNT515I   THE *field-name* MUST BE UNIQUE FOR EACH NEW MEMBER INSTALLED

Explanation: This message indicates that the value specified for the of the *field-name* field must be unique for each new member installed within a DB2 data sharing group. The installation process does not check for uniqueness of these values within a group.

User response: Enter unique values in these fields for each member of a DB2 data sharing group.

DSNT516I   FIELDS 1, 2, AND 4 MUST BE UNIQUE FOR EACH NEW MEMBER INSTALLED

Explanation: This warning message is issued to remind you that the SUBSYSTEM NAME, COMMAND PREFIX, and SUBSYSTEM SEQUENCE fields must be unique for each new member installed within a DB2 data sharing group. The installation process does not check for uniqueness of these values within a group.

This message is issued by the following panel: DSNTIPM

User response: Enter unique values in these fields for each member of a DB2 data sharing group.

DSNT517I   **CHARACTERS 2-18 MUST BE A-Z, 0-9, #, @, $, OR UNDERSCORE**

Explanation: WLM environment names that do not meet the following criteria are not allowed. A WLM environment name may contain up to eighteen characters. The first character of a WLM environment name must either be an alphabetic character (A-Z) or a standard alphabetic extender (in the United States, the standard alphabetic extenders are '#', '@', '$'). For characters two through eighteen, a character may be an alphabetic character (A-Z), an alphabetic extender ('#', '@', '$'), a numeric character (0-9), or the underscore character ('_').

This message is issued by panel: DSNTIPX

System action: The panel is redisplayed, and the cursor is placed on the field with the error.

User response: Change the WLM environment name to meet the criteria described above.

DSNT518I   **THE VALUES FOR DRDA PORT, SECURE PORT AND RESYNC PORT CANNOT BE THE SAME**

Explanation: You entered the same value for DRDA PORT, SECURE PORT, and RESYNC PORT. The values for these fields must be different.

This message is issued by the following panel: DSNTIP5

System action: The panel is redisplayed and the cursor is placed in the field in error.

User response: Change either the DRDA PORT, SECURE PORT, or the RESYNC PORT value so that they are different.

DSNT519I   **TO SPECIFY A DRDA PORT OR A SECURE PORT (OR BOTH) YOU MUST ALSO SPECIFY A RESYNC PORT.**

Explanation: You entered a value for DRDA PORT or SECURE PORT but not for RESYNC PORT, or, you entered a value for RESYNC PORT but did not enter a value for either DRDA PORT or SECURE PORT.

When DRDA PORT, SECURE PORT, or both contain a non-blank, RESYNC PORT must also contain a non-blank.

This message is issued by the following panel: DSNTIP5

System action: The panel is redisplayed and the cursor is placed on the field in error.

User response: Enter values for DRDA PORT, SECURE PORT, or both, in conjunction with a value for RESYNC PORT. Or, remove all of the values from all of the port fields.

DSNT526I   **WARNING: DSNHDECP SPECIFIES AN UNDEFINED (0) VALUE FOR** *ccsid-parm* **NOTIFY THE SYSTEM PROGRAMMER.**

Explanation: The DSNHDECP module or user-supplied applications default module that is used by the DB2 precompiler contains an undefined (0) value for the CCSID parameter that is indicated by *ccsid-parm*. The use of CCSIDs that are undefined will be disallowed by DB2 in the near future.

System action: Processing continues.

System programmer response: The *dsnhdecp* module must specify a valid, non-zero default CCSID (coded character set identifier) for each of the three encoding schemes that are supported: ASCII, EBCDIC, and UNICODE. Processing character data with an undefined CCSID can lead to unpredictable results, including loss or corruption of the data. A default CCSID must be defined even for encoding schemes not used on your DB2 system.

If *ccsid-parm* is ASCCSID, then the default CCSID for single-byte ASCII-encoded character data is undefined (set to 0). If *ccsid-parm* is SCCSID, then the default CCSID for single-byte EBCDIC-encoded character data is undefined (set to 0).

You need to take steps now to select an appropriate CCSID for each undefined CCSID, and to plan for and schedule deployment of the updated *dsnhdecp* modules on your DB2 subsystem or data sharing system.

Do not attempt to choose or change a CCSID without guidance from IBM. Using the wrong CCSID can corrupt your data and render it unrecoverable.

User response: Notify the System Programmer.

DSNT527I   **ERROR: DB2 ATTEMPTED TO START USING THE DB2-SUPPLIED DSNHDECP MODULE. NOTIFY THE SYSTEM PROGRAMMER.**

Explanation: The DSNHDECP module that is supplied by DB2 is being used to run the DB2 precompiler. DB2 requires site-dependent settings that are specified in the DSNHDECP module that was created when DB2 was installed. The DSNHDECP that was generated during installation typically is located in the SDSNEXIT library.

System action: DB2 startup is discontinued.

System programmer response: Check the STEPLIB, JOBLIB, or system link list concatenation used in the DB2 address spaces. To ensure that the proper DSNHDECP module is used, verify that the library that contains the DSNHDECP module that is generated during installation precedes the SDSNLOAD library that contains the DSNHDECP module that was supplied by DB2.

User response: Notify the System Programmer.

DSNT528I   **USE PROTECTION MUST BE YES WHEN CACHE DYNAMIC SQL IS YES**

Explanation: You entered NO for USE PROTECTION on panel DSNTIPP and YES for CACHE DYNAMIC SQL on panel DSNTIP8. You must enter YES for USE PROTECTION if CACHE DYNAMIC SQL is YES.

System action: The panel is redisplayed, and the cursor is placed on the field that is in error.

User response: If you see this message on panel DSNTIPP, enter a value of YES for USE PROTECTION, or return to panel DSNTIP8 and enter NO for CACHE DYNAMIC SQL. If you see this message on panel DSNTIP8, enter a value of NO for CACHE DYNAMIC SQL, or go to panel DSNTIPP and enter YES for USE PROTECTION.

DSNT530I   *program-name* **COULD NOT RUN BECAUSE IT WAS NOT INVOKED FROM AN APF-AUTHORIZED LIBRARY**

Explanation: *program-name* could not run for one of the following reasons:

- *program-name* was not invoked from an APF-authorized library.
- One or more libraries in the STEPLIB or JOBLIB concatenation the job that invokes *program-name* is not APF-authorized.

System action: The program terminates.

System programmer response: Ensure that *program-name* is link-edited with option AC(1) and resides in an APF-authorized library. Ensure that all libraries in the STEPLIB and JOBLIB concatenation for the job that invokes *program-name* are APF authorized. If *program-name* is an external module for a DB2 stored procedure or user-defined function, ensure that the corresponding SYSIBM.SYSROUTINES entry specifies a WLM-established address space.

User response: Notify the system programmer.

DSNT531I   *schema.routine-name* **RECEIVED SQLCODE** *sqlcode* **FROM SQL STATEMENT** *sql-statement*

Explanation: When *schema.routine-name* executed *sql-statement*, DB2 returned the SQL warning or error *sql-code*.

System action: *schema.routine-name* terminates.

System programmer response: Check the messages in the SYSPRINT data set for the stored procedures address space where routine *schema.routine-name* runs. A DSNT408I message provides specific information about the SQL warning or error.

User response: Notify the system programmer.

Related concepts:

➥ SQL codes (DB2 Codes)

Related information:

DSNT532I   *schema.routine-name* **RECEIVED SAF RETURN CODE** *return-code* **FROM RACROUTE COMMAND**
    command.

Explanation: When *schema.routine-name* executed RACROUTE command *command*, the OS/390® System
Authorization Facility (SAF) returned an unexpected code *return-code*.

System action: *schema.routine-name* terminates.

System programmer response: Refer to the *z/OS Security Server RACROUTE Macro Reference* for more information
about the SAF error.

User response: Notify the system programmer.

DSNT533I   *schema.routine-name* **RECEIVED RETURN CODE** *return-code* **FROM MVS SERVICE** *service-name*,
    FUNCTION *function-name*.

Explanation: *schema.routine-name* executed MVS service *service-name* for function *function-name* and received an
unexpected code *return-code*.

System action: *schema.routine-name* terminates.

System programmer response: Refer to the discussion of service *service-name* in the *z/OS MVS Programming:
Authorized Assembler Services Reference Volumes 1-4* for assistance with diagnosing the error.

User response: Notify the system programmer.

DSNT534I   *schema.routine-name* **RECEIVED AN ALERT FROM THE EXTENDED MCS CONSOLE:** *alert-type*.

Explanation: *schema.routine-name* has activated an extended MCS console to monitor system messages. This console
has posted an alert as indicated by *alert-type*.

System action: *schema.routine-name* terminates.

System programmer response: If *routine-name* is the DB2-supplied stored procedure WLM_REFRESH then this
message indicates an error in program DSNTWR. Refer to the *z/OS MVS Programming: Authorized Assembler Services
Guide*, the subtopic entitled "What to Do if Message Queuing Stops in the discussion of What Is an Extended MCS
Console?" for help. If this information does not help you, contact IBM for assistance.

User response: Notify the system programmer.

DSNT540I   *wlm-environment-name* **WAS REFRESHED BY** *user* **USING AUTHORITY FROM SQL ID** *sqlid*

Explanation: DB2 stored procedure WLM_REFRESH refreshed WLM environment *wlm-environment-name*. User ID
user issued the request, using authority from SQL ID *sqlid*.

System action: WLM environment *wlm-environment-name* is refreshed.

System programmer response: None

User response: None

DSNT541I   **WLM REFRESH REQUEST FAILED. RESOURCE PROFILE** *subsystem-name*.**WLM_REFRESH.***wlm-*
    environment-name **IS UNDEFINED**

Explanation: DB2 stored procedure WLM_REFRESH rejected a request to refresh WLM environment
wlm-environment-name because SAF resource profile *subsystem-name*.WLM_REFRESH.*wlm-environment-name* does not
exist. WLM_REFRESH uses the resource profile to verify whether the requestor is authorized to refresh the WLM
environment. The requestor was connected to DB2 subsystem *subsystem-name*.

System action: WLM_REFRESH rejects the refresh request.

System programmer response: Create the SAF resource profile in resource class DSNR, and permit READ access to
all primary and secondary authorization IDs that need to refresh the WLM environment.

User response: Notify the system programmer.

DSNT542I **SQL ID** *sqlid* **IS NOT AUTHORIZED TO REFRESH WLM ENVIRONMENT** *wlm-environment-name*

Explanation: DB2 stored procedure WLM_REFRESH rejected a request to refresh WLM environment *wlm-environment-name* because *sqlid* is not authorized. *sqlid* is the current SQLID of the requestor.

System action: WLM_REFRESH rejects the refresh request.

System programmer response: Give *sqlid* READ access or higher on the appropriate SAF resource profile. That profile resides in SAF resource class DSNR and has the format *ssid*.WLM_REFRESH.*wlm-environment-name*.

User response: Notify the system programmer.

DSNT543I **WLM REFRESH REQUEST FAILED: PARAMETER** *parameter* **IS NULL, BLANK, OR CONTAINS INVALID CHARACTERS**

Explanation: The argument passed to WLM_REFRESH for *parameter* cannot be used for one of the following reasons:

- It is null or blank
- It contains characters that are not allowed for a SAF resource profile or for a DB2 subsystem name

System action: The program terminates.

System programmer response:

- For the WLM_ENV_NAME parameter, specify a non-null, non-blank argument. That argument must conform to the rules for a SAF resource profile.
- For the SSID parameter, specify a null or blank argument, or a value that conforms to the rules for a DB2 subsystem name. If you specify a null or blank value, DB2 determines the subsystem name.

User response: Notify the system programmer.

Related reference:

➡ DB2 administration routines panel: DSNTIPRL (DB2 Installation and Migration)

➡ SUBSYSTEM NAME field (SSID DECP value) (DB2 Installation and Migration)

➡ WLM_REFRESH stored procedure (DB2 Application programming and SQL)

DSNT544I **SQLID** *sqlid* **IS UNDEFINED OR** *user* **IS NOT PERMITTED TO USE IT TO REFRESH** *wlm-environment-name*

Explanation: DB2 stored procedure WLM_REFRESH rejected a request by user id *user* to refresh WLM environment *wlm-environment-name* using authorization id *sqlid*. The request was denied either because *sqlid* does not exist or because *user* is not permitted to use it.

System action: WLM_REFRESH rejects the refresh request.

System programmer response: If *sqlid* exists, permit *user* to use it. Otherwise, create *sqlid*, give it READ access or higher on the appropriate SAF resource profile (which resides in SAF resource class DSNR and has the format *ssid*.WLM_REFRESH.*wlm-environment-name*), and permit *user* to use it.

User response: Notify the system programmer.

DSNT545I **WLM REFRESH REQUEST FAILED BECAUSE OF AN UNEXPECTED SYSTEM MESSAGE WITH IDENTIFIER** *message-identifier*

Explanation: The system has denied a WLM refresh request passed by DB2 stored procedure WLM_REFRESH. *message-identifier* is the identifier of a system message that was issued when the request failed.

System action: The refresh request is denied.

System programmer response: Examine the system log in the vicinity of the failed refresh request and locate message identifier *message-identifier*. Use this message and related messages to diagnose and correct the problem that is preventing the WLM refresh request.

User response: Notify the system programmer.

DSNT546I   **WLM REFRESH REQUEST CANCELLED BECAUSE AN EXTENDED MCS CONSOLE COULD NOT BE ACTIVATED WITHIN** *n* **SECONDS**

Explanation: DB2 stored procedure WLM_REFRESH rejected a request because it was unable to activate an extended MCS console within the allowed period of *n* seconds. The extended MCS console serializes the WLM_REFRESH requests and verifies that z/OS completed the refresh request.

System action: WLM_REFRESH rejects the refresh request.

System programmer response: This message indicates that WLM_REFRESH is busy processing one or more previous requests. Issue the DB2 command `-DIS PROC(SYSPROC.WLM_REFRESH)` and examine the ACTIVE, QUEUED, and MAXQUE values to analyze whether or not a large backlog of calls to WLM_REFRESH currently exists. To give WLM_REFRESH more time to respond to these requests, you can increase the wait time by updating the setting of the WAITLMT constant in the source code of the external module for WLM_REFRESH, DSNTWR. You need to prepare and bind DSNTWR again for the change to take effect.

User response: If you receive DSNT456I from subsequent calls to WLM_REFRESH, notify the system programmer.

DSNT547I   **THE RESULT OF THE WLM REFRESH REQUEST IS UNKNOWN BECAUSE THE SYSTEM DID NOT REPLY WITHIN** *n* **SECONDS**

Explanation: The DB2-supplied stored procedure, WLM_REFRESH, accepted and validated the request to refresh a WLM environment and passed it to z/OS for processing. However, z/OS did not respond within the wait period of *n* seconds.

System action: WLM_REFRESH terminates with return code 4.

System programmer response: This message indicates that your system is suppressing one or more of the messages that can be generated in response to the VARY WLM command. DSNTWR, the external module for WLM_REFRESH, uses an extended MCS console (also named DSNTWR) to monitor whether or not the VARY WLM command succeeded or failed. Messages with the IDs IWM032I (refresh complete) and IWM031I (refresh in progress) indicate that the refresh request processed successfully. So, for WLM_REFRESH to function correctly, you need to ensure that the system does not suppress messages IWM031I and IWM032I.

This error can also occur if the specified WLM environment requires more than *n* seconds to refresh. In this case, you need to increase the wait time allowed for the reply by updating the setting of the WAITLMT constant in the source code of the external module for WLM_REFRESH, DSNTWR. You need to prepare and bind DSNTWR again for the change to take effect.

User response: Notify the system programmer.

DSNT550I   *'data-set-name'* **IS UNAVAILABLE FOR CLIST OUTPUT BECAUSE** *reason*

Explanation: Data set *data-set-name* is required for output by the DB2 Install CLIST, but cannot be allocated for the reason that is indicated by *reason*. Possible reasons are:

- IT CANNOT BE ALLOCATED OR OPENED FOR OUTPUT

  An error occurred during an allocate, open, or close operation on *data-set-name.*

- ERROR PROCESSING REQUESTED DATA SET

  The request for *data-set-name* could not be processed.

- MEMBER SPECIFIED, BUT DATASET IS NOT PARTITIONED

  data-set-name already exists, but is not a partitioned data set as required by the CLIST.

- PROTECTED DATASET

  Your userid is not authorized to use *data-set-name*.

- VOLUME NOT ON SYSTEM

  data-set-name is registered in the system catalog on a DASD or tape volume which is currently unmounted or unavailable.

- UNAVAILABLE DATASET

  data-set-name is currently allocated by another user.

System action: The Install panel for specifying *data-set-name* is redisplayed. You cannot proceed to the next DB2 Install panel until the problem is resolved.

User response: Take one of the following actions according to *reason*:

- For errors allocating, opening, or processing *data-set-name*, specific diagnostic messages are displayed by TSO. If you are not seeing these messages, enter the command TSO PROFILE WTPMSG in the command line of the current DB2 install panel. Consult the appropriate MVS system messages and codes manual for further information about these messages.

- If *data-set-name* is indicated as a non-partitioned data set, verify that you have entered the intended data set name for CLIST output. If so, rename or delete the existing data set.

- Consult your system security administrator to obtain access to a protected data set.

- In the event of an unavailable data set, ask the current user(s) to release it. Ask your system programmer to identify the user(s) if you are unable to do so. Otherwise, try later when the data set has been freed.

DSNT551I   *csect-name bind-type* **FOR** *object-type* **=** *object-name* **FAILED BECAUSE CHARACTER CONVERSION FROM CCSID** *from-ccsid* **TO CCSID** *to-ccsid* **FAILED WITH ERROR CODE** *error-code* **FOR TABLE** *dbid.obid* **COLUMN** *column-number***.**

Explanation: A conversion error occurred during the conversion of a string to a different coded character set.

bind-type
    identifies the type of bind subcommand: BIND, REBIND or FREE.

object-type
    identifies the type of the object being bound: PLAN or PACKAGE.

object-name
    if 'object-type' is PLAN, 'object-name' is the name of the plan; otherwise, it is the name of the package in 'location.collection.package' format

from-ccsid
    identifies the coded character set of the string to be converted.

to-ccsid
    identifies the coded character set to which it must be converted.

error-code
    indicates the type of error as defined as follows:

        8   Length exception (for example, expansion required for PC MIXED data exceeds the maximum length of the string).

        12   Invalid code point.

        16   Form exception (for example, invalid MIXED data).

        20   Conversion procedure error (for example, an error during Unicode/390 stage 2 conversions or an exit set the length control field of the string to an invalid value).

        24   The conversion services have not been setup correctly or the conversion is not supported.

dbid.obid
    gives the database id and the table obid of the object with the conversion error. This could be either the target or the source of the conversion.

column-number
    gives the column number of the failing object. This could be either the target or the source of the conversion.

System action: The *bind-type* operation is not performed.

User response: If the *error-code* value is 24 and the conversion request is correct, check whether conversion support is correctly enabled.

For all other *error-code* values, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8

Related concepts:

➡ Introduction to character conversion (DB2 Internationalization Guide)

Related tasks:

➡ Setting up DB2 to ensure that it interprets characters correctly (DB2 Internationalization Guide)

DSNT552I   *csect-name* **CHARACTER CONVERSION FROM CCSID** *source-ccsid* **TO CCSID** *target-ccsid* **FAILED WITH ERROR CODE** *error-code*

Explanation: A conversion error occurred during the conversion of a string to a different coded character set.

source-ccsid
    The coded character set of the string to be converted.

target-ccsid
    The coded character set that the string is to be converted to.

error-code
    The type of error :

8	Length exception (for example, expansion required for PC MIXED data exceeds the maximum length of the string).
12	Invalid code point.
16	Form exception (for example, invalid MIXED data).
20	Conversion procedure error (for example, an error during Unicode/390 stage 2 conversions or an exit set the length control field of the string to an invalid value).
24	The conversion services have not been setup correctly or the conversion is not supported.
28	Either inconsistent data (for example, as a string length less than 0) was provided or Unicode System Services issued an invalid return code.

System action: The operation is not performed.

User response: Use the MVS command D UNI,ALL to see the conversions that have been defined.

If the *error-code* value is 24 and the conversion request is correct, check whether conversion support is correctly enabled.

For all other *error-code* values, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

Related concepts:

➡ Introduction to character conversion (DB2 Internationalization Guide)

Related tasks:

➡ Setting up DB2 to ensure that it interprets characters correctly (DB2 Internationalization Guide)

DSNT555I   **EXCESS** *ts-type* **PRIMARY SPACE HAS BEEN LOWERED FROM** *specified-space* **MB TO** *adjusted-space* **MB**

Explanation: This message displays on installation panel DSNTIP9, WORK FILE DATABASE, if the amount of space specified for the type of work file table space, when divided by the number of such table spaces, exceeds the system limit.

ts-type   The type of work file table space. Expected values are 4K SORT, 32K SORT, 4K TEMP, or 32K TEMP.

specified-space
    The amount of primary space (in megabytes) that is specified for the indicated type of work file table space.

adjusted-space
    The recalculated amount of primary space (in megabytes) for the indicated type of work file table space.

This message displays separately for each type of work file table space.

System action: The installation CLIST automatically reduces the specified total space to the indicated size.

User response: To accept the adjusted size and continue, press Enter.

To specify a different adjustment, change the amount of space, the number of table spaces, or both. The adjusted calculation must comply with the following requirement:

`specified-space / ts-count < 64`

Where:

specified-space
>   The value indicated in the message.

ts-count   The number of table spaces of the type indicated in the message by *ts-type*.

max-size
>   The maximum size for the type of work file table space indicated in the message, specified in gigabytes.
>
>   - 4K SORT: 64 GB
>   - 32K SORT: 64 GB
>   - 4K TEMP: 16384 GB (16 TB)
>   - 32K TEMP: 131072 GB (128 TB)

Related reference:

⇨ Work file database panel: DSNTIP9 (DB2 Installation and Migration)

DSNT556I   **THE SUM OF 4K SORT WORK TS'S, 32K SORT WORK TS'S, 4K TEMP WORK TS'S AND 32K TEMP WORK TS'S CANNOT EXCEED 500**

Explanation: This message displays on installation panel DSNTIP9 when the sum of the values entered for the different types of work file databases exceeds 500, which is the maximum number of table spaces permitted in the work file database.

System action: Panel DSNTIP9 is displayed again, and the cursor is placed in the 4K SORT WORK TS'S field.

User response: Reduce the total number of table spaces specified in the following fields so that the sum of all of these fields does not exceed 500:

- 4K SORT WORK TS'S
- 32K SORT WORK TS'S
- 4K TEMP WORK TS'S
- 32K TEMP WORK TS'S

Related reference:

⇨ Work file database panel: DSNTIP9 (DB2 Installation and Migration)

DSNT557I   *ts-type* **WORK TS'S MUST BE GREATER THAN 0 WHEN** *ts-type* **PRIMARY SPACE IS GREATER THAN 0**

Explanation: This message displays on installation panel DSNTIP9, WORK FILE DATABASE, if the PRIMARY SPACE value for a type of work file table space is non-zero and the corresponding WORK TS'S value is 0. When a non-zero value is specified for a primary space, then a non-zero value is required for the corresponding work file table space option.

ts-type   The type of work file table space. Expected values are 4K SORT, 32K SORT, 4K TEMP, or 32K TEMP.

System action: Panel DSNTIP9 is displayed again, and the cursor is placed in the *ts-type* PRIMARY SPACE field.

Programmer response: Either reduce the *ts-type* PRIMARY SPACE value to 0 or enter a non-zero value for the corresponding WORK TS'S entry. Reducing the PRIMARY SPACE value prevents the creation of additional work file table spaces of the indicated type as part of the migration process.

Related reference:

↪ Work file database panel: DSNTIP9 (DB2 Installation and Migration)

DSNT558I   *ts-type* **PRIMARY SPACE MUST BE GREATER THAN 0 WHEN TEMP** *ts-type* **WORK TS'S IS GREATER THAN 0**

Explanation: This message displays on installation panel DSNTIP9, WORK FILE DATABASE, if the WORK TS'S value for a type of work file table space is non-zero and the corresponding PRIMARY SPACE value is 0. When a non-zero value is specified for a work file table space, then a non-zero value is required for the corresponding primary space option.

ts-type   The type of work file table space. Expected values are 4K SORT, 32K SORT, 4K TEMP, or 32K TEMP.

System action: Panel DSNTIP9 is displayed again, and the cursor is placed in the *ts-type* PRIMARY SPACE field.

Programmer response: Either reduce the *ts-type* WORK TS'S value to 0 or enter a non-zero value for the corresponding PRIMARY SPACE entry. Reducing the WORK TS'S value prevents the creation of additional work file table spaces of the indicated type as part of the migration process.

Related reference:

↪ Work file database panel: DSNTIP9 (DB2 Installation and Migration)

DSNT560I   **ENTRY IS NOT A VALID** *entry-type*

Explanation: The entry in the indicated ISPF field has a syntax or other unexpected format that does not meet the requirements for a valid entry.

entry-type
          The type of entry, such as ROLE or WLM ENVIRONMENT NAME.

System action: The panel displays again, with the cursor on the field that contains the invalid entry.

User response: Correct the entry in the indicated field.

DSNT561I   **LOCKS PER TABLE(SPACE) MUST BE LESS THAN LOCKS PER USER**

Explanation: The current entry in the LOCKS PER TABLE(SPACE) field is greater than or equal to the value in the LOCKS PER USER field, which is not permitted.

System action: The panel displays again, with the cursor on the field that contains the invalid entry.

User response: Enter a value in the LOCKS PER TABLE(SPACE) field that is less than the value for LOCKS PER USER.

DSNT562I   **A ROLE NAME FOR SECURITY ADMIN** *field-indicator* **CANNOT BEGIN WITH 'SYS' AND CANNOT BE 'ACCESSCTRL', 'DATAACCESS', 'DBADM', 'DBCTRL', 'DBMAINT', 'NONE', 'NULL', 'PACKADM', 'PUBLIC', 'SECADM', or 'SQLADM'**

Explanation: The specified ROLE name is invalid because it starts with "SYS" or is one of the restricted values listed in the message text.

field-indicator
          An indicator of which field the message refers to: 1 or 2.

This check is performed only if ROLE in entered in the SEC ADMIN *field-indicator* TYPE field.

System action: The panel displays again, with the cursor on the field that contains the invalid entry.

User response: Change the value in the SECURITY ADMIN *field-indicator* field to comply with the rules for a ROLE name for a DB2 security administrator. If the entry is an authorization ID rather than a role, change the entry in the SEC ADMIN *field-indicator* TYPE field to AUTHID.

DSNT563I   AN ENTRY IN *nonblank-field* REQUIRES AN ENTRY IN *blank-field*

Explanation: The indicated installation panel fields must both contain either blank or nonblank values, but only one has a nonblank value.

nonblank-field
> The installation panel field that currently contains a nonblank value.

blank-field
> The installation panel field that currently contains a blank value.

System action: The installation panel displays again, with the cursor positioned in the field that is indicated by *nonblank-field*.

User response: Either enter blanks in the field that is indicated by *nonblank-field*, or enter a nonblank value in the field that is indicated by *blank-field*.

DSNT564I   THE ENTRY IN *first-field* CANNOT BE THE SAME AS THE ENTRY IN *second-field*

Explanation: Two installation panel fields that must have unique values have the same nonblank values.

first-field **and** *second-field*
> The installation panel fields that currently contain the same values.

System action: The installation panel displays again, with the cursor positioned in the field that is indicated by *first-field*.

User response: Update either or both of the indicated the fields so that their values are unique. Some fields can coexist with blank values in both. Review the installation panel information for requirements.

Related tasks:

Tailoring DB2 jobs to your environment using the installation CLIST (DB2 Installation and Migration)

DSNT566I   AN ENTRY OF *value-1* IN *field-1* REQUIRES AN ENTRY OF *value-2* IN *field-2*

Explanation: To specify a value of *value-1* in field *field-1*, you must first specify a value of *value-2* in field *field-2*.

value-1   The setting specified in field *field-1*.

field-1   The name of the installation panel field that currently contains a value that is dependent on the value specified in installation panel field *field-2*.

value-2   The setting specified in field *field-2*.

field-2   The name of the installation panel field that field *field-1* is dependent upon.

System action: The installation panel displays again. The cursor is positioned in the field that contains the dependent value.

Administrator response: Either specify *value-2* in *field-1* or specify *value-1* in *field-2*.

Severity: 8 (error)

DSNT568I   SAVE REQUEST IGNORED BECAUSE NO OUTPUT MEMBER WAS SPECIFIED

Explanation: A SAVE request was entered in the command line of the current DB2 installation panel but the change cannot be honored because no output member was specified in the OUTPUT MEMBER NAME field on the main panel, DSNTIPA1.

System action: The request is ignored.

User response: To continue to the next panel, press ENTER.

To exit the CLIST without saving your changes, press RETURN. Any changes that you have made will be lost. Then rerun the CLIST and specify an output member name on panel DSNTIPA1.

DSNT569I   **CHANGES HAVE BEEN SAVED TO OUTPUT MEMBER** *member-name*

Explanation: DB2 installation changes have been saved.

member-name
> The output member where the changes were saved.

System action: If the Save and continue option was selected, processing continues. Otherwise, the installation CLIST returns to the main panel.

User response: No action is required.

DSNT570I   **WARNING: THIS APPLICATION HAS SELECTED DSNTIAR TO PERFORM SQL MESSAGE FORMATTING BECAUSE THE GET DIAGNOSTICS SERVICE RETURNED AN UNEXPECTED SQLCODE WHILE FORMATTING MESSAGES FOR ONE OR MORE SQL WARNING OR ERROR CONDITIONS. A DUMP OF THE SQLCA FOR THE GET DIAGNOSTICS REQUEST FOLLOWS:**

```
SQLCODE: sqlcode
  SQLSTATE: sqlstate
  SQLERRP:  sqlerrp

SQLERRMC: sqlerrmc

SQLERRD1: sqlerrd1
  SQLERRD2: sqlerrd2
  SQLERRD3: sqlerrd3

SQLERRD4: sqlerrd4
  SQLERRD5: sqlerrd5
  SQLERRD6: sqlerrd6

SQLWARN0: w0 1: w1 2: w2
  3: w3 4: w4
  5: w5 6: w6
  7: w7 8: w8
  9: w9 A:wA
```

The dump of the SQLCA is shown here on multiple lines because of space limitations. The empty lines here indicate the correct line breaks.

Explanation: The application program called the GET DIAGNOSTICS service to format one or more of the warning or error conditions that occurred during the processing of the current SQL statement that was input into the system. However, the GET DIAGNOSTICS service also returned warnings or errors, as indicated by the SQLCA dump. The application therefore will use the DSNTIAR message formatting service instead of GET DIAGNOSTICS. **Attention:** Messages formatted by DSNTIAR might contain truncated tokens. Also, when multiple unexpected SQLCODEs were returned for the current inputted SQL statement, DSNTIAR reports on the first one.

System action: The application program continues.

System programmer response: See the problem determination section of this message.

User response: Notify the system programmer.

Problem determination: Use the SQLCA dump provided by DSNT570I to determine the cause of the problem. After the problem is resolved, run the application again.

DSNT571I   ERROR: THIS APPLICATION IS TERMINATING BECAUSE THE GET DIAGNOSTICS SERVICE
              RETURNED AN UNEXPECTED SQLCODE WHILE FORMATTING MESSAGES FOR ONE OR
              MORE SQL WARNING OR ERROR CONDITIONS FROM A ROWSET FETCH STATEMENT OR A
              NON-ATOMIC STATEMENT. A DUMP OF THE SQLCA FOR THE GET DIAGNOSTICS REQUEST
              FOLLOWS:

```
SQLCODE:  sqlcode
  SQLSTATE: sqlstate
  SQLERRP:  sqlerrp

SQLERRMC: sqlerrmc

SQLERRD1: sqlerrd1
  SQLERRD2: sqlerrd2
  SQLERRD3: sqlerrd3

SQLERRD4: sqlerrd4
  SQLERRD5: sqlerrd5
  SQLERRD6: sqlerrd6

SQLWARN0: w0 1: w1 2: w2
  3: w3 4: w4
  5: w5 6: w6
  7: w7 8: w8
  9: w9 A:wA
```

The dump of the SQLCA is shown here on multiple lines because of space limitations. The empty lines here indicate
the correct line breaks.

Explanation: The application program called the GET DIAGNOSTICS service to format one or more of the warning
or error conditions that occurred during the processing of a ROWSET FETCH statement or a non-atomic statement.
However, the GET DIAGNOSTICS request failed. Since the severity of the original warning or error conditions
cannot be determined, the application terminates after reporting this error.

System action: The application program is terminated.

User response: Notify the system programmer.

Problem determination: Use the SQLCA dump provided by DSNT571I to determine the cause of the problem. After
the problem is resolved, run the application again.

DSNT601I   THE DESCRIPTOR FOR *object-type object-name* DOES NOT EXIST IN DBD01

Explanation: An object that is defined in the DB2 catalog does not have a corresponding descriptor in the DBD01
table space of database DSNDB01.

object-type
        The type of object, either DATABASE or SPACE. A value of SPACE can refer to either a table space or an
        index space.

object-name
        The eight character name of the database, table space, or index space.

Possible causes for this error include:
- The DB2 catalog and the directory tablespace DSNDB01.DBD01 are out of sync.
- The DBD01 table space was not completely restored during a recovery process.
- There is a DB2 logic error.

System action: Processing continues.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For
information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis
Guide and Reference.

DSNT602I   NO SPACES FOUND

Explanation: This message is issued for multiple line WTO. It is produced when a DISPLAY DB(name) SPACENAM(...) is entered and no spaces were found to satisfy the request. SPACENAM(...) might have been entered as a range, a subset, or a list of spaces.

DSNT603I   spacetype name NOT type

Explanation: This message is issued for multiple-line WTO. It is produced when no table space or index space listed in the DISPLAY DB(...) SPACENAM(...) satisfies your request.

spacetype
> TABLESPACE if the space is a table space or INDEXSPACE if the space is an index space.

name   The name of the space.

type   A character string that indicates the type of request specified. The string is a list of the following, delimited by 'OR':
- 'ACTIVE' if the keyword ACTIVE is specified
- 'RESTRICTED' if the keyword RESTRICT is specified.

System action: Message DSNT301I or DSNT302I is produced if no object is found and the keywords ACTIVE and RESTRICT are omitted.

DSNT604I   NO INFORMATION AVAILABLE

Explanation: This message is issued for multiple line WTO. It is produced when there is no LOCK or USE information available. When DISPLAY DB(name) SPACENAM(...) is entered, and SPACENAM(...) is a list of spaces, this message indicates that no table spaces or index spaces whose names are listed have LOCK or USE information available. If DISPLAY DB(name) SPACENAM(...) is entered, and SPACENAM(...) is a range or a subset of spaces, no table spaces and index spaces within the specified database have LOCK or USE information available. This message is produced only when the keywords USE and LOCKS are specified.

DSNT694I   DATABASE name CANNOT BE STOPPED. IT CONTAINS THE DDL REGISTRATION TABLES

Explanation: The database specified in the message cannot be stopped because it contains the application and object registration tables. Installation SYSADM or installation SYSOPR authority is needed to stop the database.

System action: The database specified in the message is not stopped.

User response: Reissue the command with the proper authority.

DSNT695I   TABLESPACE name CANNOT BE STOPPED. IT CONTAINS A DDL REGISTRATION TABLE

Explanation: The table space specified in the message cannot be stopped because it contains an application or object registration table. Installation SYSADM or installation SYSOPR authority is needed to stop the table space.

System action: The table space specified in the message is not stopped.

User response: Reissue the command with the proper authority.

DSNT696I   INDEXSPACE name CANNOT BE STOPPED. IT IS DEFINED ON A DDL REGISTRATION TABLE

Explanation: The index space specified in the message cannot be stopped because it is defined on an application or object registration table. Installation SYSADM or installation SYSOPR authority is needed to stop the index space.

System action: The index space specified in the message is not stopped.

User response: Reissue the command with the proper authority.

DSNT700I *table-name* **IS THE ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: This message displays output from the DISPLAY RLIMIT command. This command reports the status of the resource limit facility (RLF), which is sometimes called the governor.

The output includes one of the following messages:

DSNT700I: Active
  This message is displayed if the governor is active.

DSNT701I: Not active
  This message is displayed if the governor is not active.

DISPLAY RLIMIT ends with message DSN9022I: Normal completion .

DSNT700I: Active:

Message DSNT700I indicates that the governor is active.

DSNT700I *table-name* IS THE ACTIVE RESOURCE LIMIT SPECIFICATION TABLE

table-name
  The table that the governor is using to determine resource limits.

If you want the governor to use a different resource limit specification table, issue the START RLIMIT command and specify the ID of the table.

If you do not want the governor to be active, issue the STOP RLIMIT command.

DSNT701I: Not active:

Message DSNT701I indicates that the governor is not active.

DSNT701I RESOURCE LIMIT FACILITY IS INACTIVE

If you want the governor to be active, issue the START RLIMIT command and specify the ID of the resource limit specification table to be used.

DSN9022I: Normal completion:

The DISPLAY RLIMIT output normally ends with message DSN9022I.

DSN9022I = DSNTCDIS 'DISPLAY RLIMIT' NORMAL COMPLETION

System action: Processing continues.

Related reference:

 -DISPLAY RLIMIT (DB2) (DB2 Commands)

 -START RLIMIT (DB2) (DB2 Commands)

 -STOP RLIMIT (DB2) (DB2 Commands)

Related information:
"DSN9022I" on page 1144

DSNT701I **RESOURCE LIMIT FACILITY IS INACTIVE**

Explanation: This message is a section of output from the DISPLAY RLIMIT command, which described in message "DSNT700I."

DSNT702I **RESOURCE LIMIT FACILITY HAS BEEN STOPPED. WAS USING table-name**

Explanation: This message is the normal response for a STOP RLIMIT command. Table 'table-name' was being used to determine resource limits.

System action: The governor is stopped.

System programmer response: May be used to resolve questions about what table was being used.

DSNT703I   RESOURCE LIMIT FACILITY IS ALREADY INACTIVE. STOP RLIMIT COMMAND IS IGNORED

Explanation: A STOP RLIMIT command was entered but the resource limit facility was already stopped. The command is ignored.

System action: The system continues normal processing.

DSNT704I   table-name HAS BEEN STARTED FOR THE RESOURCE LIMIT FACILITY

Explanation: This message is the normal response for a START RLIMIT command. Table 'table-name' is currently being used to determine resource limits.

System action: The governor has started utilizing the specified table.

System programmer response: May be used to resolve questions about what table is being used.

DSNT705I   csect-name **DATABASE** name **CANNOT BE STOPPED. IT CONTAINS AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Database name cannot be stopped because it contains an active resource limit specification table.

System action: The database is not stopped.

Operator response: If the database must be stopped, either stop the resource limit facility (STOP RLIMIT), or switch it to use a resource limit specification table that is not in the database (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a resource limit specification table that is not in the database.

DSNT706I   TABLE table-name **DOES NOT EXIST. THE START RLIMIT COMMAND WILL NOT USE THIS TABLE**

Explanation: The START RLIMIT command has been entered but was unable to use the table because it does not exist.

table-name
        The name of the table.

System action: RLF cannot use this table for governing.

Operator response: Verify that the 1 or 2 characters following DSNRLST or DSNRLMT in the table name are the same characters that were entered as the ID on the START RLIMIT command. If the ID appears correct, contact the system programmer.

System programmer response: A table with the fully-qualified name *qualifier.table-name* does not exist. The qualifier is the value of the RLFAUTH subsystem parameter. To determine the names of the of the existing resource limit specification tables, issue SQL statements like the following examples:

```
SELECT CREATOR,NAME FROM SYSIBM.SYSTABLES
   WHERE NAME LIKE 'DSNRLST'
```

```
SELECT CREATOR,NAME FROM SYSIBM.SYSTABLES
   WHERE NAME LIKE 'DSNRLMT'
```

If the query returns a row that matches *qualifier.table-name*, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

• Console output showing the failing START command

• The output of the SELECT command.

DSNT707I   COLUMN *column-name* IN TABLE *table-name* IS INCORRECTLY DEFINED. THE COLUMN
         DEFINITION IS *attribute* WITH LENGTH *length* AND NULL ATTRIBUTE *null-attribute*. THE
         COLUMN DEFINITION SHOULD BE *attribute* WITH LENGTH *length* AND NULL ATTRIBUTE
         null-attribute. THE START RLIMIT COMMAND WILL NOT USE THIS TABLE.

Explanation: The START RLIMIT command could not use table *table-name* because table *table-name* does not have
the proper column definition for a resource limit specification table.

System action: The START RLIMIT command will not use table *table-name* for governing.

Operator response: Contact the system programmer.

System programmer response: Table *table-name* does not have the proper column definition for a resource limit
specification table. Compare the current table column definition to the proper column definition given in the
message.

User response: See Operator Response.

Problem determination: Collect the console output showing the failing START command.

DSNT708I   INDEX *index-name* DOES NOT EXIST. THE START RLIMIT COMMAND WILL NOT USE THE
         ASSOCIATED TABLE

Explanation: The START RLIMIT command was unable to use the resource limit specification table DSNRLSTxx or
DSNRLMTxx because index DSNARLxx or index DSNMRLxx is not defined on that table.

System action: The START RLIMIT command will not use the resource limit specification table that is associated
with index *index-name*.

Operator response: Contact the system programmer.

System programmer response: Create the required index and reissue the START RLIMIT command.

User response: See Operator Response.

Problem determination: Collect the console output showing the failing START command messages.

DSNT709I   *new-table-name* NOW ACTIVE. *old-table-name* WAS OLD RESOURCE LIMIT SPECIFICATION
         TABLE

Explanation: A START RLIMIT command was entered while the facility was already active. The facility remains
active and switches from using the old table name to the new one.

Operator response: Notify the system programmer.

User response: Notify the system programmer.

DSNT710I   *table-name* ALREADY ACTIVE. START LIMIT COMMAND IGNORED

Explanation: A START RLIMIT command was entered and the governor is already active utilizing the specified
table. The command is ignored.

System action: The governor continues to use table 'table-name'.

Operator response: Verify that the ID specified on the START RLIMIT command is the correct ID. If not, reenter the
command specifying the correct ID. If it is correct, no action is needed because the facility is already active and is
using the correct table.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For
information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis
Guide and Reference.

User response: See Operator Response.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
  time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNT711I   COLUMN column-name IS MISSING IN TABLE table-name. SHOULD BE DEFINED AS attribute WITH LENGTH length AND NULL INDICATOR OF null-attribute

Explanation: The START RLIMIT command has been entered but failed because COLUMN 'column-name' in resource limit specification table 'table-name' is not defined. The proper column definition is given in the message.

System action: The START RLIMIT command is ignored.

Operator response: Verify that the 1 or 2 characters following DSNRLST are what was intended to be entered as the ID on the START RLIMIT command. If not, reenter the command specifying the proper ID. If the ID appears to be correct, contact the system programmer.

System programmer response: TABLE 'table-name' does not have COLUMN 'column-name' defined. Use ALTER TABLE to define the column as specified in the message.

User response: See Operator Response.

Problem determination: Collect the console output showing the failing START command.

DSNT712I   INDEX index-name IS NOT FOR TABLE table-name. THE START RLIMIT COMMAND WILL NOT USE THIS TABLE

Explanation: The START RLIMIT command was unable to use table table-name because index index-name is not defined for the resource limit specification table table-name.

System action: The START RLIMIT command will not use table table-name for governing.

Operator response: Contact the system programmer.

System programmer response: Drop the existing DSNARLxx or DSNRLMTxx index and create the correct index on the resource limit specification table table-name.

User response: See Operator Response.

Problem determination: Collect the console output showing the failing START command messages.

DSNT713I   INDEX index-name IS NOT UNIQUE. THE START RLIMIT COMMAND WILL NOT USE THE ASSOCIATED TABLE

Explanation: The START RLIMIT command has been entered but failed because INDEX index-name in the resource limit specification table is not defined as a UNIQUE index.

System action: The START RLIMIT command will not use the resource limit specification table associated with index index-name for governing.

Operator response: Contact the system programmer.

System programmer response: Drop INDEX index-name and create it as a unique index.

User response: See Operator Response.

Problem determination: Collect the console output showing the failing START command.

DSNT714I   INDEX index-name IS INCORRECTLY DEFINED. THE START RLIMIT COMMAND WILL NOT USE THE ASSOCIATED TABLE

Explanation: The START RLIMIT command was unable to use a resource limit specification table because index index-name is not properly defined.

System action: The START RLIMIT command will not use the resource limit specification table associated with index index-name for governing.

Operator response: Contact the system programmer.

System programmer response: Drop index index-name and create it with the correct definition.

User response: Contact the system programmer.

Problem determination: Collect the console output showing the START command messages.

Related tasks:

▶ Creating resource limit tables (DB2 Performance)

▶ Setting limits for system resource usage by using the resource limit facility (DB2 Performance)

Related reference:

▶ Resource limit facility tables (DB2 Performance)

▶ -START RLIMIT (DB2) (DB2 Commands)

DSNT715E   **ERROR reason-code READING table-name. default WILL BE RESOURCE LIMIT FOR CORRELATION-ID correlation-id, CONNECTION-ID connection-id, LUW-ID luw-id =token**

Explanation: An error occurred while the resource limit specification table 'table-name' was being read. The limit is determined from the default specified in the system parameters. This default can be:

NOLIMIT
        No limit applies.

NORUN
        Zero time limit.

integer   The limit to be used.

The reason code that defines the read error is included in the message text, as are the 'connection-ID', 'correlation-ID', and the logical-unit-of-work identifier (LUW-ID) of the thread. An LUW-ID of an '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input. Correlation IDs beginning with '0nn', where 'nn' is a number from 01 to 28, identify system agents.

System action: Processing continues using the system-defined default.

Operator response: Notify the system programmer.

System programmer response: This is an I/O error. Investigate the problem indicated by the reason code and preceding messages. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNT716E   **AN ERROR OCCURRED ESTABLISHING A RESOURCE LIMIT FOR CORRELATION-ID correlation-id, CONNECTION-ID connection-id, LUW-ID luw-id =token. limit WILL BE USED AS THE LIMIT**

Explanation: An internal error occurred while the resource limit for the specified requester was being established. The 'limit' is determined from the default specified in the system parameters. This default can be:

NOLIMIT
        No limit applies.

NORUN
        Zero time limit.

integer   The limit to be used.

A logical-unit-of-work identifier (LUW-ID) of '*' indicates that the thread does not have a LUW-ID. If the LUW-ID is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input. Correlation IDs beginning with '0nn', where 'nn' is a number from 01 to 28, identify system agents.

System action: Processing continues using the system-defined default.

Operator response: Notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNT717I   *csect-name* **TABLE SPACE** *name* **CANNOT BE STOPPED. IT CONTAINS AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Table space *name* cannot be stopped because it contains an active resource limit specification table.

System action: The table space is not stopped.

Operator response: If the table space must be stopped, either stop the resource limit facility (STOP RLIMIT), or switch it to use a resource limit specification table that is not in the table space (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a resource limit specification table that is not in the table space.

DSNT718I   *csect-name* **INDEX SPACE** *name* **CANNOT BE STOPPED. IT IS DEFINED ON AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Index space *name* cannot be stopped because it contains an index on an active resource limit specification table.

System action: The index space is not stopped.

Operator response: If the index space must be stopped, either stop the resource limit facility (STOP RLIMIT), or switch it to use a different resource limit specification table that does not have an index in the index space (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a different resource limit specification table that does not have an index in the index space.

DSNT719I   **TABLE** *table-name* **CANNOT BE READ. THE START RLIMIT COMMAND WILL NOT USE THIS TABLE**

Explanation: TABLE *table-name* cannot be accessed. This message might be preceded by related error messages that explain why the table cannot be accessed.

System action: The START RLIMIT command will not use table *table-name*.

Operator response: TABLE *table-name* cannot currently be accessed for reading. In the case of a deadlock or timeout, reissue the START RLIMIT command.

User response: See Operator Response.

DSNT720I   **csect-name INDEX** *index-name* **NEEDS TO BE** *index-type*. **THE START RLIMIT COMMAND WILL NOT USE THE ASSOCIATED TABLE**

Explanation: The START RLIMIT command was entered but failed because the named index defined for the resource limit specification table is not defined properly.

System action: The START RLIMIT command will not use the resource limit specification table associated with index *index-name* for governing.

Operator response: Contact the system programmer.

System programmer response: Drop index *index-name* and create it with the correct definition.

User response: Contact the system programmer.

Problem determination: Collect the console output showing the failing START command.

Related tasks:

➡ Creating resource limit tables (DB2 Performance)

➡ Setting limits for system resource usage by using the resource limit facility (DB2 Performance)

Related reference:

➡ Resource limit facility tables (DB2 Performance)

➡ -START RLIMIT (DB2) (DB2 Commands)

DSNT721I   *csect-name* **DATABASE** *name* **CANNOT BE STARTED WITH ACCESS (UT). IT CONTAINS AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Database *name* cannot be started with access (UT) because it contains an active resource limit specification table.

System action: The START DATABASE command is ignored.

Operator response: If the database must be started with access (UT), either stop the resource limit facility (STOP RLIMIT), or switch it to use a resource limit specification table that is not in the database (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a resource limit specification table that is not in the database.

DSNT722I   *csect-name* **TABLE SPACE** *name* **CANNOT BE STARTED WITH ACCESS (UT). IT CONTAINS AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Table space *name* cannot be started with access (UT) because it contains an active resource limit specification table.

System action: The START DATABASE command is ignored.

Operator response: If the table space must be started with access (UT), either stop the resource limit facility (STOP RLIMIT), or switch it to use a resource limit specification table that is not in the table space (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a resource limit specification table that is not in the table space.

DSNT723I   *csect-name* **INDEX SPACE** *name* **CANNOT BE STARTED WITH ACCESS (UT). IT IS DEFINED ON AN ACTIVE RESOURCE LIMIT SPECIFICATION TABLE**

Explanation: Index space *name* cannot be started with access (UT) because it contains an index on an active resource limit specification table.

System action: The START DATABASE command is ignored.

Operator response: If the index space must be started with access (UT), either stop the resource limit facility (STOP RLIMIT), or switch it to use a different resource limit specification table that does not have an index in the index space (START RLIMIT,ID=xx). In a DB2 data sharing environment, you must either stop the resource limit facility on all members of the DB2 data sharing group or switch it to use a resource limit specification table that does not have an index in the index space.

DSNT724E   *csect-name* **ERROR** *reason-code* **IN READING RESOURCE LIMIT SPECIFICATION TABLE** *table-name*. **FOR AUTHORIZATION ID** *auth-id* **RUNNING THREAD WITH CORRELATION ID** *correlation-id*, **CONNECTION ID** *connection-id*, **AND LUWID** *luw-id*, **USE OF** *operation-name* **IS NOT GOVERNED.**

Explanation: An error occurred reading resource limit specification table *table-name*. *auth-id* is assumed to be allowed to perform *operation-name*. *operation-name* is BIND or PARALLEL PROCESSING.

The reason code defining the read error is included in the message text as well as the connection ID, correlation ID, and logical unit of work (LUW) ID of the thread. An LUW-ID with an asterisk (*) indicates that the request originated at this site. If not an asterisk (*), the LUW-ID is assigned to the allied thread at the requesting site. Correlation IDs beginning with '0nn', where 'nn' is a number from 01 to 28, identify system agents.

System action: If the *operation-name* is BIND, processing continues if the *auth-id* is authorized to perform the BIND. If the *operation-name* is PARALLEL PROCESSING, query parallelism is not disabled and can be used if the value of

the CURRENT DEGREE special register is 'ANY'. If the *operation-name* is PREDGOV, the active RLF (Resource Limit Facility) assumes that the RLST table does not contain a row for user-predicted ASUTIME, and the execution of the SQL statement continues as normal.

Operator response: Notify the system programmer.

System programmer response: Investigate the problem indicated by the reason code and preceding messages. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNT725E   **AN ERROR OCCURRED WHILE GOVERNING BIND OPERATIONS OF auth-id RUNNING WITH CORRELATION-ID correlation-id, CONNECTION-ID connection-id, LUW-ID luw-id. auth-id IS ALLOWED TO PERFORM BIND OPERATIONS**

Explanation: An internal error occurred while determining whether or not 'auth-id' is allowed to perform bind operations. It is assumed that 'auth-id' is allowed to perform bind operations.

An LUW-ID '*' indicates that the request originated at this site. If not an '*', the LUW-ID is assigned to the allied thread at the requesting site. Correlation IDs beginning with '0nn' where 'nn' is a number from 01 to 28, identify system agents.

This message is issued by the following module: DSNTCAUT

System action: Processing occurs assuming the authorization ID is allowed to perform bind operations.

Operator response: Notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNT726I   *csect-name* **DATABASE** *name* **AND ANY SPACE IT CONTAINS CANNOT BE STARTED WITH ACCESS (UT). IT IS THE ACTIVE COMMUNICATIONS DATABASE**

Explanation: The database specified in the message cannot be started with access (UT) because it is the active communications database. In addition, spaces within the active communications database cannot be started with access (UT).

System action: The START DATABASE command is ignored.

Operator response: If the database or any space within it must be started with access (UT), you must stop the distributed data facility (STOP DDF). In a DB2 data sharing environment, you must stop the distributed data facility on all members of the DB2 data sharing group.

DSNT727I   TABLE *table-name* **WILL NOT BE USED BY THE RESOURCE LIMIT FACILITY**

Explanation: This message is a normal response for a START RLIMIT command. Table *table-name* cannot be used to determine resource limits because table *table-name* does not exist.

System action: The START RLIMIT command will use the existing resource limit specification table. The name of that table is in a DSNT704I message.

System programmer response: To use *table-name* to determine resource limits, create *table-name*, and issue the -START RLIMIT command again.

DSNT728I   THE FORMAT OF TABLE *authid.table-name* **IS NOT SUPPORTED IN CONVERSION MODE. THE START RLIMIT COMMAND CANNOT USE THIS TABLE**

Explanation: An attempt was made to start the resource limit facility with a table which has column definitions not supported in conversion mode.

authid   The authorization ID which created the table.

table-name
        The name of the table.

System action: The resource limit facility is not started with *authid.table-name*.

System programmer response: Correct the command to specify a table with the proper column definitions.

Related tasks:

➡ Converting resource limit facility (RLF) tables (optional) (DB2 Installation and Migration)

Related reference:

➡ DSNRLMTxx resource limit tables (DB2 Performance)

DSNT735I   object name CANNOT BE STARTED. IT CONTAINS OR IS DEFINED ON THE DDL REGISTRATION TABLE(S)

Explanation: The object specified in the message cannot be started because it contains or is defined on the DDL registration table(s). Installation SYSADM or installation SYSOPR authority is needed to start the object.

In the message, 'object' is either:
- DATABASE if the specified database contains the application and object registration tables
- TABLESPACE if the specified table space contains an application or object registration table
- INDEXSPACE if the specified index space is defined on an application or object registration table

System action: The object specified in the message is not started.

User response: Reissue the command with the proper authority.

DSNT736I   *csect-name* ASYNCHRONOUS STOP DATABASE COMMAND HAS COMPLETED FOR COMMAND: *command_string*

Explanation: Asynchronous processing for the STOP DATABASE command completed for the command whose syntax is shown in the message.

command_string is the STOP DATABASE command string for the command that is completed. The string always uses the abbreviation DB for DATABASE and SPACE for SPACENAM to save characters.

Only the first 100 characters of the STOP DATABASE command are displayed.

System action: Asynchronous processing for the STOP DATABASE command is complete.

System programmer response: No action is required.

DSNT740I   *csect-name start-or-stop* **PROFILE IS ALREADY IN-PROGRESS**

Explanation: You issued either a START or STOP PROFILE command, but this command has been issued before, and is still in-progress. The *start-or-stop* indicates which command was issued.

This message is generated from module *module-name*. When you issued either a START PROFILE or STOP PROFILE command, the command failed to acquire a conditional lock.

System action: The START or STOP PROFILE command is ignored.

System programmer response: Wait for the command to complete or stop. If necessary, resubmit the command.

DSNT741I   *csect-name: start-or-stop* **PROFILE IS COMPLETED**

Explanation: You issued either a START or STOP PROFILE command, and the command has completed successfully. This message is the normal response to a START or STOP PROFILE command.

System action: The START PROFILE or STOP PROFILE command is completed.

DSNT742I   *csect-name* **PROFILE IS ALREADY STOPPED. STOP PROFILE COMMAND IS IGNORED**

Explanation: When you issued a STOP PROFILE command, the profile was not active or had already been stopped.

System action: The STOP PROFILE command is ignored.

DSNT743I   *csect-name* **TABLE** *creator-name.table-name* **DOES NOT EXIST. THE START PROFILE COMMAND WILL NOT USE THIS TABLE**

Explanation: You issued a START PROFILE command to activate the profile function, but the table *creator-name.table-name* involved in this process does not exist. The reported table is an optional table and the START PROFILE command will not use it. The information intended to be reported in this table will not be available.

System action: The START PROFILE command continues, and the offending table will not be used.

System programmer response: Verify that all the tables exist, and then run the command again. You can ignore this message if the table is an optional table and continue the profile function.

DSNT744I   *csect-name* **TABLE** *creator-name.table-name* **DOES NOT EXIST. THE START PROFILE COMMAND FAILED.**

Explanation: You issued a START PROFILE command to activate the profile function, but the table *creator-name.table-name* involved in this process does not exist. The START PROFILE command failed.

System action: The START PROFILE command failed.

System programmer response: Verify that all of the tables exist and run the command again.

DSNT745I   *csect-name* **INDEX** *creator-name.index-name error-type*. **THE START PROFILE COMMAND FAILED**

Explanation: You issued a START PROFILE command to activate the profile function, but the index *creator-name.index-name* has an error, as indicated by *error-type*. The error can be one of the following.
- The index does not exist.
- The index is not unique.
- The index is not defined correctly.

System action: The START PROFILE command failed.

System programmer response: Verify that the table is created correctly, with the proper index, and then run the command again.

DSNT746I   *csect-name* **INDEX** *creator-name.index-name* **COLUMN** *key-column-name* **NEEDS TO BE** *asc-or-desc*. **THE START PROFILE COMMAND FAILED.**

Explanation: You have issued a START PROFILE command to activate the profile, but the index *creator-name.index-name* on column *key-column-name* needs to be in *asc-or-desc* order. *asc-or-desc* indicates the ordering, and can either be ASC or DESC.

System action: The START PROFILE command failed.

System programmer response: Verify that the table is created correctly, with the proper index, and run the command again.

DSNT747I   *csect-name* **INDEX** *creator-name.index-name* **IS NOT DEFINED FOR TABLE** *creator-name.table-name*. **THE START PROFILE COMMAND FAILED**

Explanation: You have issued a START PROFILE command to activate the profile function, but the index *creator-name.index-name* was not created for table *creator-name.table-name*.

System action: The START PROFILE command failed.

System programmer response: Verify that the table is created correctly, with the proper index, and then run the command again.

DSNT748I   *csect-name* **THE COLUMN** *column-name* **IN TABLE** *creator-name.table-name* **IS NOT DEFINED PROPERLY. THE COLUMN DEFINITION IS** *data-type* **WITH LENGTH** *length* **AND NULLABLE ATTRIBUTE** *attr*. **THE COLUMN DEFINITION SHOULD BE** *data-type* **WITH LENGTH** *length* **AND NULLABLE ATTRIBUTE** *attr*. **THE START PROFILE COMMAND WILL NOT USE THIS TABLE**

Explanation: You issued a START PROFILE command to activate the profile function, but the column *column-name* in the table *creator-name.table-name* involved in this process is improperly defined. The reported table is an optional table and the START PROFILE command will not use it. The information intended to be reported in this table will not be available.

System action: The START PROFILE command continues and the offending table will not be used.

System programmer response: Verify that the table is created correctly with the proper column, and then run the command again. You can ignore this message if the table is an optional table, and continue the profile function.

DSNT749I   *csect-name* **THE COLUMN** *column-name* **IN TABLE** *creator-name.table-name* **IS NOT DEFINED PROPERLY. THE COLUMN DEFINITION IS** *data-type* **WITH LENGTH** *length* **AND NULL ATTRIBUTE** *attr*. **THE COLUMN DEFINITION SHOULD BE** *data-type* **WITH LENGTH** *length* **AND NULL ATTRIBUTE** *attr*. **THE START PROFILE COMMAND FAILED**

Explanation: You issued a START PROFILE command to activate the profile function, but the column *column-name* in the table *creator-name.table-name* involved in the process is improperly defined.

System action: The START PROFILE command failed.

System programmer response: Verify that the table is created correctly, with the proper columns, and then run the command again.

DSNT750I   *csect-name* **THE COLUMN** *column-name* **IS MISSING IN TABLE** *creator-name.table-name*. **IT SHOULD BE DEFINED AS** *data-type* **WITH LENGTH** *length* **AND NULL ATTRIBUTE OF** *attr*. **THE START PROFILE COMMAND WILL NOT USE THIS TABLE.**

Explanation: You have issued a START PROFILE command to activate the profile function, but the column *column-name* in the table *creator-name.table-name* is missing. The reported table is an optional table, and will not be used by the START PROFILE command. The information that was intended to be reported in this table will not be available.

System action: The START PROFILE command continues and the offending table will not be used.

System programmer response: Make sure the table is created correctly with the proper columns. Then, run the command again. You can ignore this message if the table is an optional table and continue the profile function.

DSNT751I   *csect-name* **THE COLUMN** *column-name* **IS MISSING IN TABLE** *creator-name.table-name*. **IT SHOULD BE DEFINED AS** *data-type* **WITH LENGTH** *length* **AND NULL ATTRIBUTE OF** *attr*. **THE START PROFILE COMMAND FAILED**

Explanation: You have issued a START PROFILE command to activate the profile function, but the column *column-name* in the table *creator-name.table-name* is missing. The START PROFILE command failed.

System action: The START PROFILE command failed.

System programmer response: Make sure the table is created correctly with the proper columns. Then, run the command again.

DSNT752I   *csect-name* **ERROR** *reason-code* **IN** *reading-or-writing* **TABLE** *creator-name.table-name*. **THE START PROFILE COMMAND FAILED.**

Explanation: An error occurred when DB2 was reading or writing the specified table.

reason-code
> The reason code returned from the read or write operation.

System action: The read or write operation is not completed. The START PROFILE command failed.

System programmer response: Investigate the problem indicated by the reason code. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNT753I   *csect-name* **DISPLAY PROFILE REPORT FOLLOWS:**

Explanation: This message displays output from the DISPLAY PROFILE command.

csect-name
> The name of the control section that issued the message.

The output from the command consists of one or more of the following sections, in the indicated order:

Current profiling status: The output contains information about the current profiling status:

STATUS=*profile-status*

STATUS *profile-status*
> The profiling status. Expected values are:
>
> **ON**   Profiling is active.
>
> **OFF**   Profiling is inactive.
>
> **SUSPENDED**
> > Profiling was active but is now suspended because of an error. For example, the maximum push-out limit was reached or the table space is full.
>
> **STARTING**
> > Profiling is being started but is not yet complete.
>
> **STOPPING**
> > Profiling is being stopped but is not yet complete.

Information about when profiling started: If the profile status is ON or SUSPENDED, the output contains information about when profiling started:

TIMESTAMP=*profile-timestamp*

TIMESTAMP *profile-timestamp*
> The date and time when the profiling status started. This information is displayed in the format*YYYY-MM-DD-HH.MN.SS.NNNNNN*.

Information about pushouts: The output contains information about reports that were issued for monitored SQL statements:

PUSHOUT=*current-pushouts* OUT OF *maximum-pushouts*

PUSHOUT *current-pushouts*
> The current number of reports that were issued for all monitored SQL statements since the profile was started. This value is applicable only for profiles that monitor SQL statements. This value is reported only when the profile status is ON or SUSPENDED.

OUT OF *maximum-pushouts*
> The maximum number of total reports that are allowed for all (normal and exception) statement monitoring during the lifetime of a monitor profile.

End of output: The output normally ends with the following message:

```
DISPLAY PROFILE REPORT COMPLETE
```

System action: Processing continues.

System programmer response: Take the action that is indicated for the *profile-status* value:

ON No action is required. To stop profiling, issue STOP PROFILE.

OFF No action is required. To start profiling, issue START PROFILE.

STARTING or STOPPING
> Wait until the command execution is complete.

SUSPENDED
> Correct the problem that is causing the suspension of profiling activities, then stop or restart profiling.

Related concepts:

▶ Profiles (DB2 Performance)

Related reference:

▶ -DISPLAY PROFILE (DB2) (DB2 Commands)

▶ -START PROFILE (DB2) (DB2 Commands)

▶ STOP PROFILE (DB2) (DB2 Commands)

DSNT758I *csect-name* **ATTEMPT TO USE NEW COMMAND OR OPTION** *new-command* **BEFORE NEW FUNCTION MODE**

Explanation: A new command or command option was issued before new-function mode was enabled.

new-command
> The new command or command option that can be used only after new-function mode is enabled.

System action: The new command or command option is not processed.

System programmer response: To use the new command or command option, enable new-function mode.

Related reference:

▶ Command changes in Version 11 (DB2 for z/OS What's New?)

DSNT759I **A DEPRECATED EXPLAIN TABLE OR PROFILE TABLE FORMAT FOR** *csect-name* **TABLE** *creator-name***.***table-name* **IS BEING USED.**

Explanation: The EXPLAIN table or profile table uses a deprecated format.

csect-name
> The name of the control section that issued the message.

*creator-name***.***table-name*
> The table about which this message is issued.

System action: Processing continues.

User response: Update the definition of the EXPLAIN table or profile table. Either redefine the required table by dropping it and re-creating it, or in the case of EXPLAIN tables, migrate the tables to the current format.

Related tasks:

⏩ Creating EXPLAIN tables (DB2 Performance)

Related reference:

⏩ Statements (DB2 SQL)

DSNT760I   *csect-name* **TABLE** *creator-name.table-name* **MUST BE ENCODED IN UNICODE. IT CANNOT BE IN ASCII OR EBCDIC.**

Explanation: The EXPLAIN table is encoded as either ASCII or EBCIDIC. The table must be encoded in Unicode.

csect-name
> The name of the control section that issued the message.

creator-name.table-name
> The table about which this message is issued.

System action: The statement cannot be processed. The explanation information is not generated.

User response: Update the definition of the EXPLAIN table. Either redefine the required EXPLAIN table by dropping it and re-creating it, or migrate EXPLAIN tables to the current format.

Related tasks:

⏩ Creating EXPLAIN tables (DB2 Performance)

Related reference:

⏩ Statements (DB2 SQL)

DSNT771I   *csect-name* **A MONITOR PROFILE** *condition-type* **CONDITION OCCURRED** *number* **TIME(S).**

Explanation: A monitor profile has detected that a warning or exception condition occurred.

csect-name
> The name of the control section that issued the message.

condition-type
> The type of profile threshold condition that was exceeded:
>
> **EXCEPTION**
>> A profile exception threshold was exceeded.
>
> **WARNING**
>> A profile warning threshold was exceeded.

number   The number of times that any monitor profile warning or exception condition has occurred since the most recent DSNT771I message was issued, or since the START PROFILE command was issued.

System action: This message is issued at a minimum interval of five minutes.

In addition to issuing this message, the system writes a statistics class 4 IFCID 402 trace record for any profile warning or exception condition that occurs in a statistics interval. The statistics trace record provides counters that track the number of times a given profile threshold in a specific PROFILE ID was exceeded. See IFCID 402 for more information.

Operator response: Notify the system programmer.

System programmer response:

1. Determine the destination of statistics class 4 and locate and analyze the trace record.
2. If possible, use the statistics trace record to identify which profile threshold in a given profile ID detected the warning or exception condition.
3. Determine if the profile threshold is sufficient. If the profile threshold is set too low, increase the profile threshold value in the DSN_PROFILE_ATTRIBUTES table, then refresh the threshold by issuing START PROFILE command.
4. **Optional:** Stop the profile monitoring processing by performing either of the following tasks:

Disable monitoring for the profile

To disable the monitoring function for a specific profile:

a. Locate the appropriate row in SYSIBM.DSN_PROFILE_TABLE and either delete the row or change the PROFILE_ENABLED column value to N.

b. Use the START PROFILE command to reload the profile table

Disable all monitoring

To disable all monitoring for both system related and statement related monitoring events that have been specified in the profile table, issue the STOP PROFILE command.

5. **Optional:** Modify the profile to specify a different diagnostic level to provide more detail message information. In the DSN_PROFILE_ATTRIBUTES table, set the ATTRIBUTE1 column setting to either WARNING_DIAGLEVEL2 or EXCEPTION_DIAGLEVEL2.

Problem determination: Analyze the statistics class 4 trace record IFCID 402.

Related concepts:

▶ Profiles (DB2 Performance)

Related reference:

▶ -START PROFILE (DB2) (DB2 Commands)

▶ STOP PROFILE (DB2) (DB2 Commands)

▶ SYSIBM.DSN_PROFILE_ATTRIBUTES (DB2 Performance)

DSNT772I   *csect-name* **A MONITOR PROFILE** *condition-type* **CONDITION OCCURRED** *number* **TIME(S) IN PROFILE ID=***profile-id* **WITH PROFILE FILTERING SCOPE=***filtering-scope* **WITH REASON=***reason-code***.**

Explanation: A monitor profile has detected that a warning or exception condition occurred.

csect-name
The name of the control section that issued the message.

condition-type
The type of profile threshold condition that was exceeded:

EXCEPTION
A profile exception threshold was exceeded.

WARNING
A profile warning threshold was exceeded.

number   The number of times that any monitor profile warning or exception condition has occurred since the most recent DSNT772I message was issued, or since the START PROFILE command was issued.

profile-id
The profile ID of the monitor profile (defined in DSN_PROFILE_TABLE) that detected the condition.

filtering-scope
The profile filtering scope that is associated with the profile ID that is defined in DSN_PROFILE_TABLE. Expected values:

IPADDR
The profile filtering scope is defined for an IP address.

PRDID   The profile filtering scope is defined for a product ID.

ROLE_AUTHID
The profile filtering scope is defined for a role and an authorization ID.

ROLE   The profile filtering scope is defined for a role.

AUTHID
The profile filtering scope is defined for an authorization ID.

COLLID_PKGNAME
The profile filtering scope is defined for a collection ID and a package name.

DSNT800I

DSNT800I   *csect-name* **DB2 CANNOT IDENTIFY TO IRLM. IRLM CANNOT CONNECT TO ITS DATA SHARING GROUP. THE LOCK STRUCTURE THAT IRLM ATTEMPTED TO ACCESS WAS** *strname*.

Explanation: IRLM failed to connect to the IRLM data sharing group. This message accompanies abend reason code 00E30806.

System action: DB2 terminates abnormally with reason code 00E30806.

System programmer response: Some possible reasons why IRLM could not connect to its data sharing group are:

- The lock structure is not properly defined in the active MVS CFRM administrative policy. The lock structure name is a concatenation of the DB2 data sharing group name and the string 'LOCK1' separated by an underscore (_). For example, if the DB2 data sharing group name is DSNCAT, DSNCAT_LOCK1 is the lock structure name.

  Ensure that the lock structure is properly defined in the active CFRM policy. The name of the lock structure that IRLM attempted to access is given in this message. IRLM issues message DXR138E to indicate that the lock structure definition could not be accessed in the active CFRM policy.

- IRLM could not connect the lock structure. The reason for the connect failure is given in IRLM message DXR135E.

- IRLM could not join its data sharing group. The reason for the join failure is given in IRLM message DXR134E.

- IRLM joined the wrong group. The set of IRLMs that are connected to a given lock structure must all be members of the same group. Ensure that the GROUP parameter is correctly specified in the IRLM startup procedure. If you give an incorrect group name, IRLM connects the lock structure, and then times out waiting for responses from other connectors. IRLM issues message DXR133I for the timeout.

DSNT801I   ACTION A (ADD) DOES NOT ALLOW GENERIC DISPLAY (% IN DATA FIELD).

Explanation: You have to specify a specific data item in the DATA field.

This message is issued by the following panel: DSN8SSH

System action: The panel is redisplayed, and the cursor is placed on the field in error.

User response: Enter specific data in the DATA field.

DSNT802I   ONLY ACTION D (DISPLAY) IS ALLOWED FOR DEPARTMENT STRUCTURE.

Explanation: You cannot add, update, or erase a department structure. You can only display it.

This message is issued by the following panel: DSN8SSH

System action: The panel is redisplayed, and the cursor is placed on the field in error.

User response: Change the ACTION field to D, or change the OBJECT field to DE or EM.

DSNT803I   *csect-name* DB2 CANNOT IDENTIFY TO IRLM. THE REQUESTED FUNCTION LEVEL IS NOT SUPPORTED.

Explanation: An attempt was made to identify to an IRLM that does not support the level of function DB2 needs. This message accompanies abend reason code 00E30800.

System action: DB2 terminates abnormally with reason code 00E30800.

System programmer response: Ensure that the STEPLIB statement in your IRLM startup procedure is referencing the correct IRLM code library and that IRLM is at the appropriate level of maintenance.

DSNT804I   *csect-name* THERE ARE MODIFY LOCKS OWNED BY THIS DB2 THAT HAVE BEEN RETAINED

Explanation: When DB2 disconnected from IRLM during DB2 shutdown, IRLM indicated that there were still outstanding modify locks. These locks were retained by IRLM. The retained locks continue to protect the resources on which they are held from requests from other DB2s in the data sharing group. That is, IRLM rejects any lock request from another DB2 that is incompatible with the retained state of the lock. The locks are retained until the terminating DB2 subsystem is restarted and the resources on which the locks are held are recovered.

System action: DB2 continues with its shutdown processing.

System programmer response: No response is necessary.

DSNT805I   NAME TYPE PART STATUS LPL PAGES

Explanation: This message indicates the beginning of a set of messages used to display information as a result of a DISPLAY DATABASE command when the keyword LPL is specified. This message contains the headings of the display. (The message appears in one continuous line across the screen, not as shown above.) This message is always followed by one or more DSNT806I messages containing the column data. Refer to message DSNT806I for a description of the data these headings identify.

DSNT806I

DSNT806I   *space type part status LPL pages*

Explanation: This message contains the display information resulting from a DISPLAY DATABASE command when the keyword LPL is specified. It contains the column data describing the status of a table space or index space in the database listed in the preceding DSNT362I message. This message is always preceded by its column headings message DSNT805I. The message number for this message is not displayed.

The message fields have the following meanings:

space   The table space name or index space name.

type   TS for a table space, IX for an index space, LS for a LOB table space.

part   The partition number. It is blank for a nonpartitioned table space or index space. For nonpartitioned secondary indexes, it is the logical partition number preceded by the character L (for example, L0001). For data-partitioned secondary indexes, the prefix for this value will be the character D (for example, D0001).

status   One or more of the following, delimited by commas. The maximum length is 18 characters:

   ACHKP
           The table space is in aux check pending state. ACHKP on a table space indicates that CHECK DATA detected an orphan or missing LOB or an out-of-sync condition between the row in the base table and the LOB in the LOB table space.

   AUXW   The table space is in aux warning state. AUXW on a LOB table space indicates that one or more LOBs in the LOB table space is invalid. AUXW on a base table space indicates that CHECK DATA detected an orphan or missing LOB or an out-of-sync condition between the row in the base table and the LOB in the LOB table space.

   CHKP   The table space is in check pending status.

   COPY   An image copy is required for this object (table space or data set within this table space). STOP is also displayed. If the image copy is required because DB2 encountered a problem during RESTART, STOPE is displayed instead of STOP.

   GRECP
           The object (a table space, a table space partition, index space, or index space partition) is marked recovery pending because of the failure of a group buffer pool. This status value appears only on spaces. It does not apply to databases.

   LPL   The object has pages or ranges of pages that are unavailable because of logical or physical damage.

   LSTOP   The logical partition is stopped.

   PRO   The table space partition is in Persistent Read Only restricted status. Read access to the table space partition is allowed, but updates are prohibited.

   RECP   The table space or index space is in recovery pending status. The member must be restarted to restore full access to the data. For logical partitions, the RECP status can appear as RBDP*.

   RBDP*   Indicates that the entire index is inaccessible to SQL applications. However, only the logical partition needs to be recovered.

   RO   The database, table space, or index space is started for read activity only.

   RW   The database, table space, or index space is started for read and write activity.

   STOP   The database, table space, or index space is stopped.

   STOPE   DB2 implicitly stopped the table space or index space because of an internal inconsistency. A DSNT500I or DSNT501I message is issued at time of error detection to indicate the inconsistency.

   STOPP   A stop is pending or is in process for this database, table space, or index space.

   UT   The database, table space, or index space is started for utility processing only.

   UTRO   Although the table space or index space is started as RW, a utility is in process and only RO access is allowed.

   UTRW   The table space or index space is started as RW and a utility is in process.

UTUT Although the table space or index space is started as RW, a utility is in process and only UT access is allowed.

LPL pages
The pages in the specified *space* that are unavailable because of logical or physical damage.

Related concepts:

Resetting the REBUILD-pending status (DB2 Utilities)

Related tasks:

Displaying the logical page list (DB2 Administration Guide)

Resolving media failure (DB2 Utilities)

Related reference:

Advisory or restrictive states (DB2 Utilities)

Data sharing status descriptions (DB2 Data Sharing Planning and Administration)

-DISPLAY DATABASE (DB2) (DB2 Commands)

Related information:
"DSNT362I" on page 508
"DSNT805I" on page 589
"DSNT500I" on page 555
"DSNT501I" on page 556

DSNT807I *csect-name* **IRLM PURGE OF RETAINED LOCKS FAILED. RETURN CODE =** *retcode* **SUB-CODE =** *subcode*

Explanation: DB2 issued a PURGE request to IRLM to purge retained locks for this DB2. The PURGE request did not complete successfully. This message shows the return code and subcode from IRLM.

System action: DB2 abends with reason code 00E30808. Restart processing is terminated.

System programmer response: Refer to the IRLM codes.

DSNT809E *csect-name* **FEEDBACK FROM IRLM IDENTIFY INDICATED IRLM DID NOT CONNECT TO THE LOCK STRUCTURE**

Explanation: DB2 identified to IRLM, but cannot continue because IRLM did not connect to the lock structure. Global locking is ignored. This message accompanies abend reason code 00E30809.

System action: DB2 abnormally terminates with reason code 00E30809.

System programmer response: This condition happens if you are running DB2 in data sharing mode, but the associated IRLM is running with local scope (SCOPE=LOCAL in the IRLM startup procedure). If you are running DB2 in data sharing mode, ensure that the associated IRLM startup procedure specifies SCOPE=GLOBAL. Conversely, if you are not running DB2 in data sharing mode, ensure that the associated IRLM startup procedure specifies SCOPE=LOCAL.

DSNT810E *csect-name* **DB2 CANNOT IDENTIFY TO IRLM. AN INVALID LOCK STRUCTURE NAME WAS DETECTED.**

Explanation: DB2 could not identify to IRLM because of an invalid lock structure name. This message accompanies abend reason code 00E30807.

System action: DB2 startup terminates abnormally with reason code 00E30807.

System programmer response: This condition happens if you are not running DB2 in data sharing mode, but the associated IRLM is running with global scope (SCOPE=GLOBAL in the IRLM startup procedure). If you are not running DB2 in data sharing mode, ensure that the associated IRLM startup procedure specifies SCOPE=LOCAL. Conversely, if you are running DB2 in data sharing mode, ensure that the associated IRLM startup procedure specifies SCOPE=GLOBAL.

DSNT811E   *csect-name* **DB2 CANNOT IDENTIFY TO IRLM. REQUEST VIOLATES ONE DB2 PER IRLM RULE.**

Explanation: DB2 could not identify to IRLM because another DBMS is already identified to the IRLM. This message accompanies abend reason code 00E30802.

System action: DB2 abends with reason code 00E30802.

System programmer response: IRLM Version 2.1 enforces a one to one (1:1) mapping between IRLM and DB2. Multiple DBMS subsystems can no longer use the same IRLM. Ensure that:

- Each DB2 subsystem is associated with its own IRLM subsystem.
- The correct IRLM procedure name and subsystem name are specified in the DB2 system parameters.

DSNT812I   **-MEMBER NAME** *member-name*

Explanation: This message is issued with messages DSNT389I, DSNT394I or DSNT396I for each thread in a data sharing environment. This additional line reports the member name of the DB2 subsystem accessing this object.

If the characters '(CO)' follow the member name, then this member is the castout owner for the page set or partition. The castout owner for a page set or partition is the DB2 member responsible for doing all of the castout I/O for that page set or partition.

The message number and subsystem recognition character are not included in this message.

DSNT813I   **----** *status(es) page(s)*

Explanation: This message follows message DSNT389I, DSNT392I, DSNT394I, DSNT396I or DSNT806I to display more statuses, or more pages that are in the logical page list (LPL).

LPL pages can be in two formats, single pages (represented by the page number) or a range of pages (represented by the starting page number followed by a dash and the ending page number in the range).

The message number and subsystem recognition character are not included in this message because this message is part of a DISPLAY DATABASE data line.

DSNT814I   **NON-LOCAL INFORMATION UNAVAILABLE FOR** *object* = *name*

Explanation: One member of the DB2 data sharing group was unable to obtain information from another member for one of these reasons:

- Overflow of the maximum buffer area sent to the DB2 subsystem running the DISPLAY. This buffer area contains the DISPLAY information.
- Problems occurred while requesting DISPLAY DB information from peer DB2 subsystems.

Terms pertinent to this message:

object   Either DATABASE or SPACE

name   The name of the DATABASE or SPACE

User response:

- Issue DISPLAY GROUP to see if any DB2 members are down or unresponsive and then reissue the DISPLAY DATABASE command.
- Specify individual database names or space names instead of specifying '*'.

DSNT815I   **THRU** *part*

Explanation: This message follows message DSNT392I to indicate the last in a range of partitions that share the same STATUS information as the partition indicated in the preceding message DSNT392I. This message can only occur when the preceding message DSNT392I does not contain information for the following values:

- *to-page*
- *hi-page*
- *cat-name*
- *piece*

Refer to message DSNT392I for a description of these fields.

The message number and subsystem recognition character are not included in this message because this message is part of a DISPLAY DATABASE data line.

DSNT816I   *csect-name* **THE REAL-TIME STATISTICS COLLECTION PROCESS IS STOPPED**

Explanation: The statistics database stopped successfully, and the real-time statistics process is halted.

System action: DB2 will not externalize in-memory statistics.

Severity: 0 (informational)

DSNT818I   *csect-name* **INVALID SPACENAME SPECIFIED**

Explanation: Incorrect syntax is specified for the table space name while issuing START/STOP/DISPLAY Database Commands.

System action: DB2 will not process the command.

Severity: 0 (informational)

DSNT820I   **DB2 IS STARTING IRLM SUBSYSTEM=***irlm-ssid*

Explanation: Automatic restart of DB2 is starting the IRLM subsystem with the name that was specified by the IRLMSID subsystem parameter in the DSNZPARM module that was used for DB2 startup.

irlm-ssid
        The name of the IRLM subsystem that is started.

System action: The DB2 restart process start starts the specified IRLM subsystem.

Operator response: None.

System programmer response: None.

User response: None.

Problem determination: Check the IRLMNM parameter value that is specified in the IRLM startup procedure. The IRLM startup procedure is specified by the IRLMPRC subsystem parameter. The IRLMNM parameter value must be the same as the IRLMSID parameter value in the DSNZPARM module that was used at startup. If DB2 automatic restart fails, refer to message DSNT384I for more information on diagnosing the problem.

Related reference:

PROC NAME field (IRLMPRC subsystem parameter) (DB2 Installation and Migration)

SUBSYSTEM NAME field (IRLMSID subsystem parameter) (DB2 Installation and Migration)

Related information:
"DSNT384I" on page 518

Chapter 16. DSNU messages

Message identifiers that begin with DSNU communicate information about DB2 utilities.

The severity values shown for the utility messages that follow are returned as the job-step condition code from the job step during which the message is issued. If additional messages having higher severity values are issued during the same job step, the higher value is reflected as the job-step condition code.

The output of DSNU messages that are sent to SYSPRINT will have a timestamp after the message number. This timestamp uses the Julian date format of HH:MM:SS:TT.

Severity values have the following meanings:

0	Informational	No error occurred
4	Warning	A condition was detected. The user might need to take further action.
8	Error	An error was detected. Processing could not continue.
12	Severe error	An authorization error was detected. Processing could not continue.

DSNU000I   *csect-name* **OUTPUT START FOR UTILITY, UTILID =** *uuuuuuu*

Explanation: This message marks the start of output for the given utility job. The utility ID used to identify the utility job is 'uuuuuuu'.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU001I   *csect-name*- **RELEASE DEPENDENT MODULE** *module-name* **WAS NOT FOUND**

Explanation: The system could not find the specified module in the indicated library (private library, job library, or link library).

System action: The utility job is abended.

System programmer response: The last three characters of the module name indicate the DB2 release level of the subsystem identified in the parameters passed to DSNUTILB. However, LOAD could not find the release dependent utility module for that release in the indicated library. This is probably a maintenance problem.

User response: Ensure that the load module library is indicated correctly and that the indicated library contains the requested program. If the indicated library does not contain the requested program, contact the system programmer.

DSNU002I   *csect-name* **- COULD NOT LOAD RELEASE DEPENDENT MODULE** *module-name* **CODE=code REASON=reason**

Explanation: During the load of 'module-name', MVS returned the indicated system completion code and reason.

System action: The utility job is abended.

User response: Refer to the appropriate MVS publication.

DSNU003I   *csect-name* **NOT INVOKED APF AUTHORIZED**

Explanation: The stored procedure was not invoked as APF-authorized.

csect-name
> The name of the control section that issued the message.

System action: The utility is not run by the stored procedure.

System programmer response: Check the following items:
- The DSNUTILS or DSNUTILU entry in SYSIBM.SYSROUTINES specifies that the stored procedure runs in a WLM-established address space (WLM_ENVIRONMENT is not blank).
- The DSNUTILS or DSNUTILU load module is link-edited with AC(1).
- The DSNUTILS or DSNUTILU load module is in an APF-authorized library.
- All the libraries in STEPLIB JOBLIB are APF-authorized.

If you suspect an error in DB2, see the information about collecting data for DB2 diagnosis.

User response: Notify the system programmer.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.
- UTPRINT output, produced by utility invocation of SORT.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (for 04E and 04F abends). Availability of this dump is dependent on the failing environment.
- SYSABEND, SYSUDUMP, or SYSMDUMP output. Availability of these dumps is dependent on the failing environment.

Severity: 8 (error)

Related tasks:

➦ Collecting data (Collecting data)

Related reference:

➥ SYSIBM.SYSROUTINES table (DB2 SQL)

➥ DSNUTILS stored procedure (deprecated) (DB2 Utilities)

➥ DSNUTILU stored procedure (DB2 Utilities)

DSNU004I   *csect-name* **UTILITY** *utility-name* **NOT SUPPORTED**

Explanation: The specified utility is not supported by the DSNUTILS stored procedure.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility.

Only the following *utility-name* values are supported:

> ANY[1]
> CHECK DATA
> CHECK INDEX
> CHECK LOB
> COPY
> COPYTOCOPY

> DIAGNOSE
>
> LOAD
>
> MERGECOPY
>
> MODIFY RECOVERY
>
> MODIFY STATISTICS
>
> QUIESCE
>
> REBUILD INDEX
>
> RECOVER
>
> REORG INDEX
>
> REORG LOB
>
> REORG TABLESPACE
>
> REPAIR
>
> REPORT RECOVERY
>
> REPORT TABLESPACESET
>
> RUNSTATS INDEX
>
> RUNSTATS TABLESPACE
>
> STOSPACE
>
> UNLOAD

1. ANY indicates that TEMPLATE dynamic allocation is to be used. This value suppresses the dynamic allocation that is normally performed by the DSNUTILS stored procedure.

System action: The utility is not run by the stored procedure.

User response: Run the utility as a batch job.

Severity: 8 (error)

Related reference:

➡ DSNUTILS stored procedure (deprecated) (DB2 Utilities)

DSNU005I   *csect-name input-record*

Explanation: This message displays the utility statements that occur before and after a syntax error, as they were provided as input to the utility parser.

csect-name
> The name of the control section that issued the message.

input-record
> The utility statements.

An additional error message follows this message and identifies why the utility statement caused a parser error. A maximum of 500 DSNU005I messages are issued after syntax error. In the final DSNU005I instance, the message text is `DSNU005I STATEMENT ECHOING TERMINATED`.

System action: Utility processing is not initiated. No SYSIBM.SYSUTIL record is created.

User response: Correct the input error, and resubmit the job. Do not restart the job.

Severity: 8 (error)

DSNU006I   *csect-name* **DB2 SYSTEM** *system-name* **HAS NOT BEEN STARTED**

Explanation: Utility startup processing has been unable to connect to the indicated DB2 subsystem, because a -START DB2 command has not been issued for that subsystem.

System action: Utility processing is terminated without invoking the requested utility or utilities.

Operator response: Contact the system programmer to determine if a -START DB2 command should be issued for the indicated subsystem.

System programmer response: Have the operator issue a -START DB2 command for the indicated subsystem.

User response: Correct your JCL to specify the correct subsystem-name on the EXEC statement, and resubmit the job. Otherwise, contact the system programmer to determine when the subsystem will be available for processing.

Problem determination: DB2 must be started in order to execute DB2 utilities.

Severity: 8 (error)

DSNU007I   *csect-name* **DB2 SYSTEM** *system-name* **DOES NOT EXIST**

Explanation: Utility startup processing has been unable to connect to the indicated DB2 subsystem, because it does not exist on this system.

csect-name
      The name of the control section that issued the message.

system-name
      The name of the DB2 subsystem that does not exist.

System action: Utility processing is terminated without invoking the requested utility or utilities.

System programmer response: Ensure that the correct DB2 subsystem has been started with the -START DB2 command.

If you use RACF to protect connections to DB2 subsystems, you might have to create additional RACF profiles in the DSNR class. Otherwise, connections to some DB2 subsystems might fail. Use the RACF RDEFINE command to add RACF profiles in the DSNR class to protect all DB2 subsystems. For example:

```
RDEFINE DSNR DB81.BATCH UACC(CONTROL)
```

User response: Correct your JCL to specify the correct subsystem name on the EXEC statement, and resubmit the job. Contact your system programmer if you are not sure which DB2 subsystem you should specify.

Problem determination: DB2 must be started in order to execute DB2 utilities.

Severity: 8 (error)

Related reference:

⮕ -START DB2 (DB2) (DB2 Commands)

⮕ RDEFINE command (Security Server RACF Command Language Reference)

DSNU008I   *csect-name* **SPECIFIED USER FAILED VALIDITY CHECK**

Explanation: Utility startup processing has been unable to connect to DB2, because the user failed the connection validity check.

System action: Utility processing is terminated without invoking the requested utility or utilities.

System programmer response: Locally defined RACF or equivalent security system connection authorization have denied the user access to DB2. Permit DB2 connection authorization if this user should be allowed access.

User response: Contact the system programmer for assistance with your installation's security checking.

Severity: 8 (error)

DSNU009I   *csect-name* **SPECIFIED USER NOT AUTHORIZED FOR DB2 UTILITIES**

Explanation: Utility startup processing for the user indicated on the job statement failed, because the user is not authorized to execute DB2 utilities.

System action: Utility processing is terminated without invoking the requested utility or utilities.

System programmer response: If the USER specified something on the job statement that is incorrect, correct it, and resubmit the job. Use the SQL GRANT command to give the user the required authority.

User response: Contact your system programmer for assistance with your installation's security checking.

Severity: 8 (error)

DSNU010I *csect-name* **UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE =** *rr*

Explanation: The utility completed processing with a return code of 'rr'. This indicates that no terminating error was detected.

System action: In most cases, the SYSIBM.SYSUTIL record for this job step is deleted from SYSIBM.SYSUTIL. However, if the utility used was REORG and UNLOAD(PAUSE), the SYSIBM.SYSUTIL record remains intact so that REORG can be restarted at the RELOAD phase.

User response: If the return code ('rr') is 0, no response is required.

If the return code ('rr') is 4, the utility output should be examined for other messages.

Severity: 0 (informational) or 4 (warning)

DSNU011I *csect-name*- **THE CURRENT TIME ZONE OFFSET:** *offset-value* **EXCEEDS +-24 HOURS. THE ALLOWABLE RANGE IS -82397 TO +82397 UNITS. SEE SQLCODE -187.**

Explanation: A utility has detected an invalid z/OS time zone offset.

csect-name
> The name of the control section that issued the message.

offset-value
> The invalid time zone offset that was specified.

> DB2 requires the offset to be within the range -82397 to +82397, inclusive. Each unit represents 1.048576 seconds.

System action: The utility job either ends with return code 8 or abends with reason code 00E40340.

System programmer response: One or more of the following conditions exists on the z/OS system on which this error occurred:

- The system was configured at IPL with an invalid time zone offset. To correct the offset, update the CLOCK*xx* member of SYS1.PARMLIB.
- A SET DATE console command modified the date. Use the SET DATE command to correct the date.
- A SET CLOCK console command modified the time. Use the SET CLOCK command to correct the time.

User response: Contact the system programmer.

Severity: 8 (error)

Related information:

↪ -187 (DB2 Codes)

↪ 00E40340 (DB2 Codes)

DSNU012I *csect-name* **UTILITY EXECUTION TERMINATED, HIGHEST RETURN CODE =** *rr*

Explanation: The utility execution was terminated. An error was detected that would not allow processing to continue. The return code has a value of 8.

System action: Utility processing is terminated.

User response: Examine utility output for other error messages.

Severity: 8 (error)

DSNU013I *csect-name* **JOB WITH DUPLICATE UTILID FOUND IN SYSUTIL**

Explanation: An entry with the same qualifier was found in the SYSIBM.SYSUTIL table for a job that was previously terminated.

System action: The utility job is abended with reason code X'00E40018'.

User response: If you issued the job in order to re-run the utility, then use the -TERM utility command to delete the entry that has the same qualifier from the SYSIBM.SYSUTIL table. If the job has no relation to the SYSIBM.SYSUTIL entry job, then use a new qualifier.

DSNU014I   *csect-name* **JOB WITH DUPLICATE UTILID CURRENTLY EXECUTING**

Explanation: A job with the same qualifier was found to be executing.

System action: Utility processing is terminated.

User response: If the job with the same qualifier that is currently executing is correct, either wait for the job to complete before using the same qualifier, or use another qualifier. If the job currently executing is incorrect, the -TERM command can be used to terminate it, and the failing job can be resubmitted. Do not -TERM UTILITY a utility job step that is currently active in REORG beyond the UNLOAD phase.

DSNU015I   *csect-name* **JOB WITH UTILID NOT FOUND IN SYSUTIL FOR RESTART**

Explanation: A utility job was submitted to restart a utility job that had previously failed. The SYSIBM.SYSUTIL entry for the specified qualifier provided was not found.

System action: Utility processing is not initiated.

User response: Use the -DISPLAY utility command to determine the correct qualifier. The job can then be resubmitted.

Severity: 8 (error)

DSNU016I   *csect-name* **UTILITY BATCH MEMORY EXECUTION ABENDED, REASON=X0***rrr cause*

Explanation: The utility batch address space has abended. If X'0rrr' is four characters in length (that is, X'00C1'), the abend was a system X'rrr' abend. If X'0rrr' is four characters in length and system abend X'000' was indicated, a user abend was issued. If X'0rrr' is eight characters in length, the abend was a system X'04E' abend, and X'0rrr' is an abend reason code.

If the value of general register 2 is not zero, then *cause* will contain CAUSE=X'*cccccccc*' where X'*cccccccc*' is the value of general register 2. This information is for service purposes only.

System action: The utility job is abended. The SYSIBM.SYSUTIL entry for the job is retained in the SYSUTIL table.

System programmer response: If a system abend was encountered (4 characters), refer to the appropriate MVS publication. If an eight-character abend reason code was encountered, analyze the values in the SYSIBM.SYSUTIL directory table.

User response: Consult the system programmer to obtain information on failure analysis. If the cause of the failure is correctable, the utility job can be restarted. If the error cannot be corrected, use the -TERM UTILITY command to delete the failing job's utility ID from SYSIBM.SYSUTIL.

Related reference:

➥ -TERM UTILITY (DB2) (DB2 Commands)

➥ SYSIBM.SYSUTIL table (DB2 SQL)

➥ DB2 abend completion codes (X'04E' and X'04F') (DB2 Codes)

DSNU017I   *csect-name* **UTILITY DATA BASE SERVICES MEMORY EXECUTION ABENDED, REASON=X0***rrr*
                CAUSE=X0*rrr*

Explanation: The database services address space has abended. If X'0rrr' is four hexadecimal digits in length (for example, X'00C1'), the abend was an MVS X'rrr' abend. If X'0rrr' is eight hexadecimal digits in length, the abend was a DB2 subsystem X'04E' abend, and X'0rrr' is an abend reason code.

If the value of general register 2 is not zero, then *cause* will contain CAUSE=X'*cccccccc*' where X'*cccccccc*' is the value of general register 2. This information is for service purposes only.

System action: The utility job is abended. The entry for the job is retained in the SYSIBM.SYSUTIL table.

System programmer response: If an MVS system abend was encountered (4 hexadecimal digits), refer to the appropriate MVS publication. If an abend reason code was encountered (8 hexadecimal digits), look up the reason code. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Consult with the system programmer to obtain information about failure analysis. If the cause of the failure is correctable, the utility job can be restarted. If the error cannot be corrected, the -TERM command should be used to delete the failing job's utility id from SYSIBM.SYSUTIL.

Related concepts:

➡ DB2 reason codes (DB2 Codes)

Related reference:

➡ SYSIBM.SYSUTIL table (DB2 SQL)

➡ -TERM UTILITY (DB2) (DB2 Commands)

DSNU018I   *csect-name* **JOB IS NOT RESTARTABLE DUE TO -START ACCESS FORCE COMMAND**

Explanation: A utility job was submitted to restart a utility job that previously failed. Although the SYSIBM.SYSUTIL entry for the specified qualifier was found, the job is not restartable because of a -START DATABASE ACCESS FORCE command. The -START DATABASE ACCESS FORCE command deleted the utility-in-process state of one or more (but not all) target objects in the job.

System action: The utility job is abended with reason code X'00E40018'.

User response: If the specified qualifier is correct, use the -TERM UTILITY command to terminate the utility and resubmit the job as a nonrestartable job. If the -TERM UTILITY command fails, use the -START DATABASE ACCESS FORCE command to force ALL target objects (table spaces and indexes) in the job. When the last object is forced, the SYSIBM.SYSUTIL record is deleted. If the qualifier is incorrect, correct it and restart the job.

DSNU019I   *csect-name* **THIS UTILITY IS NOT ALLOWED AT A DB2 TRACKER SITE**

Explanation: When a DB2 subsystem serves as a tracker site, it cannot run utility *csect-name*. A DB2 subsystem is also a tracker site if the value of subsystem parameter TRKRSITE is YES. You may run only the following utilities at a tracker site DB2:

- REPORT
- RECOVER without the TORBA, TOLOGPOINT, or TOCOPY options
- DIAGNOSE

System action: The utility terminates.

System programmer response: If this DB2 subsystem does not need to serve as a remote site tracker system, remove its tracker site status before re-trying the utility. You may remove the tracker site status by stopping the DB2 subsystem, then restarting it with TRKRSITE = NO.

User response: Contact the system programmer.

Severity: 8 (error)

DSNU020I   *csect-name* **JOB STEP INCOMPLETE**

Explanation: The REORG utility was invoked with the UNLOAD PAUSE option. This causes the immediate suspension of the job after the unload phase completes. Subsequent utilities that are invoked in the same job step are not run.

csect-name
        The name of the control section that issued the message.

System action: Further utility processing is terminated.

User response: Resubmit the job with the RESTART(PHASE) parameter after the requested user action has been taken with the unload data set.

Problem determination: Check the syntax of the UNLOAD keyword utility control statement.

Severity: 4 (warning)

DSNU021I • DSNU022I

Related reference:

⏎ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

⏎ Syntax and options of the REORG INDEX control statement (DB2 Utilities)

DSNU021I   *utility-name* **UTILITY WITH UTILID =** *utilid* **WHICH ENDED IN PHASE =** *phase-name*, **IS ATTEMPTING TO RESTART (***restart-type***)**

Explanation: The utility is attempting to restart with a different restart method than the one that was specified.

utility-name
>   The name of the utility.

utilid   The utility ID.

phase-name
>   The utility phase in which the utility job ended.

restart-type

>   The restart method that is being used. *restart-type* can be one of the following values:

>   **PHASE**
>   >   The utility restarts from the beginning of the phase that was being processed.

>   **CURRENT**
>   >   The utility restarts from the last commit point that was taken during the execution of the utility phase.

The type of restart might be changed during restart execution depending on the utility and the phase where the original job ended.

System action: The system continues processing.

User response: No action is required.

Severity: 0

Related concepts:

⏎ Restart of an online utility (DB2 Utilities)

DSNU022I   *csect-name* **- A UTILITY RUNNING UNDER A STORED PROCEDURE REQUIRES A WLM ENVIRONMENT WITH NUMTCB=1. UTILID** *utility-ID* **IS RUNNING WITH NUMTCB =** *tcb-number*.

Explanation: When the utility stored procedure invoked a utility, the stored procedure was running in a stored procedure address space that was started with NUMTCB>1. The stored procedure address space in which the utility stored procedure runs must be started with NUMTCB=1.

csect-name
>   The name of the control section that issued the message.

utility-ID
>   The utility identifier or the UID parameter that was used in the utility job step.

tcb-number
>   The NUMTCB parameter of the job that started the WLM-managed stored procedure address space. The NUMTCB value represents the maximum number of task control blocks (TCBs) that can run concurrently in an address space that is initiated by the WLM environment.

System action: Execution of the utility is stopped.

System programmer response: Change the PROCLIB member that was used to start the WLM-managed stored procedure address space to specify NUMTCB=1. Then, refresh the WLM environment and run the utility again.

Severity: 8 (error)

Related tasks:

➡ Setting up a WLM application environment for stored procedures during installation (DB2 Installation and Migration)

Related reference:

➡ DB2-supplied stored procedures and user-defined functions (DB2 Installation and Migration)

➡ DSNUTILU stored procedure (DB2 Utilities)

DSNU023I   *csect-name* **- A WLM ENVIRONMENT FOR A UTILITIES STORED PROCEDURE USES A SYSIN DATA SET WITH RECFM=***recfm* **AND LRECL=***lrecl***. THOSE DATA SET CHARACTERISTICS ARE INCORRECT FOR UTILITIES.**

Explanation: The DSNUTILU stored procedure invoked a utility. Utilities that run under the DSNUTILU stored procedure require that the SYSIN data set is defined with a record format of V or VB, and a record length of 32708. A routine that ran in the stored procedure address space before DSNUTILU set the record format or record length to other values.

csect-name
> The name of the control section that issued the message.

recfm   The record format of the SYSIN data set.

lrecl   The record length of the SYSIN data set.

System action: The utility that DSNUTILU invoked does not run.

User response: Run the VARY REFRESH command to refresh the WLM environment. Then rerun the application that calls DSNUTILU.

To prevent the problem in the future, take one of the following actions:

- Use the WLM environment only for running DSNUTILU.
- Add a SYSIN DD statement like the following one to the WLM environment startup procedure, to set the record format and record length to the correct values:

```
//SYSIN    DD   UNIT=SYSDA,SPACE=(4000,(20,20),,,ROUND),
//         DCB=(RECFM=V,LRECL=32708)
```

Severity: 8 (error)

DSNU024I   *csect-name* **PARM FIELD ERROR** *parm-field*

Explanation: An error was detected in the PARM field on the EXEC statement. was the input parameter field.

csect-name
> The name of the control section that issued the message.

parm-field
> The input parameter field that contained the error.

System action: Utility processing is not initiated.

User response: Correct the indicated parameter on the EXEC DD statement, and resubmit the job.

Problem determination: Check the syntax of the EXEC DD statement.

Severity: 8 (error)

Related concepts:

➡ Utility control statements (DB2 Utilities)

Related reference:

➡ Invoking a DB2 utility by creating the JCL data set yourself (DB2 Utilities)

DSNU025I   *csect-name* **INVALID RESTART PARAMETER (restart)**

Explanation: Only the following values may be specified for the *restart* parameter: *NO* or *null*, CURRENT, or PHASE.

System action: The utility is not run by the stored procedure.

User response: Correct and re-run.

Severity: 8 (error)

DSNU028I   *csect-name* **UNABLE TO CONNECT TO DB2, REASON=X***reason code*

Explanation: A connection to the DB2 subsystem failed. 'reason-code' is the DB2 reason code that was returned.

System action: The utility job is terminated.

User response: Check the subsystem ID specified in the EXEC DD statement. If it is correct, the subsystem was not ready for processing. Ask the system programmer when the requested subsystem will again be available for processing.

Severity: 8 (error)

DSNU030I   *csect-name* **UNABLE TO ALLOCATE** *dsn*, **RC=***rr*, **CODE=***ccc*

Explanation: The utility was unable to allocate a necessary image copy data set. *dsn* is the data set name that the utility attempted to dynamically allocate. *rr* is the return code from the MVS dynamic allocation service, and *ccc* is the reason code associated with the return code.

System action: The utility proceeds along one or the following paths:

If the data set was an incremental image copy data set, the utility proceeds as if it and any later incremental image copies did not exist.

If the data set was a full image copy data set, the utility attempts to use the most recent prior full image copy data set.

If the data set is used by the COPYTOCOPY utility, and the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, COPYTOCOPY processes the next item in the list.

The allocation failure may create other problems that will cause the utility to fail. This is indicated by other messages.

System programmer response: If the utility fails because of the allocation error (as shown by a subsequent message), correct the error, then resubmit the job.

User response: If the job fails, contact the system programmer.

Problem determination: Refer to the appropriate MVS publication for a complete description of the possible return and reason codes from dynamic allocation.

DSNU031I   **UNABLE TO** *action dsn*, **RC=***rc*, **CODE=***rsn*

Explanation: A utility was unable to perform the indicated action on a data set. Possible actions are UNALLOCATE or CATALOG. The data set name that the utility was processing is *dsn*. The return code from the MVS DYNALLOC (SVC 99) or CATALOG (SVC 26) service is *rc*. The reason code associated with the return code is *rsn*.

System action: The system action following this message varies. Following catalog errors from both DYNALLOC and CATALOG, processing continues. Following DYNALLOC errors unrelated to the catalog, the utility job is abended with reason code X'00E40076'.

Problem determination: For CATALOG errors, refer to the "Return Codes from Catalog" section of *z/OS DFSMSdfp: Using DFSMSdfp in the z/OS Environment* for possible return codes. For DYNALLOC errors, refer to *z/OS MVS Programming: Authorized Assembler Services Guide* for possible return and reason codes.

DSNU032I   **I/O ERROR ON WORKFILE,** *additional-information*

Explanation: An I/O error was encountered on a necessary OS/VS data set.

additional-information
> A message that describes the error and the data set in error. This message is returned to the DB2 utility by SVC 68 (SYNADAF) from BSAM.

The data set might have been needed for an unload data set, image copy data set, or load or reorganization work file data set. The utility is stopped at the beginning of the phase that used the OS/VS data set.

System action: The utility job is abended. The utility job step is in the stopped state.

Operator response: Except in the case of an unload data set during the RELOAD phase, the job can be restarted if an undamaged output data set is provided.

Problem determination: Refer to the appropriate MVS publication for the format of the error message.

DSNU033I   *csect-name* **THE CONTROL STATEMENT DOES NOT IDENTIFY THE OBJECT ON WHICH THE UTILITY OPERATES**

Explanation: In the utility control statement, a keyword or operand was specified before the name of the table space, index, index space, or list on which the utility operates.

csect-name
> The name of the control section that issued the message.

System action: Parsing is discontinued and utility processing is terminated.

User response: Specify the object of the utility before any keywords or operands that relate to the object.

Problem determination: Check the syntax of the utility control statement.

Severity: 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

➡ DB2 stand-alone utilities (DB2 Utilities)

DSNU034I   **DATA SET TYPE INVALID FOR DDNAME 'dddddddd'**

Explanation: A required DD statement with DDNAME='dddddddd' was found to have a data set type which is not valid for this application.

System action: The utility is terminated with a return code 8.

System programmer response: Correct the JCL and resubmit the job.

User response: The data set must be capable of storing intermediate data that can be retrieved by DB2. Check that the data set is not specified as type DUMMY, SYSIN, or SYSOUT. If you require assistance in specifying a valid data set type, consult the system programmer.

Severity: 8 (error)

DSNU035I   *csect-name* **UNABLE TO FIND DD CARD FOR DDNAME, 'dddddddd'**

Explanation: A DD statement with ddname = 'dddddddd' was expected, but not found.

System action: The utility is terminated with a return code 8.

System programmer response: Correct the JCL and resubmit the job.

User response: If the DDNAME is one you submitted, review and correct it. If the DDNAME was not provided by you, but by the system, consult the system programmer.

Severity: 8 (error)

DSNU036I • DSNU039I

DSNU036I　　*csect-name* **UNABLE TO OPEN DCB FOR DDNAME** *dddddddd*

Explanation:　The DCB associated with DDNAME 'dddddddd' could not be opened.

System action:　The utility job is abended with a reason code of X'00E40070'. The utility job step is in the stopped state.

System programmer response:　Correct the job control language and resubmit the job.

User response:　If the ddname is one you submitted, review and correct it. If the ddname was not provided by you but by the system, consult the system programmer.

DSNU037I　　*csect-name* **- RECFM=U FOR DDNAME** *dddddddd*

Explanation:　A data set to be used by the utility has an undefined record format RECFM=U. All data sets must be fixed or variable, blocked or unblocked.

System action:　The utility job is abended with a reason code of X'00E40070'. The utility job step is in the stopped state.

Operator response:　Notify the system programmer.

System programmer response:　Ensure that no data sets involved in the utility job have RECFM=U. Resubmit the job after ending the stopped utility by the TERM UTILITY command, or do a RESTART(PHASE) of the stopped utility.

DSNU038I　　*csect-name* **BLOCK SIZE FOR DCB, DDNAME** *dddddddd*, **IS TOO SMALL, MUST BE AT LEAST** *bbbbb* **BYTES**

Explanation:　The data set with DDNAME 'dddddddd' has a block size that is too small to accommodate a record that needs to be written.

System action:　The utility job is abended with a reason code of X'00E40079'. The utility job step is left in the stopped state.

User response:　Delete the data set in error. Change the JCL so that the block size for the data set with DDNAME 'dddddddd' is at least 'bbbbb' bytes. Resubmit the utility job with the RESTART(PHASE) option.

DSNU039I　　*csect-name* **- LRECL=0 FOR DDNAME** *dd-name*

Explanation:　A data set that is to be used by the DB2 utility has a record length of zero after OPEN.

csect-name
　　　　The name of the control section that issued the message.

dd-name
　　　　The name of the data definition statement.

Data sets with any of the following record formats must have a valid record length greater than zero:
- Fixed
- Fixed blocked
- Variable
- Variable blocked

System action:　The utility job abends with reason code 00E40070. The utility job step is left in the stopped state.

Operator response:　Notify the user or system programmer.

System programmer response:　Correct the data set definition, or add DCB information to the JCL. Do a RESTART(PHASE) of the stopped utility, or resubmit the job after issuing a -TERM UTILITY command for the stopped utility.

User response:　If the DDNAME is for a data set that you submitted, review and correct it. It might be necessary to provide a valid LRECL in the DCB parameter in the JCL. If the DDNAME was not provided by you but by the system, consult the system programmer.

Related concepts:

↪ Restart of an online utility (DB2 Utilities)

Related reference:

↪ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

↪ 00E40070 (DB2 Codes)

DSNU040I   *csect-name* **ERROR ACCESSING DDNAME** *ddname*, **DECB=X**hex string

Explanation: An error has occurred during BSAM access of a utility temporary file identified by 'ddname'. The DECB indicating the error is given as a hexadecimal string.

System action: The executing utility issues an X'04E' abend with abend reason code '00E40071' following this message.

Operator response: An operator with SYSOPR authority can issue the -TERM UTILITY command to release the resources claimed by the stopped utility (this may not be necessary).

User response: Refer to the information given in the Problem Determination section of this message. Correct the problem. Either issue the -TERM command to terminate the utility job and resubmit it, or specify the RESTART parameter and resubmit it.

Problem determination: The DECB provided contains the ECB in the first four bytes. The first byte of the ECB contains status bits that indicate the cause of the error. Refer to the appropriate MVS publication for the meaning of the status bits.

DSNU041I   *csect-name* **INVALID CONCATENATED DATA SET FOR DDNAME** *dddddddd*

Explanation: A non-input concatenated data set was detected. Only input data sets can be concatenated.

System action: The utility job abended with reason code X'00E40070'. The utility job step is in the stopped state.

Operator response: Notify the system programmer.

System programmer response: Remove the concatenated data sets from the data set with DDNAME 'dddddddd'. Resubmit the job after ending the stopped utility with the TERM UTILITY command, or restart the utility at the beginning of the stopped phase by using RESTART(PHASE).

DSNU042I   *csect-name* **SORT PHASE STATISTICS. NUMBER OF RECORDS=**n **ELAPSED TIME=**$hh/mm/ss$

Explanation: This message provides statistics on the SORT phase of a utility. The number of records sorted, 'n', and the duration of the SORT phase in hours, minutes, and seconds are provided.

System action: The system continues to process normally.

Severity: 0 (informational)

DSNU044I   *csect-name* **ERROR FROM SORT COMPONENT RC=**n, **UTILITY STOPPED**

Explanation: A nonzero return code 'n' was returned from an MVS sort. Error messages from sort are printed on the UTPRINT data set.

System action: The executing utility job is placed in the stopped state.

Operator response: An operator with SYSOPR authority can issue the -TERM UTILITY command to release the resources claimed by the stopped utility (this may not be necessary).

User response: Determine the cause of the error. If the error can be corrected, the utility may be restarted from the beginning of the sort phase by using the RESTART(PHASE) option. If the error is such that the utility cannot complete, the -TERM command should be issued to remove the stopped utility from the system. The -TERM utility option should rarely be exercised when the stopped utility is REORG.

Problem determination: This abend is accompanied by abend reason code X'00E40005', and an SVC dump is requested. In most cases, error messages from MVS sort identify the problem and the dump is not needed. If sort is

unable to open the UTPRINT data set, only the message and the abend are provided. For example, if the UTPRINT DD statement is omitted, correct the problem with UTPRINT, and resubmit the job.

DSNU046I   *csect-name* **UTILITY STATEMENT IS TOO LONG OR TOO COMPLEX**

Explanation: The utility command and associated control statements have caused internal fields to exceed DB2 subsystem limits.

System action: Utility processing is not initiated.

User response: Reduce the size or complexity of the utility statement where possible. For example, a LOAD into several tables may be broken up into a separate load statement for each table or the size of comparison fields for WHE, CONTINUEIF, and NULLIF operands may be reduced in size.

Severity: 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

➡ DB2 stand-alone utilities (DB2 Utilities)

Related reference:

➡ Limits in DB2 for z/OS (DB2 SQL)

DSNU047I   *csect-name* **A REQUIRED DD CARD OR TEMPLATE IS MISSING. NAME=***missing-dd*

Explanation: The DB2 utility being executed requires a data set to continue processing. The DDNAME or TEMPLATE name for the missing data set is provided.

System action: Utility processing terminates.

Operator response: Notify the system programmer.

System programmer response: Correct the JCL to include the indicated DD statement or add a TEMPLATE control statement.

Severity: 8 (error)

Related concepts:

➡ Invoking stand-alone utilities (DB2 Utilities)

Related tasks:

➡ Invoking DB2 online utilities (DB2 Utilities)

Related reference:

➡ Syntax and options of the TEMPLATE control statement (DB2 Utilities)

➡ Sample TEMPLATE control statements (DB2 Utilities)

DSNU048I   *csect-name* **INVALID OPERAND FOR KEYWORD** *invalid-keyword*

Explanation: The utility parser has determined that the operand specified for keyword *invalid-keyword* is invalid.

System action: Utility processing is not initiated.

User response: Correct the operand in error, and resubmit the utility job.

Severity: 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

➡ DB2 stand-alone utilities (DB2 Utilities)

DSNU049I   *csect-name* **- INVALID OPERAND** *invalid-operand* **FOR KEYWORD** *keyword-name*

Explanation: The utility parser has determined that operand *invalid-operand* cannot be specified for keyword *keyword-name*. This includes a negative value following the keywords MAXRO or DELAY.

System action: Utility processing is not initiated.

User response: Correct the operand in error, and resubmit the utility job.

Severity: 8 (error)

Related reference:

▶ Syntax and options of the REORG INDEX control statement (DB2 Utilities)

▶ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU050I   *csect-name utility-statement*

Explanation: This message echoes the utility statement that was input from SYSIN.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU051I   *csect-name* **INSUFFICIENT OPERANDS FOR KEYWORD** *keyword-name*

Explanation: An operand that is required when the keyword *keyword-name* is specified was omitted.

System action: Utility processing is not initiated.

User response: Correct the operand specification, and resubmit the job.

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU052I   *csect-name* **KEYWORD** *keyword-name* **INCOMPATIBLE WITH OTHER KEYWORDS OR OPERANDS**

Explanation: The specified keyword *keyword-name* cannot be used with other keywords or operands.

System action: Processing terminates.

User response: Correct or remove the invalid keyword, and resubmit the job.

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU053I   *csect-name* **FIELD** *cccccccc* **NOT FOUND**

Explanation: This error is caused by one of the following situations:

- Column name *cccccccc* is not a column name in the target table.
- A field name in a WHEN clause does not match a column name in the target table.
- The INCURSOR option was used to specify a cursor for the LOAD utility input data set. DB2 replaced expressions in the SELECT list of the cursor with new column names, such as DSN_SELECT_CLAUSE_EXPRESSION, which did not match column names in the target table.

System action: Utility processing is not initiated.

User response: Correct the LOAD utility statement, and resubmit the job.

If the error is caused by the existence of an INCURSOR option with expressions in the SELECT list of the cursor, rename expressions in the SELECT list by using the AS keyword.

Problem determination: Valid column names can be found in the SYSIBM.SYSCOLUMNS table.

Severity: 8 (error)

DSNU054I *csect-name* **TABLESPACE** *ssssssss* **NOT FOUND**

Explanation: The operand of the TABLESPACE keyword, 'ssssssss', did not specify an existing table space name.

System action: Utility processing is not initiated.

User response: Correct the table space name and/or database name, and resubmit the job.

Problem determination: Valid table space names can be found in the SYSIBM.SYSTABLESPACE catalog table.

Severity: 8 (error)

DSNU055I *csect-name* **INDEX** *iiiiiiii* **NOT FOUND**

Explanation: The index name specified, 'iiiiiiii', was invalid.

System action: Utility processing is not initiated.

User response: Correct the index name, and resubmit the job.

Problem determination: Valid index names can be found in the SYSIBM.SYSINDEXES catalog table.

Severity: 8 (error)

DSNU056I *csect-name* **TABLE** *table-name* **NOT FOUND**

Explanation: A utility job specified an invalid table name.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

This message occurs when:
- No table exists with the specified name.
- In the case of the UNLOAD utility, the table was identified with a view or alias. The UNLOAD utility cannot unload from a view or alias whose base table is remote.

System action: The utility job step is terminated, and resources are released to their prior state.

User response: Correct the table name, the database name, or both, and resubmit the utility job step.

Problem determination: To confirm table names, query the SYSIBM.SYSTABLES catalog table.

Severity: 8 (error)

Related concepts:

⮊ DB2 online utilities (DB2 Utilities)

Related reference:

⮊ SYSIBM.SYSTABLES table (DB2 SQL)

DSNU057I *csect-name* **DATABASE** *database-name* **NOT FOUND**

Explanation: The operand of the DATABASE keyword, 'database-name', did not specify an existing database name.

System action: The error is sufficient to terminate further processing of the utility job step. The utility job step is terminated, and resources are released to their prior state.

User response: Correct the database name and resubmit the utility job step.

Problem determination: The correct database names can be found by querying the SYSIBM.SYSDATABASE catalog table.

Severity: 8

DSNU058I *csect-name utility-name* **UTILITY NOT ALLOWED AGAINST AN OBJECT IN RECOVER PENDING STATE**

Explanation: An attempt has been made to execute a utility against an object that is in RECOVER PENDING STATE. CHECK, COPY, MODIFY, LOAD, QUIESCE, REORG, and RUNSTAT are not allowed.

System action: Utility processing is not initiated.

System programmer response: Determine which of the three utilities, LOAD REPLACE, RECOVER, or REPAIR, should be used to remove the object from the RECOVER PENDING STATE.

User response: Remove the object from the RECOVERY PENDING STATE by using either LOAD REPLACE, RECOVER, or REPAIR against the object. Resubmit the job.

Severity: 8 (error)

DSNU059I *csect-name utility-name* **UTILITY NOT ALLOWED TO RUN ON A ROSHARE READ DATABASE**

Explanation: An attempt was made to execute a utility on a ROSHARE READ database. Only the RUNSTATS, DIAGNOSE, and REPAIR DBD utilities are allowed to run on read-only shared databases.

System action: Utility processing is not initiated.

User response: Resubmit the utility job on the owning system, if necessary.

Severity: 8 (error)

DSNU060I *csect-name* **USER** *userid* **NOT AUTHORIZED FOR** *util-name* **UTILITY ON DATABASE** *dbase-name*

Explanation: User *userid* attempted to execute a utility *util-name* against an object in database *dbase-name*. The user was not authorized to execute utilities for database *dbase-name*, the user's authorization did not include this utility, or the user's authorization did not have a valid seclabel.

System action: Utility processing is not initiated.

System programmer response: Determine why the user was denied access. Either the authorization should be corrected, or another authorized user should invoke the utility. When LISTDEF was used, it can be ignored if the user wanted to execute utility on the object(s) he or she owns in implicit database with owner privilege.

User response: Ensure that the correct database was specified. If this operation involves a table with multilevel security with row granularity, then you need to have a valid security label to execute this utility.

Problem determination: The following catalog tables can be examined to determine who has proper authority: SYSIBM.SYSUSERAUTH, SYSIBM.SYSDBAUTH, SYSIBM.SYSRESAUTH, SYSIBM.SYSINDEXES (ownership), or SYSIBM.SYSTABLESPACE (ownership).

Severity: 12 (severe error)

DSNU061I *csect-name* **USER** *userid* **NOT AUTHORIZED, UTILID=***utilid*

Explanation: The named user issued a utility command and was not authorized to do so.

The -TERM UTILITY command can only be issued by:
- The submitter of the utility job.
- Users with SYSOPR, SYSADM, or SYSCTRL authority.
- Users with DBADM, DBCTRL, or DBMAINT authority over every database used by the utility.

  Even for users with sufficient database authority, the TERM UTILITY command will fail in these situations:
  - The user issued the command before the utility determined what databases it would access.
  - The utility job included the DIAGNOSE, REPORT, or STOSPACE utility. Those utilities must be terminated by the submitter of the job or by a holder of SYSOPR, SYSCTRL, or SYSADM authority.

Any user can issue a -DISPLAY UTILITY command.

System action: The -TERM UTILITY request for utility identifier *utilid* is rejected.

User response: Take the appropriate action:
- Wait to allow the utility to identify the databases it will access; then reissue the command.
- Obtain the required authorization and reissue the command.
- Have another user with proper authorization issue the command.

DSNU062I   *csect-name util-name* **UTILITY MAY NOT BE RUN ON OBJECT** *qualifier.object-name*

Explanation: An attempt was made to execute utility 'util-name' against object 'qualifier.object-name'. This is not allowed. The named object cannot be processed by the named DB2 utility.

System action: The utility job is terminated.

User response: Remove the invalid utility control statements and resubmit the job.

Severity: 8 (error)

DSNU063I   *csect-name utility-name* **UTILITY NOT ALLOWED AGAINST AN OBJECT IN REBUILD PENDING STATE**

Explanation: An attempt has been made to execute a utility against an object that is in REBUILD PENDING STATE. CHECK, COPY, MODIFY, LOAD, QUIESCE, REORG, and RUNSTAT are not allowed.

System action: Utility processing is not initiated.

System programmer response: Rebuild the index with the REBUILD utility to remove the object from REBUILD PENDING state.

User response: Remove the object from the REBUILD PENDING STATE by using either LOAD REPLACE, RECOVER, or REPAIR against the object. Resubmit the job.

Severity: 8 (error)

DSNU064I   *csect-name* **UTILITY NOT ALLOWED AGAINST SYSTEM DATABASE**

Explanation: An attempt was made to execute a utility against database DSNDB01 or DSNDB06. The following utilities are not allowed against DSNDB01:
- LOAD
- REORG TABLESPACE against SYSUTILX (allowed against other DSNDB01 table spaces)
- CHECK DATA
- REPAIR DBD
- RUNSTATS
- MODIFY STATISTICS

The following utilities are not allowed against DSNDB06:
- LOAD (except SYSIBM.SYSSTRINGS)
- CHECK DATA
- REPAIR DBD

System action: Execution of the utility job is stopped.

User response: Remove the invalid utility control statements and resubmit the job.

Severity: 8 (error)

DSNU065I   *csect-name* **UTILITY NOT ALLOWED AGAINST A WORK FILE**

Explanation: An attempt has been made to execute a utility against a work file database, such as DSNDB07, or the TEMP database. Utilities cannot be run on objects in the work file or TEMP databases.

csect-name
> The name of the control section that issued the message.

System action: Execution of the utility job is stopped.

User response: Remove the invalid utility control statements and resubmit the job.

Severity: 8 (error)

DSNU066I   *csect-name* **OPERAND** *oooooo* **NOT ALLOWED FOR** *uuuuuu* **UTILITY against** *tttttt* **TABLE**

Explanation: An attempt was made to specify the REPLACE option when executing the LOAD utility on a DB2 catalog table. It is not permissible to replace existing rows in a DB2 catalog table. Rows can only be added to the existing rows in a DB2 catalog table.

System action: Utility processing is not initiated.

User response: Correct the operand, and resubmit the job.

Severity: 8 (error)

DSNU067I   *csect-name* **- TABLE SPACE** *table-space-name* **IS NOT A LOB TABLE SPACE**

Explanation: The table space specified, *table-space-name*, is not a LOB table space. The object of the utility must be LOB table space.

System action: Utility returns completion code=8.

Programmer response: Correct the utility statement.

DSNU068I   **KEYWORD NOT SUPPORTED FOR** *lob-or-xml* **TABLESPACE**

Explanation: The keyword specified in the utility statement is not supported for the XML or LOB table space indicated by the *lob-or-xml* variable. Also, the *lob-or-xml* reported by the message cannot be explicitly named in the utility statement.

System action: Utility execution is terminated.

User response: Correct the utility statement.

Severity: 8 (error)

DSNU069I   *csect-name* **- AUXILIARY TABLE** *table-name* **NOT PERMITTED AS EXCEPTION TABLE**

Explanation: A auxiliary table was specified as an exception table.

System action: Utility returns completion code=8.

Programmer response: Specify an exception table that is not an auxiliary table.

DSNU070I   *csect-name* **KEYWORD OR OPERAND** *x* **INVALID WITH** *y*

Explanation: The use of a keyword or operand in a utility control statement is inconsistent with another keyword or operand.

x **and** *y* The keywords or operands that cannot be used together.

System action: The control statement in error is rejected and processing is halted.

User response: Correct the control statement in error and resubmit the job.

For example, the REORG TABLESPACE utility issues this message when the AUTOESTSPACE option is used and the table space is not organized by hash. In this case, remove the inconsistent option.

Severity: 8 (error)

Related concepts:

 DB2 online utilities (DB2 Utilities)

Related reference:

 Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU071I　　*csect-name* **KEYWORD** *x* **REQUIRES KEYWORD** *y*

Explanation:　When keyword 'x' is specified, keyword 'y' must also be specified.

System action:　The control statement in error is rejected and processing is halted.

User response:　Correct the control statement in error, and resubmit the job.

Severity:　8 (error)

DSNU072I　　*csect-name* **INVALID KEYWORD(S) SPECIFIED FOR TABLESPACE LIST**

Explanation:　The specified keyword cannot be included in a list of table spaces.

System action:　Processing terminates.

User response:　Remove the specified keyword from the list of table spaces.

Severity:　8 (error)

DSNU073I　　*csect-name* **- KEYWORD** *keyword* **IGNORED**

Explanation:　An unsupported or inapplicable keyword was used.

keyword　The keyword that was used.

System action:　Utility processing proceeds.

System programmer response:　Correct the syntax and run the utility again.

Severity:　0

Related concepts:

 DB2 online utilities (DB2 Utilities)

 DB2 stand-alone utilities (DB2 Utilities)

DSNU074I　　**UTILITY NOT SUPPORTED FOR** *object-type* **TABLE SPACE**

Explanation:　The utility statement failed because the table space specified was a LOB or XML table space, and the utility does not support they type of table space indicated.

System action:　The statement cannot be executed.

Programmer response:　Correct the utility statement.

DSNU075I　　*csect-name* **- TABLE=***schema-name***.***table-name***, LOB COLUMN=***column-name***, PARTITION:**
　　　　　　partition-number **- DEFINITION IS NOT COMPLETE**

Explanation:　The definition of the LOB column is not complete.

csect-name
　　　　The name of the control section that issued the message.

*schema-name***.***table-name*
　　　　The name of the table, qualified by the name of the table schema or creator.

column-name
　　　　The name of the LOB column.

partition-number
>   If the object is partitioned, the partition number that contains the incomplete definition.

>   The phrase PARTITION: *partition-number* is present in the message text only if the object is partitioned.

System action: If the utility is able to continue processing, it does so and issues this message with a severity of 4. Otherwise, utility processing terminates and this message is issued with a severity of 8.

Programmer response: Complete the LOB column definitions. A LOB column definition is not complete until the LOB table space, auxiliary table, and index on the auxiliary table have been created for the LOB column.

If the table space is partitioned, make sure that the LOB definition is complete for all partitions.

Severity: 4 (warning)

8 (error)

DSNU076I *csect-name* **- KEYWORD=***keyword* **INVALID FOR** *object-description* **TABLE SPACE=***database-name.tablespace-name*

Explanation: A utility statement included a keyword that is not valid for the type of object.

csect-name
>   The name of the control section that issued the message.

keyword The invalid keyword that was specified for the utility.

object-description
>   A description of the object attributes that make the keyword invalid.

database-name.tablespace-name
>   The name of the table space that the utility was attempting to process.

System action: The utility terminates with a completion code 8.

Programmer response: Correct the utility statement.

DSNU077I *csect-name* **-** *utility* **IS PROHIBITED BECAUSE** *object* **IS** *pending-state*

Explanation: The *utility* specified is prohibited because the *object* is in a *pending-state*.

System action: Utility terminates and returns completion code=8.

Programmer response:
- Run the CHECK LOB utility if the object is a LOB table space and the state is CHECK PENDING.
- Run the REBUILD INDEX utility if the object is an auxiliary index and the state is REBUILD PENDING.
- Run the CHECK INDEX utility if the object is an auxiliary index and the state is CHECK PENDING.

Severity: 8 (error)

DSNU078I *csect-name* **INVALID DBCS IDENTIFIER** *dbcs-id*

Explanation: The DBCS identifier *dbcs-id* is invalid. Reason codes are:

101   An odd number of bytes exists between the shift-out and the shift-in

102   No shift-in was found at the end of the identifier

103   DBCS blanks X'4040' are not allowed

104   There are no characters between the shift-out and the shift-in

105   Shift-out can not be the first byte of the DBCS character between the shift-out and the shift-in.

System action: Processing is terminated.

User response: Correct the DBCS identifier and resubmit the job.

Severity: 8 (error)

Related concepts:

SQL identifiers (DB2 SQL)

Language elements (DB2 SQL)

DSNU079I   *csect-name* **UNBALANCED PARENTHESES IN CONTROL STATEMENT**

Explanation: The utility parser has determined that there are unbalanced parentheses in the input string.

System action: Utility processing is not initiated.

User response: Determine the location of the missing parenthesis and insert it. The utility job can then be resubmitted.

Severity: 8 (error)

Related concepts:

Utility control statements (DB2 Utilities)

DB2 online utilities (DB2 Utilities)

DB2 stand-alone utilities (DB2 Utilities)

DSNU080I   *csect-name* **NO UTILITY STATEMENTS FOUND IN** *dd-name*

Explanation: The data set identified by dd-name did not contain any text. The system might have processed the end of file prematurely, or this job requires that the utility statements be skipped.

System action: Processing continues.

User response: If the data set is SYSIN, provide at least one utility statement, if none now exists, or verify that this job executes no utility statements. If the data set is the LISTDEF (SYSLISTD) or TEMPLATE (SYSTEMPL) data set, provide the required control statements in that data set or in SYSIN.

Severity: 4 (warning)

DSNU081I   *csect-name* **OPERAND** *operand-name* **EXCEEDS KEYWORD** *keyword-name* **MAXIMUM LENGTH**

Explanation: The utility parser has determined that operand *operand-name* exceeds the maximum length for keyword *keyword-name*.

System action: Utility processing is not initiated.

User response: Correct the operand, and resubmit the job.

Severity: 8 (error)

Related concepts:

DB2 online utilities (DB2 Utilities)

DB2 stand-alone utilities (DB2 Utilities)

Related reference:

Limits in DB2 for z/OS (DB2 SQL)

DSNU082I   *csect-name* **INVALID KEYWORD** *kkkkkkkk*

Explanation: The given keyword was not recognized within the context of this control statement.

System action: Utility processing is not initiated.

User response: Correct the keyword, and resubmit the job.

Severity: 8 (error)

DSNU083I   *csect-name* **INVALID HEXADECIMAL OPERAND** *hhhhhhhh*

Explanation: The utility parser has determined that the operand *hhhhhhhh* is invalid as a hexadecimal operand. It contains one of the following errors:

- the operand contains a character that is not hexadecimal
- it has an odd number of characters
- for a UX or GX string constant, it consists of a number of characters that is not a multiple of 4

System action: Utility processing is not initiated.

User response: Correct the keyword, and resubmit the job.

Severity: 8 (error)

DSNU084I   *csect-name* **INVALID GRAPHIC OPERAND** *gggg*

Explanation: The utility parser has determined that the operand 'gggg' is invalid as a hexadecimal operand. The graphic operand is either missing one or both of the shift characters or is not an even number of characters.

System action: Processing is terminated.

User response: Correct the utility statement, and resubmit the job.

Severity: 8 (error)

DSNU085I   *csect-name* **INVALID SPECIAL CHARACTER OPERAND** *ssss*

Explanation: The utility parser has determined that the operand 'ssss' is invalid as a special character operand. For a special character operand to be invalid, it must have either a shift-out or a shift-in character and be missing the corresponding shift character, or not have an even number of characters between the shift-out, shift-in characters.

System action: Processing is terminated.

User response: Correct the utility statement, and resubmit the job.

Severity: 8 (error)

DSNU086I   *csect-name ttttt* **I/O ERRORS ON SPACE=** *dddddddd*. **DATA SET NUMBER=** *nnn*. **I/O ERROR PAGE RANGE=** *aaaaaa, bbbbbb*.

Explanation: DB2 detected an error. This message is issued for, but not limited to the following reasons:

- I/O errors were detected on the named object. *ttttt* is either 'READ' or 'WRITE'. *nnn*; is the data set number. *aaaaaa* is the first damaged page and *bbbbbb* is the last damaged page of the data set.
- A broken page was detected.
- A page parity error was detected for a 'must complete' agent.
- A read or write I/O error was detected. A LPL recovery might be in progress so the page is added to the WEPR instead of the LPL.

System action: Processing continues, but the pages within the error range are no longer accessible.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

System programmer response: Determine the appropriate level of recovery, and invoke the recovery utility. If user-defined data sets are being used and full data set or table space RECOVER is to be used, then, prior to invoking the RECOVER utility, the table space must be stopped, new data set(s) defined, and the table space restarted. If the data set is part of a ROSHARE READ database, determine whether it was a link or DASD failure. It there was a link failure, then fix the link, and issue a -START DATABASE with the ACCESS(FORCE) parameter. If the error was caused by a DASD failure, then correct the problem from the system on which the database was defined as ROSHARE OWNER.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of database recovery procedures.

An IOS (I/O subsystem) message might be included on the MVS console SYSLOG indicating the type of I/O error

that occurred. If an IOS message was issued, follow the Problem Determination directions for the message.

Severity: 0 (informational)

DSNU087I *csect-name* **INVALID OPERAND TYPE FOR KEYWORD** *keyword-name* **-** *invalid-operand*

Explanation: The utility parser has determined that the operand *invalid-operand* is the wrong type for the keyword *keyword-name*. An example of such an error would be a decimal operand where only a hexadecimal operand is allowed.

System action: Utility processing is not initiated.

User response: Correct the keyword, and resubmit the job.

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU088I *csect-name* **INVALID CHARACTER OPERAND** *invalid-operand*

Explanation: The utility parser has determined that the operand *invalid-operand* is invalid as a character operand.

System action: Utility processing is not initiated.

User response: Correct the keyword, and resubmit the job.

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU089I *csect-name* **OPERAND** *invalid-operand* **IS TOO LONG**

Explanation: The utility parser has determined that the operand *invalid-operand* is too long for the keyword specified.

System action: Utility processing is not initiated.

User response: Correct the keyword, and resubmit the job.

Problem determination:

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU090I *csect-name* **DD OR TEMPLATE NAME** *nnn* **IS INVALID FOR KEYWORD** *kkk*

Explanation: The DDNAME or TEMPLATE *nnn* was used as an improper input of the specified keyword *kkk* or from one of the reserved ddnames: SYSIN, SYSPRINT, UTPRINT, SORTWKnn, SORTLIB.

System action: Processing terminates.

User response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU091I   *csect-name* **DDNAME** *nnn* **IS A DUPLICATE**

Explanation: The DDNAME 'nnn' was used as operand in the input statement more than once.

System action: Processing terminates.

User response: Change the ddname to a different one.

Severity: 8 (error)

DSNU092I   *csect-name* **NUMBER OF OPERANDS FOR KEYWORD** *keyword-name* **EXCEEDS MAXIMUM**

Explanation: The keyword *keyword-name* contains more than the maximum number of operands.

System action: Processing terminates.

User response: Correct the number of operands for the specified keyword.

Severity: 8 (error)

Related concepts:

DB2 online utilities (DB2 Utilities)

DB2 stand-alone utilities (DB2 Utilities)

DSNU093I   *csect* **- INVALID CHARACTER STRING** *string* **IN CONTROL STATEMENT**

Explanation: An invalid character string *string* was found in a utility control statement where a keyword, operand, operator or punctuation was expected.

System action: Processing terminates.

User response: Correct the utility control statement, and resubmit the job.

Severity: 8 (error)

DSNU094I   *csect-name* **KEYWORD 'wwwww' IS DUPLICATED**

Explanation: The keyword 'wwwww' is not allowed twice.

System action: The control statement in error is rejected and the processing is halted.

User response: Correct the control statement in error, and resubmit the job.

Severity: 8

DSNU095I   *csect-name* **UTILITY** *uuuuuuuu* **INCONSISTENT WITH LAST INVOCATION**

Explanation: This message can occur when a utility job step is restarted and a utility statement was modified, added or deleted from SYSIN.

System action: Utility processing is terminated. The utility job step remains in SYSIBM.SYSUTIL in the stopped state.

User response: Restart the utility job with the statements in the same order as they were in when the job was first run. If a statement was modified, change the statement back to the format in the original invocation and restart the job. Utility statements should not be altered for restart.

Severity: 8 (error)

DSNU096I   *csect-name* **OPERAND** *operand* **INCONSISTENT WITH LAST INVOCATION**

Explanation: When the utility job step was restarted, the indicated operand was found to be inconsistent with the restart point.

System action: The restart of utility processing is not initiated.

Operator response: An operator with SYSOPR authority can terminate the utility job step (this may not be necessary). A REORG utility that is beyond the UNLOAD phase should rarely be terminated.

User response: To restart, restore the operand to its prior definition. If this is not acceptable, terminate the utility job step, take corrective action, and resubmit the job.

Related concepts:

▶ Termination of REORG TABLESPACE (DB2 Utilities)

▶ Termination of REORG INDEX (DB2 Utilities)

▶ Restart of REORG INDEX (DB2 Utilities)

▶ Restart of REORG TABLESPACE (DB2 Utilities)

Related reference:

▶ REORG TABLESPACE (DB2 Utilities)

▶ REORG INDEX (DB2 Utilities)

DSNU097I   *csect-name* **- *input-value* WILL BE REQUIRED SYNTAX IN FUTURE RELEASES**

Explanation: The identified *input-value* is not required in this release, but it will be part of the required syntax in future releases.

System action: Utility processing continues.

Programmer response: Check the syntax of the utility control statement.

Severity: 4 (warning)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

DSNU099I   *csect-name* **- IRLM '*req-type*' REQUEST FAILED, IRLM RETURN CODE = X'*return-code*' IRLM REASON CODE = X'*reason-code*'**

Explanation: The request to the IRLM to obtain or release a lock failed for the reasons specified in *return-code* and *reason-code*. In a non data-sharing environment, the requested lock is the utility serialization lock. In a data sharing environment, the requested lock is the utility serialization lock, the utility identifier lock, or the utility execution lock. The type of request is indicated by *req-type*.

System action: Utility and command processing terminates.

System programmer response: Determine the cause of the IRLM error and take appropriate action.

User response: Refer to the IRLM messages and codes section for problem determination and the appropriate action. Restart the utility by specifying the RESTART parameter, or terminate the utility with the -TERM command and then resubmit the job. If this message was received during the -DIS UTIL, -TERM UTIL, or -START DATABASE ACCESS(FORCE) command, then the command can be reissued.

Problem determination: Obtain the system console log output, dump, and SYS1.LOGREC information. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Related concepts:

▶ IRLM return and reason codes (IRLM)

DSNU100I   *csect-name* **USERID =** *userid* **MEMBER =** *member-name* **UTILID =** *utilid* **PROCESSING UTILITY STATEMENT** *statement-number* **UTILITY =** *utility-name* **PHASE =** *utility-phase* **COUNT =** *object-count* **NUMBER OF OBJECTS IN LIST =** *list-size* **LAST OBJECT STARTED =** *last-started* **STATUS = STOPPED JOBNAME =** *job-name* **TIME STARTED =** *time-started*

Explanation: This message displays output from the DISPLAY UTILITY command. The output contains information about the status of utility jobs at the time that the DISPLAY UTILITY command was issued.

The DISPLAY UTILITY output begins with one of the following messages, depending on the status of the first utility that is reported:

DSNU100I: A stopped utility
> This message is displayed if the utility is stopped.

DSNU105I: An active utility
> This message is displayed if the utility is active.

DSNU106I: A terminating utility
> This message is displayed if the utility is in the process of terminating.

The initial DSNU100I, DSNU105I, or DSNU106I message is followed by additional instances of these messages as needed and one or more of the following messages:

- DSNU107I: Invalid member name
- DSNU108I: Invalid use of the MEMBER option
- DSNU111I: Subtask information
- DSNU116I: RECOVER utility LOGAPPLY phase
- DSNU347I: Deadline information for REORG INDEX or REORG TABLESPACE
- DSNU383I: LOG phase information for REORG INDEX or REORG TABLESPACE
- DSNU384I: MAXRO, LONGLOG, and DELAY information for REORG INDEX or REORG TABLESPACE
- DSN9022I: Normal completion

DSNU100I: A stopped utility:

Message DSNU100I is displayed for each instance of a stopped utility.

```
DSNU100I  csect-name USERID = userid
                MEMBER = member-name
                UTILID = utilid
                PROCESSING UTILITY STATEMENT statement-number
                UTILITY = utility-name
                PHASE = utility-phase   COUNT = object-count
                NUMBER OF OBJECTS IN LIST = list-size
                LAST OBJECT STARTED = last-started
                STATUS = STOPPED
                JOBNAME = job-name
                TIME STARTED = time-started
```

The specified utility job stopped during processing for one of the following reasons:

- The utility is REORG INDEX or REORG TABLESPACE, and the UNLOAD(PAUSE) option was specified. This option indicates that the utility is to stop after it unloads the data. Therefore, the utility is stopped in the RELOAD phase.
- A DB2 subsystem failure or cancellation occurred. For example, a cancellation occurs if you issue the STOP DB2 command.

csect-name
> The name of the control section that issued the message.

userid The user ID that ran the utility job.

member-name
> The name of the DB2 member in a data-sharing environment. In a non-data-sharing environment, *member-name* is blank.

utilid The utility ID.

statement-number
> The utility statement number.

utility-name
> The name of the utility.

DSNU100I

utility-phase
> The phase in which the utility is to be restarted if the utility job step is restarted with the RESTART(PHASE) option. This value applies to only those utilities that have phases.

object-count
> The number of pages or records that were processed in a utility phase. object-count has the following meanings for the listed utilities:

> **For CHECK INDEX, LOAD, REBUILD INDEX, REORG INDEX, and REORG TABLESPACE:**
>> The number of records that were processed. For the LOG phase of REORG INDEX and REORG TABLESPACE, this value indicates the number of log records that were processed. For the SWITCH phase of REORG INDEX and REORG TABLESPACE, this value is 0.

> **For COPY, MERGECOPY, RECOVER (RESTORE phase), and RUNSTATS:**
>> The number of pages that were processed.

> **For STOSPACE:**
>> The number of table spaces or indexes that were processed.

> For all other utilities, ignore the object-count field.

list-size If the utility is processing a list of objects, the total number of objects in the list. This value might not be accurate in the UTILINIT phase, which occurs before the list size is determined.

last-started
> If the utility is processing a list of objects, the item number of the last object for which processing started. This value might not be accurate in the UTILINIT phase, which occurs before the list size is determined.

job-name
> The name of the job that invoked the utility. If a stored procedure invoked the utility, this value is the job name in the JCL that started the WLM-managed stored procedure address space.

time-started
> The date and time when the utility started. The format is YYYY-MM-DD-HH:MM:SS.

Related information:

Syntax and options of the REORG INDEX control statement (DB2 Utilities)

Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

-STOP DB2 (DB2) (DB2 Commands)

Restart of an online utility (DB2 Utilities)

DSNU105I: An active utility:

Message DSNU105I is displayed for each instance of an active utility.

```
DSNU105I   csect-name USERID = userid
                  MEMBER = member-name
                  UTILID = utilid
                  PROCESSING UTILITY STATEMENT statement-number
                  UTILITY = utility-name
                  PHASE = utility-phase    COUNT = object-count
                  NUMBER OF OBJECTS IN LIST = list-size
                  LAST OBJECT STARTED = last-started
                  STATUS = ACTIVE
                  JOBNAME = job-name
                  TIME STARTED = time-started
```

In addition to being part of the DISPLAY UTILITY output, message DSNU105I can also be displayed when you run the REORG INDEX or REORG TABLESPACE utility. In that case, this message is displayed when the utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option. The information in the message reflects the state at the time that this event occurred.

csect-name
> The name of the control section that issued the message.

userid   The user ID that ran the utility job.

member-name
> The name of the DB2 member in a data-sharing environment. In a non-data-sharing environment, *member-name* is blank.

utilid   The utility ID.

statement-number
> The utility statement number.

utility-name
> The name of the utility.

utility-phase
> The phase in which the utility was executing when this message was issued. This value applies to only those utilities that have phases.

object-count
> The number of pages or records that were processed in a utility phase. *object-count* has the following meanings for the listed utilities:

> **For CHECK INDEX, LOAD, REBUILD INDEX, REORG INDEX, and REORG TABLESPACE:**
>> The number of records that were processed. For the LOG phase of REORG INDEX and REORG TABLESPACE, this value indicates the number of log records that were processed. For the SWITCH phase of REORG INDEX and REORG TABLESPACE, this value is 0.

> **For COPY, MERGECOPY, RECOVER (RESTORE phase), and RUNSTATS:**
>> The number of pages that were processed.

> **For STOSPACE:**
>> The number of table spaces or indexes that were processed.

> For all other utilities, ignore the *object-count* field.

list-size   If the utility is processing a list of objects, the total number of objects in the list. This value might not be accurate in the UTILINIT phase, which occurs before the list size is determined.

last-started
> If the utility is processing a list of objects, the item number of the last object for which processing started. This value might not be accurate in the UTILINIT phase, which occurs before the list size is determined.

job-name
> The name of the job that invoked the utility. If a stored procedure invoked the utility, this value is the job name in the JCL that started the WLM-managed stored procedure address space.

time-started
> The date and time when the utility started. The format is YYYY-MM-DD-HH:MM:SS.

Related information:

> Syntax and options of the REORG INDEX control statement (DB2 Utilities)

> Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU106I: A terminating utility:

Message DSNU106I is displayed for each instance of a utility that is in the process of terminating when the DISPLAY UTILITY command is issued. The utility is terminating in response to a TERM UTILITY command request.

```
DSNU106I  csect-name utility-name UTILITY,
          MEMBER = member-name
          UTILID = utilid TERMINATING
```

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility.

utilid   The utility ID.

DSNU100I

member-name
> The name of the DB2 member in a data-sharing environment. In a non-data-sharing environment, *member-name* is blank.

Related information:

-TERM UTILITY (DB2) (DB2 Commands)

DSNU107I: Invalid member name:

Message DSNU107I is displayed if an invalid member name was specified in the MEMBER option of the DISPLAY UTILITY command. DB2 ignores the invalid member name and continues to process the next member name that was specified.

DSNU107I *csect-name* MEMBER NAME *member-name* IS NOT A MEMBER OF THIS DATA SHARING GROUP.

csect-name
> The name of the control section that issued the message.

member-name
> The invalid member name.

DSNU108I: Invalid use of the MEMBER option:

Message DSNU108I is displayed when the DISPLAY UTILITY command is issued with the MEMBER option and data sharing is not active. If data sharing is not active, MEMBER is not a valid option. DB2 ignores the MEMBER option and its parameters and continues processing.

DSNU108I *csect-name* DATA SHARING ENVIRONMENT IS NOT ACTIVE, MEMBER KEYWORD NOT APPLICABLE.

csect-name
> The name of the control section that issued the message.

DSNU111I: Subtask information :

If the utility is using subtasks to process work in parallel, message DSNU111I provides an estimate of how much processing the utility subtask has completed. This message immediately follows a DSNU105I message. One DSNU111I message is displayed for each subtask.

In a data-sharing environment, this message is issued only when the DISPLAY UTILITY command is entered on the same member where the utility is executing.

DSNU111I *csect-name* - SUBPHASE=*subphase-name* COUNT=*object-count*

csect-name
> The name of the control section that issued the message.

subphase-name
> The activity that the subtask was performing at the time that the DISPLAY UTILITY command was issued.

object-count
> The number of records that were processed by the utility subtask.

DSNU116I: RECOVER utility LOGAPPLY phase:

If the utility is the RECOVER utility and is in the LOGAPPLY phase, message DSNU116I provides information about how much processing RECOVER has completed in this phase.

```
DSNU116I csect-name - RECOVER LOGAPPLY PHASE DETAILS:
                      STARTING TIME = start-time
                      START RBA = start-rba START LRSN = start-lrsn
                      END RBA = end-rba      END LRSN = end-lrsn
                      LAST COMMITTED RBA/LRSN = last-committed
                      ELAPSED TIME = elapsed-time
```

csect-name
> The name of the control section that issued the message.

start-time
> The time that the LOGAPPLY phase started.

start-rba **and** *end-rba*

> The relative byte address (RBA) of the first and last log records in the range of log records that are to be applied. The range accounts for all objects in the list of objects to be recovered. In a data-sharing environment, these values are a series of zeros.

start-lrsn **and** *end-lrsn*

> In a data-sharing environment, the log record sequence number (LRSN) of the first and last log records in the range of log records that are to be applied. The range accounts for all objects in the list of objects to be recovered.

last-committed

> The RBA or LRSN of the last log record that was applied at the last commit point in the LOGAPPLY phase.
>
> This value might not increase if multiple DISPLAY UTILITY commands are issued consecutively, because the log records are sorted by object and page number.

elapsed-time

> The time that has elapsed since the start of the LOGAPPLY phase, in the format *hh:mm:ss*.

Related information:

> RECOVER (DB2 Utilities)
>
> How recovery works in a data sharing group (DB2 Data Sharing Planning and Administration)
>
> Optimizing the LOGAPPLY phase (DB2 Utilities)
>
> How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU347I: Deadline information for REORG INDEX or REORG TABLESPACE:

Message DSNU347I displays the most recently specified value of the DEADLINE option for the REORG INDEX or REORG TABLESPACE utility. A previous invocation of the ALTER UTILITY command might have specified a value for DEADLINE. If not, the most recently specified value is the value that was specified with the invocation of REORG INDEX or REORG TABLESPACE.

This message is displayed if all of the following conditions are true:
- The utility is REORG INDEX or REORG TABLESPACE.
- The SHRLEVEL REFERENCE or SHRLEVEL CHANGE option was specified.
- The utility is active, as indicated by the DSNU105I message.

In addition to being part of the DISPLAY UTILITY output, message DSNU347I can also be displayed when you run REORG INDEX or REORG TABLESPACE. In that case, this message is displayed when the utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option. The information in the message reflects the state at the time that this event occurred.

```
DSNU347I csect-name
         DEADLINE = yyyy-mm-dd-hh.mm.ss.mmmmmm
```

csect-name

> The name of the control section that issued the message.

yyyy-mm-dd-hh.mm.ss.mmmmmm

> The timestamp value of the DEADLINE option.

Related information:

> Syntax and options of the REORG INDEX control statement (DB2 Utilities)
>
> Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)
>
> -ALTER UTILITY (DB2) (DB2 Commands)

DSNU383I: LOG phase information for REORG INDEX or REORG TABLESPACE:

Message DSNU383I displays information about the LOG phase for the REORG INDEX or REORG TABLESPACE utility. It provides an estimate of how far the utility has proceeded in the indicated utility job step.

This message is displayed if all of the following conditions are true:
- The utility is REORG INDEX or REORG TABLESPACE.
- The SHRLEVEL CHANGE option was specified.

DSNU100I

- The utility is active, as indicated by the DSNU105I message.
- The utility is in the LOG phase.

In addition to being part of the DISPLAY UTILITY output, message DSNU383I can also be displayed when you run REORG INDEX or REORG TABLESPACE. In that case, this message is displayed when the utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option. The information in the message reflects the state at the time that this event occurred.

```
DSNU383I csect-name CURRENT ITERATION NUMBER = nnnnnn
WRITE ACCESS ALLOWED IN CURRENT ITERATION = www
ITERATION BEFORE PREVIOUS ITERATION:
    ELAPSED TIME = ha:ma:sa
    NUMBER OF LOG RECORDS PROCESSED = nnnnna
PREVIOUS ITERATION:
    ELAPSED TIME = hb:mb:sb
    NUMBER OF LOG RECORDS PROCESSED = nnnnnb
CURRENT ITERATION:
    ESTIMATED ELAPSED TIME = hc:mc:sc
    ACTUAL ELAPSED TIME SO FAR = hd:md:sd
    ACTUAL NUMBER OF LOG RECORDS BEING PROCESSED = nnnnnd
CURRENT ESTIMATE FOR NEXT ITERATION:
    ELAPSED TIME = he:me:se
    NUMBER OF LOG RECORDS TO BE PROCESSED = nnnnne
```

csect-name
> The name of the control section that issued the message.

nnnnnn The current iteration number.

www An indication of whether write access is allowed in the current iteration of log processing. Possible values are YES or NO.

ha:ma:sa
> The elapsed time for the iteration before the previous iteration. If the current iteration number is 1 or 2, this value is 00:00:00

nnnnna The number of log records that were processed in the iteration before the previous iteration. If the current iteration number is 1 or 2, this value is 0.

hb:mb:sb
> The elapsed time for the previous iteration. If the current iteration number is 1, this value is 00:00:00.

nnnnnb The number of log records that were processed in the previous iteration. If the current iteration number is 1, this value is 0.

hc:mc:sc The estimated elapsed time for the current iteration.

hd:md:sd
> The elapsed time so far for the current iteration.

nnnnnd The actual number of log records that are being processed for the current iteration.

he:me:se The currently estimated elapsed time for the next iteration.

nnnnne The currently estimated number of log records to be processed for the next iteration.

Related information:
> REORG INDEX (DB2 Utilities)
>
> REORG TABLESPACE (DB2 Utilities)

DSNU384I: MAXRO, LONGLOG, and DELAY information for REORG INDEX or REORG TABLESPACE:

Message DSNU384I displays the most recently specified values of certain options for the REORG INDEX or REORG TABLESPACE utility. A previous invocation of the ALTER UTILITY command might have specified a value for each option. If not, the most recently specified value is the value that was specified with the invocation of REORG INDEX or REORG TABLESPACE.

This message is displayed if all of the following conditions are true:

- The utility is REORG INDEX or REORG TABLESPACE.

| • The SHRLEVEL CHANGE option was specified.

| • The utility is active, as indicated by the DSNU105I message.

| In addition to being part of the DISPLAY UTILITY output, message DSNU384I can also be displayed when you run
| REORG INDEX or REORG TABLESPACE. In that case, this message is displayed when the utility estimates that the
| SWITCH phase will not start by the deadline that is specified in the DEADLINE option. The information in the
| message reflects the state at the time that this event occurred.

```
| DSNU384I csect-name -
|          MAXRO = maxro-value SECONDS
|          LONGLOG = longlog-value
|          DELAY = delay-value SECONDS
```

| *csect-name*
| The name of the control section that issued the message.

| *maxro-value*
| The most recently specified value of the MAXRO option. Possible values are either an integer or DEFER.

| *longlog-value*
| The most recently specified value of the LONGLOG option. Possible values are CONTINUE, TERM, or
| DRAIN.

| *delay-value*
| The most recently specified value of the DELAY option. The value is an integer value.

| For more information about these option values, see the option descriptions in the REORG INDEX or REORG
| TABLESPACE syntax information.

| **Related information:**
| Syntax and options of the REORG INDEX control statement (DB2 Utilities)
| Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)
| -ALTER UTILITY (DB2) (DB2 Commands)

| **DSN9022I: Normal completion:**

| The DISPLAY UTILITY output normally ends with message DSN9022I.

| In a non-data-sharing environment, this message is displayed as follows:

```
| DSN9022I - DSNUGCC  '-DISPLAY UTILITY' NORMAL COMPLETION
```

| In a data-sharing environment, this message is displayed as follows:

```
| DSN9022I -DB1G DSNUGCC '-DIS UTIL' NORMAL COMPLETION
```

| **Related information:**
| "DSN9022I" on page 1144

| **System action:**   In most cases, processing continues with the following exceptions:
| • For each DSNU100I message, DB2 waits for directions to either restart or terminate the utility.
| • For each DSNU105I message that was issued because REORG INDEX or REORG TABLESPACE estimated that the
| SWITCH phase will not start by the specified deadline, reorganization terminates with message DSNU374I.
| • For each DSNU106I message, the utility stops and releases all resources that are associated with the utility. After
| the utility terminates, DB2 can reuse the utility ID.

| **Operator response:**   For each DSNU100I message, if you need to release the resources that are claimed by the
| stopped utility, terminate the utility by issuing the TERM UTILITY command.

| **User response:**   For each DSNU100I message, restart or terminate the utility job step. To restart the job step, follow
| the directions in Restart of an online utility (DB2 Utilities). You must have authority to run the indicated utility and
| to complete any other utility functions in the utility job step. To terminate the utility, issue the TERM UTILITY
| command. You must be the originator or have SYSOPR or SYSADM authority.

| For each DSNU105I message that was issued by REORG INDEX or REORG TABLESPACE, take the actions that are
| listed in "DSNU374I" on page 688.

For each DSNU107I message, correct the member name and reissue the DISPLAY UTILITY command.

For a DSNU108I message, make sure that data sharing is active before issuing the DISPLAY UTILITY command with the MEMBER option.

In all other cases, no action is necessary.

If you issue a subsequent DISPLAY UTILITY command, any additional processing is reflected in the output. The output also includes any updates to the DEADLINE, MAXRO, LONGLOG, or DELAY options that were made by an ALTER UTILITY command.

Severity: For DSNU107I and DSNU108I:

4 (warning)

For all other DISPLAY UTILITY output messages:

0

Related reference:

➡ -DISPLAY UTILITY (DB2) (DB2 Commands)

➡ -TERM UTILITY (DB2) (DB2 Commands)

➡ -ALTER UTILITY (DB2) (DB2 Commands)

Related information:

"DSNU374I" on page 688

DSNU103I   *csect-name* **- UNABLE TO ACCESS SYSUTIL TABLE AT THIS TIME**

Explanation: An -ALTER UTILITY, -DISPLAY UTILITY, or -TERM UTILITY command was issued at the same time that the DSNDB01.SYSUTILX table space was the object of an active utility job. The utility job might be holding locks that would cause a deadlock condition to occur.

System action: The command processing is terminated.

User response: Enter the command again. This message is issued in response to each command until the utility action on the DSNDB01.SYSUTILX table space completes.

Severity: 8 (error)

DSNU104I   *csect-name* **- DSNDB01.SYSUTILX IN A PENDING STATE PROHIBITS FURTHER PROCESSING**

Explanation: The -ALTER UTILITY, -TERM UTILITY, or -DISPLAY UTILITY command was issued when the DSNDB01.SYSUTILX table space was in a pending state, which prohibits further processing. The pending state was either copy or recovery pending.

System action: Command processing is terminated.

System programmer response: Issue a -DISPLAY DATABASE command to determine the state of the DSNDB01.SYSUTILX table space. If the state is copy pending, do a full image copy. If the state is recovery pending, use the RECOVER utility to recover DSNDB01.SYSUTILX. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer so that the pending condition can be resolved.

Severity: 0 (informational)

DSNU105I   *csect-name* **- USERID=***userid* **MEMBER=***member-name* **UTILID=***utilid* **PROCESSING UTILITY STATEMENT** *statement-number* **UTILITY=***utility-name* **PHASE=***utility-phase* **COUNT=***object-count* **NUMBER OF OBJECTS IN LIST=***list-size* **LAST OBJECT STARTED=***last-started* **STATUS=ACTIVE JOBNAME =** *job-name* **TIME STARTED =** *time-started*

Explanation: This message is displayed in one of the following circumstances:

• As part of the output from the DISPLAY UTILITY command

- When the REORG INDEX or REORG TABLESPACE utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option

In either case, this message is described as part of the DISPLAY UTILITY output in "DSNU100I" on page 620.

DSNU106I   *csect-name utility-name* **UTILITY, MEMBER =** *member-name* **UTILID =** *utilid* **TERMINATING**

Explanation: This message is a section of output from the DISPLAY UTILITY command, which described in message "DSNU100I" on page 620.

DSNU107I   *csect-name* **MEMBER NAME** *member-name* **IS NOT A MEMBER OF THIS DATA SHARING GROUP.**

Explanation: This message is a section of output from the DISPLAY UTILITY command, which described in message "DSNU100I" on page 620.

Severity: 4 (warning)

DSNU108I   *csect-name* **DATA SHARING ENVIRONMENT IS NOT ACTIVE, MEMBER KEYWORD NOT APPLICABLE.**

Explanation: This message is a section of output from the DISPLAY UTILITY command, which described in message "DSNU100I" on page 620.

Severity: 4 (warning)

DSNU109I   *csect-name* **UTILITY** *utilid* **CAN ONLY BE TERMINATED OR ALTERED BY MEMBER =** *member-name*

Explanation: The -TERM UTILITY or -ALTER UTILITY command cannot terminate or alter the utility job with identifier *utilid*, because it belongs to member *member-name* of the DB2 data sharing group. Activeutility jobs can be terminated or altered only from the system on which they were running.

System action: If this message was issued for the -TERM UTILITY command, the command ignores the *utilid* and continues to process the next utilid, if any. If this message was issued for the -ALTER UTILITY command, the command is rejected, and further processing of the command is ended.

User response: Make sure the *utilid* identifies the utility you want to terminate or alter. If it does, issue the command from the DB2 member identified in the message.

Severity: 8 (error)

DSNU110I   *csect-name* **UTILID =** *utilid* **INVALID**

Explanation: An invalid utility ID was specified for one of the following commands:
- ALTER UTILITY
- DISPLAY UTILITY
- TERM UTILITY

csect-name
> The name of the control section that issued the message.

utilid   The invalid utility ID.

A utility ID is a 1 - 16 byte alphanumeric token that can contain any number of periods.

System action: The command is rejected and command processing ends.

User response: Review the utility ID that is provided in the message to determine why it does not conform to the syntax rules of a utility ID. Correct the utility ID and reissue the request.

Severity: 8 (error)

Related reference:

▣ -ALTER UTILITY (DB2) (DB2 Commands)

▣ -DISPLAY UTILITY (DB2) (DB2 Commands)

▣ -TERM UTILITY (DB2) (DB2 Commands)

DSNU111I   *csect-name* **- SUBPHASE=***subphase-name* **COUNT=***n*

Explanation: This message is a section of output from the DISPLAY UTILITY command, which described in message "DSNU100I" on page 620.

DSNU112I   *csect-name* **- NO AUTHORIZED UTILITY FOUND FOR UTILID =** *utilid*

Explanation: An ALTER UTILITY, DISPLAY UTILITY, or TERM UTILITY command was issued. However, DB2 cannot find the utility job that is identified by the specified utility ID. The utility ID might not be correct, or the utility might be terminated already.

csect-name
        The name of the control section that issued the message.

utilid   The utility ID.

System action: The command request is rejected.

User response: Verify that the utility ID was entered correctly. To view all currently valid identifiers, issue DISPLAY UTILITY (*) or DISPLAY UTILITY (*partial-utility-id**).

Severity: 0

Related reference:

▣ -ALTER UTILITY (DB2) (DB2 Commands)

▣ -DISPLAY UTILITY (DB2) (DB2 Commands)

▣ -TERM UTILITY (DB2) (DB2 Commands)

DSNU113I   *csect-name* **DISPLAY OR TERM UTILITY MESSAGE LIMIT HAS BEEN REACHED**

Explanation: This message indicates that the DISPLAY or TERM utility command that was specified generated more messages than will fit in the amount of storage that is available.

System action: The DISPLAY or TERM UTILITY command terminates.

User response: Specify a set of utility jobs using a partial-utility-id to limit the number of utilities being displayed or terminated. See the DB2 Command Reference for more information.

Severity: 0

DSNU114I   *csect-name* **UTILITY COMMANDS SUBSYSTEM FAILURE, REASON=X***code*

Explanation: This message indicates an abend has occurred in the utility commands subsystem.

System action: The command abends. In the case of the -TERM command, termination actions taken previously under this command are backed out.

User response: Prior messages and/or the SVC dump identify the failure. If the condition can be corrected, the command may be reissued.

Problem determination: In most cases, the failure is not user related. In some cases, prior messages indicate a user error that may be corrected. The problem may be as simple as SYSIBM.SYSUTIL being temporarily locked by another command or a utility job. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNU115I *csect-name utility-name* **UTILITY WITH UTILID =** *utilid* **CAN ONLY BE TERMINATED BY A MEMBER WITH THE SAME RELEASE =** *release*

Explanation: Command TERM UTILITY cannot terminate the utility job with identifier *utilid* because it belongs to a member with a release level identified in the message. Some utilities can be terminated only from a member of the DB2 data sharing group with the same release level.

System action: Command TERM UTILITY is rejected for the indicated utility job.

User response: Make sure the *utilid* identifies the utility you want to terminate. If it does, issue command TERM UTILITY from a member with a release level identified in the message in the DB2 data sharing group.

Severity: 0 (informational)

DSNU116I *csect-name* **- RECOVER LOGAPPLY PHASE DETAILS:**

Explanation: This message is a section of output from the DISPLAY UTILITY command, which is described in message "DSNU100I" on page 620.

DSNU118I *csect-name* **MAXIMUM NUMBER OF ERROR MESSAGES ISSUED**

Explanation: The utility issued the maximum number of messages that can be generated with the available memory for message buffers. DB2 suppresses any additional error messages.

System action: Utility processing continues.

User response: Refer to the message information for any other error messages that are issued by the utility and take the specified action to resolve the problems. Then rerun the utility to ensure that all errors are resolved.

Severity: 0 (informational)

DSNU120I *csect-name* **DEADLOCK INFORMATION: INTERVAL COUNT =** *interval-count*, **NUMBER OF WAITERS =** *waiter-count*.

Explanation: Utility processing encountered a resource that is currently unavailable because the resource is involved in a deadlock condition.

csect-name
> The name of the control section that issued the message.

interval-count
> Unique ID for this deadlock interval.

waiter-count
> The number of waiters involved in the timeout.

The introductory message text is followed by set of column headings and one or more rows with information about the resource holders:

LOCK NAME *lock-name*
> The lock resource name.

TYPE *lock-type*
> The type of lock, such as PG or DBD.

FUNC/STATE/DURATION *lock-information*
> Information about the lock function, state, or duration. Expected values depend on the type of information.

> **Lock function**
>> The requested lock function: LOCK, UNLOCK, or CHANGE.

> **Lock state**
>> The lock resource state, such as SHARED or EXCLUSIVE.

> **Lock duration**
>> The lock resource duration, such as MANUAL or PLAN.

DSNU121I

HOLDER/WAITER *hw-type*
> Indicator if the resource is a holder or a waiter.

PLAN-ID *plan-id*
> The plan name of the holder or waiter.

CORR-ID *correlation-id*
> The correlation ID of the holder or waiter.

MEMBER NM *member-name*
> The DB2 member name.

System action: The utility abnormally terminates with a X'04E' abend and a utility reason code. The dump is suppressed. Message DSNT500I is issued.

System programmer response: Use the information provided in messages DSNU121I (specifically information about holders and waiters) and DSNT500I to identify the lock resources and the utility jobs and applications that are involved in the deadlock.

User response: Use the TERM UTIL command to restart or terminate the utility.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IRLM trace output.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.
- Listing of the IRLM procedure.
- Results from SELECT * from SYSIBM.SYSDATABASE.

Related reference:

➡ DB2 abend completion codes (X'04E' and X'04F') (DB2 Codes)

➡ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

"DSNT500I" on page 555

Appendix C, "Resource types," on page 1153

DSNU121I   *csect-name* **TIMEOUT INFORMATION: NUMBER OF HOLDERS/WAITERS =** *hw-count*, **TIMEOUT FACTOR =** *timeout-factor*.

Explanation: Utility processing encountered a resource that is currently unavailable because of a timeout condition. The resource cannot be granted and the wait period has exceeded the maximum wait period specified by the RESOURCE TIMEOUT field on panel DSNTIPI (the IRLMRWT subsystem parameter).

csect-name
> The name of the control section that issued the message.

hw-count
> The number of holders or waiters involved in the timeout.

timeout-factor
> An internally defined wait period.

The introductory message text is followed by set of column headings and one or more rows with information about the resource holders:

LOCK NAME *text*
> The lock resource name.

TYPE *text*
>> The type of lock, such as PG or DBD.

FUNC/STATE/DURATION *text*
>> Information about the lock function, state, or duration. Expected values depend on the type of information.

>> **Lock function**
>>> The requested lock function: LOCK, UNLOCK, or CHANGE.

>> **Lock state**
>>> The lock resource state, such as SHARED or EXCLUSIVE.

>> **Lock duration**
>>> The lock resource duration, such as MANUAL or PLAN.

HOLDER/WAITER *text*
>> Indicator if the resource is a holder or a waiter.

PLAN-ID *text*
>> The plan name of the holder or waiter.

CORR-ID *hex*
>> The correlation ID of the holder or waiter.

MEMBER NM *text*
>> The DB2 member name.

System action: The utility abnormally terminates with a X'04E' abend and a utility reason code. The dump is suppressed. Message DSNT500I is issued.

System programmer response: Use the information provided in messages DSNU121I (specifically information about holders and waiters) and DSNT500I to identify the lock resources and the utility jobs and applications that are involved in the timeout.

User response: Use the TERM UTIL command to restart or terminate the utility.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SVC dump (or system dump), taken to SYS1.DUMPxx data set, as result of an operator-initiated dump command or SLIP trap exit.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- IRLM trace output.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.
- Listing of the IRLM procedure.
- Results from SELECT * from SYSIBM.SYSDATABASE.

Related reference:

➡ RESOURCE TIMEOUT field (IRLMRWT subsystem parameter) (DB2 Installation and Migration)

➡ DB2 abend completion codes (X'04E' and X'04F') (DB2 Codes)

➡ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

"DSNT500I" on page 555

Appendix C, "Resource types," on page 1153

DSNU122I   *csect-name* **USER** *userid* **NOT AUTHORIZED FOR** *util-name* **UTILITY ON THE OBJECT** *object-type* *obj-qual.obj-name* **IN THE IMPLICIT DATABASE** *db-name*

Explanation: User *userid* attempted to execute a utility *util-name* against an object in the implicit database *db-name*. The user either does not have ownership of the object in the implicit database, has no DSNDB04 authorization or was not given authorization to execute utilities on object *obj-name*.

System action: Utility processing is not initiated.

System programmer response: Determine why the user was denied access. Either the authorization should be corrected, or another authorized user should invoke the utility.

User response: Ensure that the correct database was specified. If this operation involves a table with multilevel security with row granularity, then this user needs to have a valid security label to execute this utility. If this error occurred processing a LISTDEF list, alter the list to exclude the objects that lack proper authorization. If the list cannot be modified, OPTIONS EVENT(ITEMERROR, SKIP) may be used to continue processing when this error occurs.

Problem determination: The following catalog tables can be examined to determine who has the proper authority: SYSIBM.SYSUSERAUTH, SYSIBM.SYSDBAUTH, SYSIBM.SYSRESAUTH, SYSIBM.SYSINDEXES (ownership), or SYSIBM. SYSTABLESPACE (ownership).

Severity: 12 (severe error)

DSNU123I   *csect-name* **- ATTEMPT TO USE NEW FUNCTION BEFORE NEW FUNCTION MODE**

Explanation: Functions that this release of DB2 introduces cannot be used before new-function mode has been enabled. A DB2 utility attempted to execute one of these functions, but new-function mode has not been enabled.

csect-name
> The name of the control section that issued the message.

System action: The utility job is terminated.

User response: Identify the new function that the utility is using. Either delay executing this function until new-function mode has been enabled, or modify the utility to perform functions that are allowed before new-function mode is enabled.

Severity: 8 (error)

DSNU124I   *csect-name* **-** *old-keywords* **SPECIFICATION IS IGNORED AND** *new-keywords* **IS IN EFFECT FOR CURRENT UTILITY EXECUTION**

Explanation: The utility determines that the specified keywords either do not apply or cannot be used with the current utility execution. The utility executes based on the specification of new keywords.

csect-name
> The name of the control section that issued the message.

old-keywords
> The keywords that the utility could not use

new-keywords
> The keywords that the utility substitutes for the ones that it cannot use.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU125I   *csect-name function* **WILL NO LONGER BE AVAILABLE IN A FUTURE RELEASE OF DB2**

Explanation: The specified function is deprecated and will be removed from a future release of DB2.

csect-name
> The name of the control section that issued the message.

function The deprecated function that was specified.

System action: Processing continues.

User response: No immediate action is required. However, prepare for the utility change in a future release of DB2.

Severity: 0

DSNU126I   *csect-name function* **IS NO LONGER SUPPORTED.**

Explanation: The specified function is no longer supported in the current DB2 release and environment.

csect-name
>   The name of the control section that issued the message.

function   The deprecated function that was specified.

System action: Processing terminates.

User response: Identify the deprecated function, and submit the corrected utility job.

| **Severity:**   8 (error)

DSNU160I   *csect-name - utility-name* **UTILITY, UTILID =** *utilid* **REQUESTED TO TERMINATE, PHASE =** *utility-phase* **WAS ACTIVE**

Explanation: This message is issued in response to a -TERM UTILITY command. It acknowledges that the request for termination was received and verified. At the time the -TERM UTILITY request was received, utility 'utility-name' was ACTIVE in the utility job step identified by UTILID 'utilid'.

System action: DB2 has accepted the request to terminate the indicated utility job step. The utility terminates at its next cleanup point.

User response: This message is an acknowledgement that the request has been accepted and is in progress.

DSNU161I   *csect-name - utility-name* **UTILITY WITH UTILID =** *util-id* **CAN NOT BE TERMINATED AT THE DB2 TRACKER SITE**

Explanation: DB2 does not permit the termination of certain utilities at a tracker site because terminating these utilities may leave the target objects in an unpredictable state. The next recovery cycle should cause the utilities to resolve. You may not terminate the following utilities at a tracker site:
* COPY
* LOAD
* REORG
* REPAIR

System action: The utility does not terminate.

System programmer response: Monitor the status of these utilities after the next tracker site recovery cycle.

User response: Contact the system programmer.

Severity: 8 (error)

DSNU162I   *csect-name - utility-name* **UTILITY WITH UTILID =** *util-id* **IS TERMINATING WITH OBJECTS THAT HAVE BEEN STARTED WITH ACCESS(FORCE) AND THEN DROPPED**

Explanation: Database objects that are used by the terminating utility were started with the command START DATABASE ACCESS(FORCE) and then dropped.

System action: Termination process continues.

User response: If the terminating utility is online REORG, TERM UTILITY does not rename and delete residual DB2-managed or user-defined data sets. You must delete those data sets manually.

Severity: 4 (warning)

DSNU164I   *csect-name cccccccc* **UTILITY, UTILID =** *uuuu* **ALREADY TERMINATING**

Explanation: This message is in response to a -TERM UTILITY command. It indicates that the utility job step indicated by UTILID 'uuuuu' has already been requested to terminate and is in the process of doing so now.

System action: DB2 is currently in the process of terminating the indicated utility job step. Upon completion, all

resources will have been released to their prior state, and all tracking of the indicated UTILID will have been removed. At that point, the UTILID is available for reuse.

User response: This message indicates that either the request was redundant, or someone else is also interested in the termination of the subject utility job step.

Severity: 0 (informational)

DSNU166I *csect-name utility-name* **UTILITY, UTILID =** *uid* **NOT EXECUTING, CLEANUP COMPLETE**

Explanation: This message is in response to a -TERM UTILITY command. The 'utility-name' utility identified by UTILID 'uid' was found to be stopped at the time of the request to terminate.

System action: Cleanup is performed for the indicated stopped utility. All resources held by the utility are freed and all storage of the utility job is deleted. The utility job is no longer restartable.

System programmer response: Resources held by this utility may have prevented applications, commands, and other utilities from executing. These have been freed, so that the inhibited tasks may now be performed.

Severity: 0 (informational)

DSNU169I **THE OBJECT CONVERSION REQUESTED BY** *requestor-type requestor-name requestor-operand* **IS IGNORED**

Explanation: Page set conversion between BASIC format and EXTENDED format was requested by the subsystem parameter setting or the utility keywords that are indicated in the message. The page set conversion is not allowed at this time and was ignored. The message occurs in the following situations:

- An object cannot be converted to EXTENDED mode while the subsystem is in conversion mode. This message is accompanied by message DSNU123I (ATTEMPT TO USE NEW FUNCTION).
- An object cannot be converted back to BASIC mode after the 6-byte logging limit is reached.

requestor-type
> The type of request, either ZPARM (for a subsystem parameter) or KEYWORD (for a utility keyword).
>
> If the type is ZPARM, the subsystem exceeded the logging threshold for 6-byte RBA and LRSN values, and the BASIC format is no longer valid on the subsystem. The subsystem must be converted to EXTENDED format. Objects cannot be opened for update until conversion is complete.

requestor-name
> The name of the requestor.

requestor-operand
> The operand that is ignored: NONE, BASIC, or EXTENDED.

System action: The request is ignored. Message DSNU123I is issued.

User response: Specify RBALRSN_CONVERSION NONE or contact your system administrator to change the OBJECT_UTILITY_CONVERSION subsystem parameter to NONE. Do not use RBALRSN_CONVERSION BASIC or EXTENDED while the system is in conversion mode. Do not use RBALRSN_CONVERSION BASIC after the 6-byte logging limit is reached.

System administrator response: While the system is in conversion mode, configure your subsystem parameters with OBJECT_UTILITY_CONVERSION set to NONE.

Severity: 4 (warning)

| **Related reference**:

| [→] REORG TABLESPACE (DB2 Utilities)

| [→] REORG INDEX (DB2 Utilities)

| [→] UTILITY OBJECT CONVERSION field (UTILITY_OBJECT_CONVERSION subsystem parameter) (DB2
| Installation and Migration)

| [→] OBJECT CREATE FORMAT field (OBJECT_CREATE_FORMAT subsystem parameter) (DB2 Installation and
| Migration)

| **Related information**:
| "DSNU123I" on page 634

DSNU170I   *csect-name* **UTILITY JOB TERMINATED BY TERM COMMAND**

Explanation: A -TERM UTILITY command was issued for this utility job step.

System action: The executing utility terminates processing and frees all held resources. All storage of the utility job
has been deleted. Therefore, the job is not restartable. Subsequent utilities in the same job step are not executed.

User response: All or portions of the utility job step can be resubmitted as needed.

Severity: 8 (error)

DSNU172I   *csect-name* **UTILITY JOB STOPPED BY STOP DB2 COMMAND**

Explanation: A -STOP DB2 command was issued during execution of the utility job step.

System action: The executing utility stops processing but continues to hold resources. Subsequent utilities in the
same job step are not executed. The storage of the utility job is retained. Therefore, the utility job step is restartable.

User response: Following a -START DB2 command, the utility job step may be restarted. Alternatively, the utility
may be terminated using the -TERM UTILITY command and resubmitted as a new job. A REORG utility that has
passed the UNLOAD phase should rarely be terminated.

Severity: 8 (error)

DSNU180I   *csect-name* **UTILITY IS NOT COMPATIBLE WITH THE** *utility-name* **UTILITY UTILID =** *utility-id*
                OBJECT = *space-name*.*object-name*

Explanation: Certain utilities cannot process the same table space or index space at the same time.

csect-name
        The name of the control section that issued the message.

utility-name
        The name of the utility.

utility-id
        The ID of the utility (UTILID).

space-name
        The name of the table space or index space.

object-name
        The name of the object.

These objects, or their related log entries, might be target objects of the utilities or a mapping table, or they might be
catalog or directory table spaces that are updated by the utilities. For example, DSNDB06.SYSCOPY is updated by
LOAD.

System action: The starting utility is not initiated, and further utility processing in that utility job step is terminated.

User response: Use the DISPLAY UTILITY command with the *utility-id* that is provided in the message to determine
when the utility that caused your utility to be rejected is completed.

To avoid this error when running concurrent REBUILD INDEX with SHRLEVEL CHANGE, specify multiple indexes

on a single utility execution instead of invoking multiple utilities.

Severity: 8 (error)

Related concepts:

➡ Concurrency and compatibility of utilities (DB2 Performance)

➡ DB2 online utilities (DB2 Utilities)

Related reference:

➡ -DISPLAY UTILITY (DB2) (DB2 Commands)

DSNU181I *csect-name* **RESTART NOT ALLOWED FOR** *utility-name* **UTILITY**

Explanation: An attempt was made to restart the named utility. DB2 does not allow this utility to be restarted.

System action: Utility processing is abended. Abend reason code 00E40018 might be issued.

System programmer response: If necessary, take actions to prevent access to the table space before terminating the utility.

User response: Issue the -TERM UTIL command to terminate the utility job and then rerun the job without the RESTART parameter.

DSNU182I *csect-name utility-name* **MUST BE THE ONLY UTILITY IN SYSIN**

Explanation: Utilities must process certain objects alone or in a small logically related group in a single utility control statement. Also, some utility control statements must be the only control statement in the job step. This message indicates that one of these restrictions was violated.

The following list gives examples of these restrictions:

- A CATMAINT utility control statement with the UPDATE UTILX option must be the only control statement in the job step.
- A BACKUP SYSTEM utility control statement must be the only control statement in the job step.
- The CHECK DATA utility must process DSNDB01.SYSUTILX alone. No other objects can be specified in the same CHECK DATA control statement, and no other control statements can be in the job step.
- The CHECK INDEX utility must process DSNDB01.SYSUTILX indexes alone. No other objects can be specified in the same CHECK INDEX control statement, and no other control statements can be in the job step.
- The COPY utility must process DSNDB01.SYSUTILX and its indexes alone. No other objects can be specified in the same COPY control statement, and no other control statements can be in the job step.
- The QUIESCE utility must process DSNDB01.SYSUTILX alone. No other objects can be specified in the same COPY control statement, and no other control statements can be in the job step.
- The RECOVER utility must process DSNDB01.SYSUTILX and its indexes alone. No other objects can be specified in the same RECOVER control statement, and no other control statements can be in the job step.
- The RECOVER utility must process DSNDB01.DBD01 and its indexes alone. No other objects can be specified in the same RECOVER control statement, and no other control statements can be in the job step.
- A REORG INDEX utility control statement for DSNDB01.SYSUTILX indexes must be the only control statement in the job step.
- During enabling-new-function mode, a REORG TABLESPACE utility control statement for DSNDB01.DBD01 or DSNDB01.SPT01 must be the only control statement in the job step.
- The REPAIR utility must process DSNDB01.SYSUTILX and its indexes alone. No other objects can be specified in the same REPAIR control statement, and no other control statements can be in the job step.
- A RESTORE SYSTEM utility control statement must be the only control statement in the job step.

In most cases, these utilities tolerate the presence of DIAGNOSE, LISTDEF, OPTIONS, and TEMPLATE control statements, if those statements are required for processing.

csect-name
        The name of the control section that issued the message.

utility-name
>   The name of the DB2 utility.

System action: If any work was performed by utilities in the same utility job step before the identified utility, that work persists. However, the utility that is identified in this message and any subsequent utilities are not executed, and the utility job step is placed in the stopped state.

User response: See the information about the failing utility for specific restrictions, which vary by utility.

Then, make the failing job step and control statement small enough to avoid the error. Remove other control statements and other objects from the failing statement until the error goes away. Remove the control statements that follow the failing statement and run them in a separate job step. If the error still occurs, remove the control statements that precede the failing statement, except those statements that are required to run the statement, such as TEMPLATE statements. If the error still occurs and the control statement lists multiple objects, put each object in separate control statements. Some grouping of objects might be possible in some cases, such as DSNDB01.SYSUTILX with its indexes.

Severity: 8 (error)

Related concepts:

⏩ DB2 online utilities (DB2 Utilities)

DSNU183I   *csect-name - utility-name* **UTILITY NOT COMPATIBLE WITH OTHER ACTIVE UTILITIES**

Explanation: The utility identified by 'utility-name' is executing against table space DSNDB01.SYSUTILX. This table space is integral to controlling the execution of utilities. Control over other utilities running at the same time would be lost if the utility were allowed to execute. Therefore, the identified utility must be executed alone. There must be no other active DB2 utilities in the system.

System action: Utility execution is terminated.

User response: Rerun the utility when there are no other utilities executing. Use the -DISPLAY UTILITY command to see what utilities are ACTIVE. Utilities in the STOPPED state are allowed.

Severity: 8 (error)

DSNU184I   *csect-name* **- DO NOT RECOVER OBJECT qual.obj-nm UNTIL THE NEXT TRACKER SITE RECOVERY CYCLE**

Explanation: If DB2 detects an inconsistency relating to a utility's status on object 'qual.obj-nm' at the tracker site, DB2 issues this message. This event can occur if the SYSUTIL record for the utility either does not exist, or reflects incorrect information about hte status of the utility.

System action: The utility issues ABEND04E with return code 00E40009.

System programmer response: Wait for the next recovery cycle to recover the object named in this message. If the object named in this message is a catalog or directory object, you cannot recover this object, or any other objects in the list of catalog and directory objects, or any user data or indexes. Wait until the next recovery cycle to recover the object.

User response: Contact the system programmer.

Severity: 8 (error)

DSNU185I   *csect-name* **THE VSAM DATA SET DOES NOT EXIST FOR** *obj-type obj-qual.obj-name part part-number*

Explanation: The VSAM data set(s) for the specified object has not been defined yet because the object was created with DEFINE NO.

System action: The utility skips the object that was created with DEFINE NO, and processing continues.

Severity: 0 (informational)

DSNU186I   *csect-name*- **A SPATIAL INDEX CANNOT BE PROCESSED BY THIS UTILITY**

Explanation: The target objects specified for the utility include one or more spatial indexes, which are not supported by this utility. If the utility specifies a table space, then you can drop the spatial indexes, run the utility, and then re-create the indexes.

System action: The utility does not execute.

Severity: 8

DSNU188I   *csect-name* **OPTION PRESORTED YES IS NOT VALID WHEN USED WITH LOAD** *reason-text*. **OPTION IS IGNORED.**

Explanation: The LOAD utility ran with the option PRESORTED YES in a situation where the option is not supported.

csect-name
> The name of the control section that issued the message.

reason-text
> The reason why the option is not valid.
> - PARTITION PARALLELISM ACTIVE
> - WITHOUT PARALLEL INDEX BUILD ACTIVE
> - SHRLEVEL CHANGE
> - SORTKEYS NO
> - ON A TABLE WITH NO INDEXES

System action: Utility processing continues, but the PRESORTED YES option is ignored. No BUILD subtask is initiated by the RELOAD phase. Keys are sorted and indexes built after RELOAD processing completes.

User response: No action is required.

Severity: 4 (warning)

Related reference:

➡ LOAD (DB2 Utilities)

DSNU189I   *csect-name* **OPTION PRESORTED YES WAS SPECIFIED, BUT THE LOAD UTILITY INPUT DATA SET KEYS WERE FOUND TO BE OUT OF SEQUENCE IN THE INPUT DATA. STATISTICS WILL NOT BE COLLECTED OR INVALIDATED FOR INDEX** *creator-name*.*index-name*

Explanation: The LOAD utility ran with the PRESORTED YES option, but the input data had keys out of sequence.

csect-name
> The name of the control section that issued the message.

creator-name.*index-name*
> The name of the index, identified by the name of its creator.

System action: Utility processing continues.

Inline statistics are not collected. If either the REPLACE or PART REPLACE option was used and keys are found out of order, real-time statistics are invalidated for the index.

User response: Run the RUNSTATS utility on the object.

Severity: 4 (warning)

Related reference:

➡ LOAD (DB2 Utilities)

➡ RUNSTATS (DB2 Utilities)

DSNU190I   *csect-name* **UNABLE TO RESTART LOAD RESUME WITH THE PRESORTED YES OPTION SPECIFIED**

Explanation: An attempt was made to restart the LOAD utility during the RELOAD phase with both the RESUME and PRESORTED YES options. During the RELOAD phase, the RESUME option cannot be used with the PRESORTED YES option. Use LOAD REPLACE PRESORTED YES instead.

csect-name
> The name of the control section that issued the message.

System action: Utility processing is abended. Either reason code 00E40018 or 00E40310 might be issued.

User response: Recover the table space and all indexes within the table space that were left in a rebuild pending state when the LOAD utility failed

Related concepts:

▶ Restart of LOAD (DB2 Utilities)

Related reference:

▶ LOAD (DB2 Utilities)

Related information:

▶ 00E40018 (DB2 Codes)

▶ 00E40310 (DB2 Codes)

DSNU191I   *csect-name* **LOB AND/OR XML COLUMN DATA WILL BE OMITTED WHEN USING THE FORMAT INTERNAL OPTION FOR TABLE** *creator-name.table-name*

Explanation: Either the LOAD or UNLOAD utility ran with FORMAT INTERNAL option on a table that contains LOB columns, XML columns, or both. LOB and XML data are not supported by the FORMAT INTERNAL option.

csect-name
> The name of the control section that issued the message.

creator-name.table-name
> The name of the table, identified by the name of its creator.

Use the FORMAT INTERNAL option only if any LOB or XML data is not of interest.

System action: Processing continues.

The LOAD utility omits the LOB and XML table spaces and leaves the base table space in the auxiliary check pending (ACHKP) status.

The UNLOAD utility unloads the base row data in the internal format. LOB and XML data is not unloaded.

User response: Correct the auxiliary check-pending status on the table that was loaded.

Severity: 4 (warning)

Related reference:

▶ LOAD (DB2 Utilities)

▶ UNLOAD (DB2 Utilities)

▶ Auxiliary CHECK-pending status (DB2 Utilities)

DSNU192I   *csect-name* **OPTION** *option-name* **IS NOT VALID WHEN USED WITH MORE THAN ONE TABLE FOR THE** *utility-name* **UTILITY**

Explanation: An attempt was made to use the LOAD or UNLOAD utility for multiple tables with an option that can be used against only one table.

csect-name
> The name of the control section that issued the message.

option-name
> The name of the option.

> **LOAD utility**
>> FORMAT INTERNAL or PRESORTED YES

> **UNLOAD utility**
>> FORMAT INTERNAL

utility-name
> The name of the utility.

System action: Utility processing terminates.

User response: Specify the option that is indicated by *option-name* for only one table.

Severity: 8 (error)

Related reference:

➡ LOAD (DB2 Utilities)

➡ UNLOAD (DB2 Utilities)

DSNU200I   *csect-name* **- ERROR RANGE ON TABLE SPACE** *database.tablespace-name* **PROHIBITS PROCESSING**

Explanation: The utility encountered an error range on the identified table space or partition, which prohibits it from continuing.

System action: The utility fails serialization and terminates.

User response: Correct the error range for the table space by running RECOVER ERROR RANGE. Then resubmit the original failing utility.

Severity: 8 (error)

DSNU201I   *csect-name* **- ERROR RANGE ON INDEX** *creator.index-name* **PROHIBITS PROCESSING**

Explanation: The utility encountered an error range on the identified index space or partition which prohibits it from continuing.

System action: The utility fails serialization and terminates.

User response: Identify the incorrect error range by issuing the DISPLAY command for the index. Run the RECOVER or REBUILD INDEX utility to restore the index or index partition that has the error range. Then resubmit the original failing utility.

Severity: 8 (error)

DSNU202I   *csect-name* **- RECOVER PENDING ON TABLE SPACE** *database.tablespace-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against a table space that is in a recover pending state.

System action: Utility processing is not initiated.

System programmer response: Determine which of the three utilities, LOAD REPLACE, RECOVER, or REPAIR, should be used to remove the object from the recover pending state.

User response: Remove the table space from the recover pending state by using either LOAD REPLACE, RECOVER, or REPAIR against the object. LOAD REPLACE with KEEPDICTIONARY is not allowed on a table space (with compression active) in the recover pending state. Resubmit the job.

Severity: 8 (error)

DSNU203I   *csect-name* **- RECOVER PENDING ON INDEX** *creator.index-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against an index that is in a rebuild pending state.

System action: Utility processing is not initiated.

User response: Remove the index from the rebuild pending state by using the REBUILD INDEX utility against the index. . Resubmit the job.

Severity: 8 (error)

DSNU204I   *csect-name* **- PAGESET REBUILD PENDING ON** *obj-type obj-qual.obj-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against a logical partition of an index when the index is in PAGESET rebuild pending state,

System action: Utility processing terminates.

User response: Remove the index space from PAGESET rebuild pending state by running the REBUILD INDEX utility against the entire index, not just a single logical partition. Resubmit the job.

Severity: 8 (error)

DSNU205I   *csect-name* **- LOGICAL ERROR RANGE (LOGICAL PAGE LIST) ON TABLE SPACE**
          database.tablespace-name **PROHIBITS PROCESSING**

Explanation: The utility cannot continue because it encountered a logical error range (logical page list) on the identified table space or partition.

System action: The utility terminates.

User response: Use the -START DATABASE command to correct the logical error range (LPL) for the table space. Then resubmit the original failing utility.

Severity: 8 (error)

DSNU206I   *csect-name* **- LOGICAL ERROR RANGE (LOGICAL PAGE LIST) ON INDEX** *creator.index-name*
          PROHIBITS PROCESSING

Explanation: The utility cannot continue because it encountered a logical error range (logical page list) on the identified index space or partition.

System action: The utility terminates.

User response: Issue the -DISPLAY command for the index to identify the logical error range (LPL). There are two ways to restore the index or index partition that has the LPL before resubmitting the original failed utility.
* Issue the START DATABASE command to clear the LPL status, then run the RECOVER or REBUILD INDEX utility.
* Run the RECOVER or REBUILD INDEX utility to restore the entire nonpartitioning index (NPI).

Severity: 8 (error)

DSNU207I   *csect-name* **- GROUP BUFFER POOL RECOVERY PENDING ON TABLE SPACE** *database.tablespace-name* **PROHIBITS PROCESSING**

Explanation: You cannot execute a utility on a table space that is in a group buffer pool recovery pending status.

System action: Utility processing does not start.

User response: Use the -START DATABASE command to remove the table space from the group buffer pool recovery pending status. Then resubmit the job.

Severity: 8 (error)

DSNU208I *csect-name* **- GROUP BUFFER POOL RECOVERY PENDING ON INDEX** *creator.index-name* **PROHIBITS PROCESSING**

Explanation: You cannot execute a utility on an index that is in a group buffer pool recovery pending (GRECP) status.

System action: Utility processing does not start.

User response: Run the REBUILD INDEX utility on the index to remove the table space from the group buffer pool recovery pending status. Then resubmit the job.

Severity: 8 (error)

DSNU209I *csect-name* **- RESTART PENDING ON** *obj-type database.tablespace-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against a table space or index space that is in a restart pending state.

System action: Utility processing is not initiated.

System programmer response: Use the DISPLAY THREAD TYPE(POSTPONED) to determine the status of the resource. Use the RECOVER POSTPONED command if recover has not already been started. Resubmit the job.

Severity: 8 (error)

DSNU210I *csect-name* **- INFORMATIONAL COPY PENDING ON** *obj-type obj-qual.obj-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute the RECOVER utility with the ERROR RANGE or PAGE options against an index in the informational copy pending state.

System action: Utility processing is not initiated.

User response: Use the REBUILD INDEX utility instead.

Severity: 8 (error)

DSNU211I *csect-name* **CHECK PENDING ON** *obj-type obj-qual.obj-name* **PROHIBITS PROCESSING**

Explanation: The requested utility can not be run against an index in the check pending state. The index may be out of sync with its table space.

System action: Utility processing is not initiated.

User response: Run either the CHECK INDEX utility or the REBUILD INDEX utility. The CHECK INDEX utility will identify any errors (for example, if the index keys do not match the data in the table space.) If there are no errors, the check pending state will be reset. The REBUILD INDEX utility will rebuild the index from the data in the table space.

Severity: 8 (error)

DSNU212I *csect-name* **- REORG PENDING ON TABLE SPACE** *database.index-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against a table space that is in a REORG PENDING state.

System action: Utility processing is not initiated.

User response: Execute REORG with SHRLEVEL NONE against the partitions in REORG PENDING state to rebalance the partitions and remove the REORG PENDING state.

Severity: 8 (error)

DSNU213I *csect-name* **- AUX CHECK PENDING ON TABLE SPACE** *table-space* **PROHIBITS PROCESSING**

Explanation: The *processing* is prohibited because the *table-space* is in an AUX CHECK PENDING state.

System action: Utility terminates and returns completion code=8.

Programmer response: Run the CHECK DATA utility with the AUXERROR INVALIDATE option. If there are no AUX CHECK errors, AUX CHECK PENDING will be reset. If AUX CHECK errors are found, AUX CHECK PENDING will be reset and the table space will be set to an AUX WARNING state. This allows the rows with AUX CHECK errors to be corrected. Correct the errors and rerun CHECK DATA. If no more AUX CHECK errors are found, the AUX WARNING state will be reset.

DSNU214I   *csect-name* **- REBUILD PENDING ON** *obj-type obj-qual.obj-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against an object that is in REBUILD-pending state.

System action: Utility processing is not initiated.

User response: Remove the object from the REBUILD-pending state by running REBUILD INDEX, and then resubmit the job.

Severity: 8 (error)

DSNU215I   *csect-name* **REFRESH PENDING ON** *ob-type database.objectname* **PROHIBITS PROCESSING**

Explanation: DB2 attempted to execute a utility against a table space or index that has refresh pending status.

System action: Utility processing is not initiated.

User response: If the object in refresh pending status is a table space, use the RECOVER utility to recover the table space to a prior point in time. You can also reload the table space with LOAD REPLACE. If the object in refresh pending status is an index, use the REBUILD INDEX utility to rebuild the index from the data.

Severity: 8 (error)

DSNU231I   *csect-name* **DICTIONARY WITH** *n* **ENTRIES HAS BEEN SUCCESSFULLY BUILT FROM** *m* **ROWS FOR TABLESPACE** *tablespace-name*

Explanation: A compression dictionary was built for the specified tablespace. This message is issued during the RELOAD phase of the LOAD utility and at the end of the UNLOAD phase of the REORG utility.

The variable *n* is the number of dictionary entries, and can be any of the following values: 512, 1024, 2048, 4096, or 8192. *n* depends on the number of rows being loaded or reorganized.

m is the number of rows used to build the dictionary.

System action: Processing continues.

Severity: 0 (informational)

DSNU232I   *csect-name* **KEEPDICTIONARY REQUESTED BUT COMPRESS ATTRIBUTE NOT DEFINED FOR TABLE SPACE** *table-space-name*

Explanation: The KEEPDICTIONARY option was specified in the LOAD or REORG TABLESPACE utility statement. This keyword should only be used with table spaces that have the COMPRESS YES attribute. Otherwise, it is ignored. The COMPRESS column of SYSIBM.SYSTABLEPART contains the current compression status.

System action: Processing continues. If the dictionary exists, it is erased if LOAD REPLACE or REORG is specified.

Severity: 4 (warning)

DSNU233I   *csect-name* **COMPRESSION DICTIONARY DOES NOT EXIST FOR TABLE SPACE** *table-space-name*

Explanation: The KEEPDICTIONARY option was specified in the LOAD or REORG TABLESPACE utility statement for a table space that does not have a compression dictionary.

System action: Processing continues. A dictionary is built for the table space.

Severity: 4 (warning)

DSNU234I   *csect-name* **COMPRESSION REPORT FOR TABLE SPACE** *table-space-name*

Explanation: The data in the table space was compressed. This message is issued at the end of the RELOAD phase for the LOAD and REORG utility. The statistics that follow apply only to the data that was loaded or reorganized. For example, RESUME(YES) reports statistics only on the additional rows that were loaded, not what was previously in the table space.

The compression statistics follow:

noncmpbytes
> The total number of uncompressed KB (where KB = 1024 bytes) for the loaded or reorganized data rows (including the row prefix). This number was rounded to the nearest KB.

cmpbytes
> The total number of compressed KB for the loaded or reorganized data rows (including the row prefix). This number was rounded to the nearest KB.

percsave
> The percentage of bytes saved from the loaded or reorganized data rows as a result of compression.

The values above reflect only the number of bytes for the data row and the respective row prefix (6 bytes). DB2 page and dictionary page overhead is not included.

pctrowcomp
> The percentage of the rows that were compressed within the total number of rows that were loaded or reorganized. The stored length and the external length of a row was compared to determine if the row was compressed. If an EDITPROC is involved, its effect on the length of the row is considered in the calculation.

noncmppages
> The estimated number of DB2 pages required by the uncompressed data rows that were loaded or reorganized. Space required for PCTFREE and FREEPAGE is included. This number was rounded up to the next integer.

cmppages
> The estimated number of DB2 pages required by the compressed data rows that were loaded or reorganized. Space required for PCTFREE and FREEPAGE is included. Dictionary pages are also included unless a LOAD RESUME(YES) was run. This number was rounded up to the next integer.

auncmprowlen
> Average uncompressed row length.

avgcmprowlen
> Average compressed row length.

pagesave
> An estimate of the percentage of DB2 pages saved by compressing the loaded or reorganized data rows.

System action: Processing continues.

Module:

noncmpbytes
> KB WITHOUT COMPRESSION

cmpbytes
> KB WITH COMPRESSION

percsave
> PERCENT OF THE BYTES SAVED FROM COMPRESSED DATA ROWS

pctrowcomp
> PERCENT OF THE LOADED ROWS WERE COMPRESSED

noncmppages
> PAGES REQUIRED WITHOUT COMPRESSION

cmppages
> PAGES REQUIRED WITH COMPRESSION

auncmprowlen
>   BYTES FOR AVERAGE UNCOMPRESSED ROW LENGTH

avgcmprowlen
>   BYTES FOR AVERAGE COMPRESSED ROW LENGTH

pagesave
>   PERCENT OF THE DB2 DATA PAGES SAVED USING COMPRESSED DATA

Severity: 0 (informational)

DSNU235I   *csect-name* **THE COMPRESSION DICTIONARY WAS NOT BUILT FOR TABLE SPACE** *ts-name*
>   *reason-text*

Explanation: The compression dictionary could not be built.

csect-name
>   The name of the control section that issued the message.

ts-name   The name of the table space.

reason-text
>   The reason that the dictionary could not be built:

>   **blank**   If no reason is indicated, then insufficient data was available to build the compression dictionary.

>>   **LOAD utility**
>>>   Either the input data set is empty or all of the input records were discarded.

>>   **REORG TABLESPACE utility**
>>>   The table space is empty.

>>   **Dictionary built during data insertion**
>>>   The partition does not yet contain enough data.

>   **CANNOT STORE DICTIONARY**
>>   The dictionary was built but cannot be stored. This condition is usually the result of a communication error between data sharing group members.

System action: The data is not compressed.

Processing continues.

User response: Take the action that corresponds to the *reason-text* value:

blank

>   **LOAD utility**
>>   Make sure that the input data set contains data and that the input records are not discarded by the WHEN clause or because of conversion errors.

>   **REORG TABLESPACE utility**
>>   Make sure that the table space is populated with data.

>   **Dictionary built during data insertion**
>>   No action is required. The data remains uncompressed. Optionally, you can:
>>   - To build a compression dictionary on a subsequent insertion of data, use the DB2 START DATABASE command to reopen the table space or partition.
>>   - To compress the data manually, use the REORG TABLESPACE utility.

CANNOT STORE DICTIONARY
>   Check that no communication error has occurred between data sharing group members. Optionally, you can:
>   - To build a compression dictionary on a subsequent insertion of data, use the DB2 START DATABASE command to reopen the table space or partition.
>   - To compress the data manually, use the REORG TABLESPACE utility.

Severity: 0

Related reference:

➡ LOAD (DB2 Utilities)

➡ REORG TABLESPACE (DB2 Utilities)

DSNU236I   *csect-name* **DICTIONARY FOR TABLE SPACE** *table-space-name* **HAS BEEN ERASED.**

Explanation: The compression dictionary of the specified table space was erased.

The COMPRESS attribute of the table space is NO, and there are no active rows in this table space. In this case, the LOAD RESUME NO utility erases the dictionary.

System action: Processing continues.

Severity: 0 (informational)

DSNU241I   *csect-name* **DICTIONARY WITH** *n* **ENTRIES HAS BEEN SUCCESSFULLY BUILT FROM** *m* **ROWS FOR TABLE SPACE** *table-space-name*, **PARTITION** *part-num*

Explanation: A compression dictionary was built for the specified partition of the table space. This message is issued during the RELOAD phase of the LOAD utility and at the end of the UNLOAD phase of the REORG utility.

The variable *n* is the number of dictionary entries: 512, 1024, 2048, 4096, or 8192. *n* depends on the number of rows being loaded or reorganized.

m is the number of rows used to build the dictionary.

System action: Processing continues.

Severity: 0 (informational)

DSNU242I   *csect-name* **KEEPDICTIONARY REQUESTED BUT COMPRESS ATTRIBUTE NOT DEFINED ON TABLE SPACE** *table-space-name*, **PARTITION** *part-num*

Explanation: Either the KEEPDICTIONARY option was specified in the LOAD or REORG TABLESPACE utility statement or the COPYDICTIONARY option was specified in a LOAD utility statement. These keywords should be used only with table space partitions that have the COMPRESS YES attribute. Otherwise, it is ignored. The COMPRESS column of SYSIBM.SYSTABLEPART contains the current compression status.

System action: Processing continues. If the dictionary exists, it is erased if either LOAD REPLACE or REORG is specified.

Severity: 4 (warning)

DSNU243I   *csect-name* **COMPRESSION DICTIONARY DOES NOT EXIST FOR TABLE SPACE** *table-space-name*, **PARTITION** *part-num*

Explanation: Either the KEEPDICTIONARY option was specified in the LOAD or REORG TABLESPACE utility statement or the COPYDICTIONARY option was specified in a LOAD utility statement for a table space partition that does not have a compression dictionary.

System action: Processing continues. If KEEPDICTIONARY was specified, a dictionary is built for the table space partition. If COPYDICTIONARY was specified, no dictionaries are copied.

Severity: 4 (warning)

DSNU244I   *csect-name* **COMPRESSION REPORT FOR TABLE SPACE** *table-space-name*, **PARTITION** *part-num*

Explanation: The data in the table space partition was compressed. This message is issued at the end of the RELOAD phase for the LOAD and REORG utility. The statistics that follow apply only to the data that was loaded or reorganized. For example, RESUME(YES) reports statistics only on the additional rows that were loaded, not what was previously in the table space.

The compression statistics follow:

noncmpbytes

The total number of uncompressed KB (where KB = 1024 bytes) for the loaded or reorganized data rows (including the row prefix). This number was rounded to the nearest KB.

cmpbytes

The total number of compressed KB for the loaded or reorganized data rows (including the row prefix). This number was rounded to the nearest KB.

percsave

The percentage of bytes saved from the loaded or reorganized data rows as a result of compression.

The values above reflect only the number of bytes for the data row and the respective row prefix (6 bytes). DB2 page and dictionary page overhead is not included.

pctrowcomp

The percentage of the rows that were compressed within the total number of rows that were loaded or reorganized. The stored length and the external length of a row was compared to determine if the row was compressed. If an EDITPROC is involved, its effect on the length of the row is considered in the calculation.

noncmppages

The estimated number of DB2 pages required by the uncompressed data rows that were loaded or reorganized. Space required for PCTFREE and FREEPAGE is included. This number was rounded up to the next integer.

cmppages

The estimated number of DB2 pages required by the compressed data rows that were loaded or reorganized. Space required for PCTFREE and FREEPAGE is included. Dictionary pages are also included unless a LOAD RESUME(YES) was run. This number was rounded up to the next integer.

auncmprowlen

Average uncompressed row length.

avgcmprowlen

Average compressed row length.

pagesave

An estimate of the percentage of DB2 pages saved by compressing the loaded or reorganized data rows.

System action: Processing continues.

Module:

noncmpbytes

KB WITHOUT COMPRESSION

cmpbytes

KB WITH COMPRESSION

percsave

PERCENT OF THE BYTES SAVED FROM COMPRESSED DATA ROWS

pctrowcomp

PERCENT OF THE LOADED ROWS WERE COMPRESSED

noncmppages

PAGES REQUIRED WITHOUT COMPRESSION

cmppages

PAGES REQUIRED WITH COMPRESSION

auncmprowlen

BYTES FOR AVERAGE UNCOMPRESSED ROW LENGTH

avgcmprowlen

BYTES FOR AVERAGE COMPRESSED ROW LENGTH

pagesave

PERCENT OF THE DB2 DATA PAGES SAVED USING COMPRESSED DATA

Severity: 0 (informational)

DSNU245I   *csect-name* **THE COMPRESSION DICTIONARY WAS NOT BUILT FOR TABLE SPACE** *ts-name*,
      PARTITION *partition-number reason-text*

Explanation: The compression dictionary could not be built.

csect-name
> The name of the control section that issued the message.

ts-name The name of the table space.

partition-number
> The partition number.

reason-text
> The reason that the dictionary could not be built:

> **blank**   If no reason is indicated, then insufficient data was available to build the compression dictionary.

>> **LOAD utility**
>>> Either the input data set is empty or all of the input records were discarded.

>> **REORG TABLESPACE utility**
>>> The partition is empty.

>> **Dictionary built during data insertion**
>>> The partition does not yet contain enough data.

> **CANNOT STORE DICTIONARY**
>> The dictionary was built but cannot be stored. This condition is usually the result of a communication error between data sharing group members. Optionally, you can:
>> - To build a compression dictionary on a subsequent insertion of data, use the DB2 START DATABASE command to reopen the table space or partition.
>> - To compress the data manually, use the REORG TABLESPACE utility.

System action: The data is not compressed.

Processing continues.

User response: Take the action that corresponds to the *reason-text* value:

blank

> **LOAD utility**
>> Make sure that the input data set contains data and that the input records are not discarded by the WHEN clause or because of conversion errors.

> **REORG TABLESPACE utility**
>> Make sure that the partition is populated with data.

> **Dictionary built during data insertion**
>> No action is required. The data remains uncompressed. Optionally, you can:
>> - To build a compression dictionary on a subsequent insertion of data, use the DB2 START DATABASE command to reopen the table space or partition.
>> - To compress the data manually, use the REORG TABLESPACE utility.

CANNOT STORE DICTIONARY
> Check that no communication error has occurred between data sharing group members.

Severity: 4 (warning)

Related reference:

➡ LOAD (DB2 Utilities)

➡ REORG TABLESPACE (DB2 Utilities)

DSNU246I *csect-name* **DICTIONARY FOR TABLE SPACE** *table-space-name*, **PARTITION** *part-num* **HAS BEEN ERASED.**

Explanation: The compression dictionary of the specified partition of the table space was erased.

The COMPRESS attribute of the table space partition is NO, and there are no active rows in this partition. In this case, the LOAD RESUME NO utility erases the dictionary.

System action: Processing continues.

Severity: 0 (informational)

DSNU247I *csect-name* **FROM TABLE SPACE** *table-space-name*, **PARTITION** *part-num1* **COPIED to PARTITION** *part-num2*

Explanation: A compression dictionary from partition *part-num1* was successfully copied to *part-num2* for the specified table space.

System action: Processing continues.

Severity: 0 (informational)

DSNU250I *csect-name* **UNLOAD PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The UNLOAD phase of the REORG utility completed normally. The time that elapsed between the beginning of the REORG utility and the completion of the UNLOAD phase was hh/mm/ss, hours, minutes, and seconds, respectively.

System action: Action subsequent to this message depends on the UNLOAD option of the REORG utility statement:
- If the UNLOAD option is CONTINUE, execution of the utility continues into the RELOAD phase.
- If the UNLOAD option is PAUSE, execution of the utility is stopped at the beginning of the RELOAD phase.
- If the UNLOAD option is ONLY or EXTERNAL, the REORG utility is finished. Processing continues to the next utility in the utility job step, or, if there are no more utility statements, the utility job step terminates normally.

Operator response: The operator's response depends on the UNLOAD option of the REORG statement:
- If the UNLOAD option is CONTINUE, no operator response is required. The execution of the REORG utility is proceeding.
- If the UNLOAD option is PAUSE, further processing is dependent upon restarting the utility job step with the RESTART(PHASE) option. Until that time, resources are held exclusively by the REORG utility job step.
- If the UNLOAD option is ONLY or EXTERNAL, the REORG utility processing is complete. The data set defined by the DD statement nominated by the UNLDDN option is available for installation disposition.

User response: The user's response depends on the UNLOAD option in the REORG utility statement.
- If the UNLOAD option is CONTINUE, no user response is required. The message is informational only.
- If the UNLOAD option is PAUSE, the REORG utility is stopped in the reload phase. To complete the REORG process, the user must explicitly restart the utility job step using the RESTART(PHASE) option.
- If the UNLOAD option is ONLY or EXTERNAL, the REORG utility has completed, and the requested output is on the data set identified by the DD statement of the UNLDDN REORG utility statement option.

Severity: 0 (informational)

DSNU251I *csect-name* **UNLOAD PHASE STATISTICS. NUMBER OF RECORDS UNLOADED=** *mmmmmm* **FOR** *obj-type obj-qual.obj-name* **PART** *nn*

Explanation: This message is issued at the completion of the UNLOAD phase of the REORG utility. It gives the number of records that were unloaded *mmmmmm* for partition *nn* of the tablespace, index, or indexspace that was specified. The count can indicate the amount of processing.

System action: Only valid table records are unloaded. Dropped table record counts are not reported.

Operator response: Count can be indicative of the amount of processing that remains.

User response: The record count represents the current cardinality of the table or index that was indicated in the error message. Consider invoking the RUNSTATS utility and then rebinding the involved PLANs if this value differs

greatly from the SYSTABLE.CARD value so that the SQL optimizer can bring the PLANs up to date.

Severity: 0 (informational)

DSNU252I *csect-name* **UNLOAD PHASE STATISTICS. NUMBER OF RECORDS UNLOADED=** *mmmmmmm* **FOR** *obj-type obj-qual.obj-name*

Explanation: This message is issued at the completion of the UNLOAD phase of the REORG utility. It gives the number of records that were unloaded *mmmmmm* for the tablespace, index, or indexspace that was indicated in the error message.

System action: Only valid table or index records are unloaded. Dropped table record counts are not reported.

Operator response: Count can be indicative of the amount of processing remaining.

User response: Record count represents the current cardinality of the table or index that was specified in the error message. Invocation of the RUNSTATS utility and a subsequent rebinding of the involved PLANs should be considered if this value differs greatly from the SYSTABLE.CARD value, so that the SQL optimizer can bring the PLANs up to date.

Severity: 0 (informational)

DSNU253I *csect-name* **UNLOAD PHASE STATISTICS. NUMBER OF RECORDS** *action=* *mmmmmmm* **FOR TABLE** *table-name*

Explanation: This message is issued at the completion of the UNLOAD phase of the REORG utility. If *action* is "UNLOADED", it gives the number of records *mmmmmm* that were unloaded for the specified table name. If *action* is "DISCARDED", it gives the number of records *mmmmmm* that were discarded for the specified table name.

System action: Table records were unloaded or discarded.

Severity: 0 (informational)

DSNU254I *csect-name* **- RECORD COUNTS INCONSISTENT FOR** *pppppppp* **PHASE, NUMBER OF RECORDS EXPECTED =** *nnnnnnnn*, **NUMBER OF RECORDS PROCESSED =** *aaaaaaaa*

Explanation: This message is issued during the *pppppppp* phase for the REORG utility to report the condition where the number of records, *nnnnnnnn*, passed to the current phase is not equal to the number of records, *aaaaaaaa*, processed in this phase.

System action: Processing continues if UNLOAD PAUSE option was used.

System programmer response: Determine the failing environment. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Check if unload data set was altered by user. Determine if the inconsistent record counts between the unload and reload phases is a potential problem. If it is an internal DB2 problem, report to system programmer.

Problem determination: If the unload data set was manipulated by user during restart process, make sure it was handled properly. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

• Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
• Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
• Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 4 (warning)

DSNU255I   *csect-name* **- KEY COUNT INCONSISTENT FOR** *utility-phase* **PHASE, NUMBER OF KEYS EXPECTED =** *expected-keys*, **NUMBER OF KEYS PROCESSED =** *processed-keys*.

Explanation: This message is issued during the *utility-phase* phase for the LOAD or REORG utility to report the condition where the number of keys, *expected-keys*, passed to the current phase is not equal to the number of keys, *processed-keys*, processed in this phase. One or more of the affected indexes might be inconsistent.

System action: Processing stops and all indexes are placed in rebuild pending state.

System programmer response: Determine the failing environment. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: TERM the utility and re-access the indexes by running REPAIR utility to reset the rebuild-pending states on indexes. Run the CHECK INDEX utility to determine if any of the affected indexes are inconsistent. Inconsistent indexes can be rebuilt by the REBUILD INDEX utility.

Problem determination: If the working data set was manipulated by the user during the restart process, make sure it was handled properly.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)
Related reference:

⮕ LOAD (DB2 Utilities)

⮕ REORG TABLESPACE (DB2 Utilities)

⮕ -TERM UTILITY (DB2) (DB2 Commands)

⮕ REORG INDEX (DB2 Utilities)

⮕ REPAIR (DB2 Utilities)

⮕ CHECK INDEX (DB2 Utilities)

⮕ REBUILD INDEX (DB2 Utilities)

DSNU256I   *csect-name* **INVALID PARTITION SPECIFICATION**

Explanation: An invalid PART option was specified.

csect-name
    The name of the control section that issued the message.

The PART option is invalid in the following situations:
- When it is specified for a nonpartitioned table space.
- If the indicated PART integer is not pertinent to the partitions existing for the indicated table space.
- If a ROWID column is part of the partitioning key.
- If LOAD INTO TABLE PART *x* is performed on a table space where an identity column is part of the partitioning key
- If LOAD INTO TABLE PART *x* is performed on a partition-by-growth table space
- If REORG PART *x* is performed on a partition-by-growth hash table space
- If LOAD INTO TABLE PART *x* REPLACE is performed on a table space that is being converted to hash table space
- When the table space is placed in the restrictive REORG-pending (REORP) state because a RECOVER job was run on the table space to recover the data to a point in time prior to the materialization of pending definition changes

System action: The error is sufficient to terminate further processing of the utility job step. The utility job step is terminated, and resources are released to their previous state.

User response: Determine the correct partition specification for the table space that was the subject of the specified control utility statement, and resubmit the utility job step. Examine the SYSIBM.SYSTABLESPACE in the DB2 catalog

| for the correct range of partition numbers for the table space.

| If a partition-level operation is not allowed, modify the utility job for table-space-level operations.

| **Related reference**:

| ➡ CHECK DATA (DB2 Utilities)

| ➡ LOAD (DB2 Utilities)

| ➡ QUIESCE (DB2 Utilities)

| ➡ REORG TABLESPACE (DB2 Utilities)

| ➡ REPAIR (DB2 Utilities)

| ➡ SYSIBM.SYSTABLESPACE table (DB2 SQL)

DSNU257I   *csect-name* **UNABLE TO RESTART BUILD PHASE**

Explanation: Restarting in the BUILD PHASE or the SORTBLD phase of the LOAD utility with the RESUME(YES) option is not permitted.

System action: Utility processing is not initiated.

User response: Recover all indexes within the table space which were left in a rebuild pending state when the LOAD utility failed.

Problem determination: Check the constraints for restarting the LOAD utility.

Severity: 8 (error)

Related concepts:

➡ Restart of LOAD (DB2 Utilities)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU258I   *csect-name* **BUILD PHASE STATISTICS. NUMBER OF INDEXES =** *mmm*

Explanation: This message is issued at normal completion of the BUILD phase of the LOAD or REORG or REBUILD INDEX utility. The number of indexes that were updated is indicated as 'mmm'.

System action: The BUILD phase of either the LOAD or REORG or REBUILD INDEX utility is completed. Upon completion of the utility job step, they are available for REFERENCE processing.

Severity: 0 (informational)

DSNU259I   *csect-name* **BUILD PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: This message is issued at the normal completion of the BUILD phase of the REORG, LOAD, and REBUILD INDEX utilities. The time that has elapsed between the beginning of the phase and the completion of this phase was 'hh' hours, 'mm' minutes, and 'ss' seconds.

System action: Processing continues to the next phase of the LOAD or REORG or REBUILD INDEX utility. If this is the last phase, processing continues to the next utility statement in the utility job step, if any.

DSNU260I   *csect-name obj-type obj-qual.obj-name* **IS EMPTY**

Explanation: This message is issued as a warning when REBUILD INDEX(ALL) or REORG TABLESPACE are executed on an empty table space.

This message is also issued when COPY detects that the first space map page in the table space or partition is missing. This is an error condition that might be caused by a delete and redefine of the data set outside of DB2, or by a previous DB2 failure that occurred on the data set.

System action: For REBUILD INDEX or REORG TABLESPACE, the return code of 4 will allow the next utility function in the job step, if any, to be initiated.

For COPY, the table space, index space, or partition being copied will be placed in recover pending status. The return code of 8 will prohibit further utility processing in the job step.

System programmer response: If the return code is 8, the table space, indexspace, or partition being copied should be recovered with the RECOVER utility.

User response: Notify the system programmer if the return code is 8. If the return code is 4, no action is required.

Severity: 4 (warning) or 8 (error)

DSNU269I   *csect-name* **FIELD** *field-name* **IS NOT ALLOWED**

Explanation: You have specified a GENERATED ALWAYS column in the field specification list. Only a column defined as GENERATED BY DEFAULT is allowed to be specified in the field specification list.

Identity columns or row change timestamp columns that were defined as GENERATED ALWAYS cannot be specified in a LOAD field specification list. Also, they cannot be implied in LOAD FORMAT UNLOAD or LOAD with no field specification list.

System action: The LOAD utility is terminated

User response: Omit the GENERATED ALWAYS column or identity column from the LOAD field specification list and rerun the job.

Severity: 8 (error)

DSNU270I   *csect-name* **-** *table-name*,*column-name* **IN VIOLATION OF INSTALLATION DEFINED FIELD**
              PROCEDURE *procedure-name*.*function-code*, *return-code*, *reason-code*, *message-token*

Explanation: An installation-supplied field procedure has returned an error during encoding for LOAD or decoding for REORG. SQLCODE -681 contains more detailed information.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

column-name
> The name of the column.

procedure-name
> The name of the stored procedure.

function-code
> The function code.

return-code
> A numeric value that indicates the reason for the failure.

reason-code
> A numeric value that indicates the reason for the failure.

message-token
> The message token.

System action: The utility issues a system '04E' abend with a reason code of '00E40336'.

System programmer response: Examine the field procedure to determine why an error return code was generated.

User response: Request the dump produced as a result of the error to the system programmer so that the cause of the error can be determined and corrected.

Severity: 8 (error)

DSNU271I • DSNU275I

DSNU271I   *csect-name* **- REORG PENDING ON FOR TABLE SPACE** *database.index-name* **PART** *part number* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a REORG utility to redistribute data in a partitioned table space. The partition number stated in this message was found to have the REORP state on, but was not specified on the PART *n1:n2* parameter of the REORG utility.

If you are running REORG to materialize identity column values, you must reorganize the entire table space.

System action: Utility processing is not initiated.

System programmer response: Identify the correct partition range specification for the REORG utility by issuing the DISPLAY command for the table space, and resubmit the utility job step.

Severity: 8 (error)

DSNU272I   *csect-name* **- FIELD PROCEDURE** *procedure-name* **COULD NOT BE LOADED**

Explanation: An installation supplied field procedure could not be loaded during encoding for LOAD or decoding for REORG. See SQLCODE -682 for more detail.

System action: The utility will terminate with a return code of 8.

System programmer response: Determine why the field procedure could not be loaded.

User response: Take the job output to the system programmer.

Severity: 8 (error)

DSNU274I   *csect-name* **- REORG SHRLEVEL REFERENCE WITH REBALANCE DOES NOT APPLY PENDING DEFINITION CHANGES FOR** *database-name.tablespace-name*.

Explanation: The object was reorganized, but no pending definition changes were materialized.

csect-name
>   The name of the control section that issued the message.

database-name.tablespace-name
>   The name of the table space, qualified by the name of the database.

System action: Utilities continue to process.

User response: Use the REORG utility with SHRLEVEL REFERENCE (without REBALANCE) or with SHRLEVEL CHANGE to materialize pending definition changes.

Severity: 4 (warning)

Related reference:

⏩ REORG TABLESPACE (DB2 Utilities)

DSNU275I   *csect-name* **- PENDING DEFINITION CHANGES ARE NOT APPLIED FOR INDEX,** *database-name.index-name* **BECAUSE THERE ARE PENDING DEFINITION CHANGES AT THE TABLESPACE LEVEL.**

Explanation: Pending definition changes at the index level were not applied because pending definition changes exist at the table space level. This message is issued in response to the REORG INDEX utility.

csect-name
>   The name of the control section that issued the message.

database-name.index-name
>   The name of the index, qualified by the name of the database.

System action: Utilities continue to process.

User response: Execute REORG TABLESPACE with SHRLEVEL REFERENCE or SHRLEVEL CHANGE against the entire table space to materialize pending definition changes at both the table space level and index level.

Severity: 4 (warning)

Related reference:

➥ REORG INDEX (DB2 Utilities)

DSNU280I   *csect-name* **- ERROR=X'reason-code' FROM EDIT ROUTINE** *procedure-name* **FOR** *table-name*

Explanation: The edit routine provided by your installation has returned an error during the load of a record into the table indicated. The reason code is set by the edit routine.

System action: The LOAD or REORG utility will receive ABEND04E RC00E40323.

System programmer response: Verify that the edit routine is correct.

User response: Correct any errors in the input data. Ensure that the edit routine provided is correct. The utility can be restarted or terminated.

Severity: 8 (error)

DSNU281I   *csect-name* **RECORD** (*n*) **OF TABLE** *table-name* **VIOLATING VALID ROUTINE** *validproc-name*, **ERROR CODE=X'reason-code'**

Explanation: The validation exit routine provided by your installation blocked the load of record 'n' into TABLE 'table-name'. The reason-code was set by the exit routine.

System action: The LOAD utility continues to process the following input record.

System programmer response: Verify that the validation routine is correct.

User response: Check the error input records against the error reason code returned from the validation procedure.

Severity: 4 (warning)

Related reference:

➥ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU282I   *csect-name* **UNABLE TO RESTART AT LAST CHECKPOINT WITH DDNAME** *'dddddddd'*

Explanation: The data set with DDNAME *dddddddd* is a concatenated data set and the utility can be restarted only at the beginning of the phase (RESTART PHASE), not at the last checkpoint (RESTART CURRENT).

System action: The utility will issue a system '04E' abend with a reason code of '00E40323'.

User response: Change the RESTART option to RESTART PHASE and resubmit the job step.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 8

DSNU283I *csect-name error-type* **SQLCODE =** *sqlcode*, **SQLERRM =** *sqlerrm*, **SQLSTATE =** *sqlstate*, **SQLERRP =** *sqlerrp*, **SQLERRD =** *sqlerrd1 sqlerrd2 sqlerrd3 sqlerrd4 sqlerrd5 sqlerrd6*

Explanation: An error occurred that returned SQLCA error information to the utility. This message contains multiple components.

csect-name
>   The name of the control section that issued the message.

error-type
>   Indicates the type of error:
>   - Trigger error
>   - LOB error
>   - XML error
>   - Key evaluation error
>   - Application of pending definition change

sqlcode The SQL return code. A negative number indicates that an error has occurred.

sqlerrm A length indicator and one or more tokens, separated by X'FF', that are substituted for variables in the descriptions of error conditions. It may contain truncated tokens. A length of 70 bytes might indicate a possible truncation.

sqlstate A return code for the outcome of the most recent execution of an SQL statement

sqlerrp The name of the procedure that detected the error.

sqlerrd1 **through** *sqlerrd6*
>   SQL diagnostic information. Each field contains a value that indicates the number of rows that qualified for the SQL statement, the number of rows in the result table, a rough estimate of required resources, or an internal error code.

System action: Processing terminates.

For the LOAD utility with a DISCARD data set, the rows that encounter the error are placed in the discard file. Otherwise the utility abends.

User response: Determine the cause for the SQL error by using information about the specific SQLCODE. Correct the error and run the utility again.

Severity: 8 (error)

Related concepts:

SQL error codes (DB2 Codes)

Related reference:

Description of SQLCA fields (DB2 SQL)

DSNU284I *csect-name*- **LOAD UTILITY NOT ALLOWED AGAINST A SYSTEM-MAINTAINED MATERIALIZED QUERY TABLE** *table-name*

Explanation: An attempt has been made to run the LOAD utility against a system-maintained materialized query table. This is not allowed.

System action: Utility processing terminates.

User response: If the table name was incorrect, correct the table name and resubmit the utility job step. Otherwise, take one of the following actions:
- Use the REFRESH TABLE SQL statement to insert data into the system-maintained materialized query table.
- Use the ALTER statement to change the materialized query table to a user-maintained materialized query table or a base table, and resubmit the utility job step.

Severity: 8

DSNU286I _csect-name_ **REORG TABLESPACE** _dbname.tsname_ **OFFPOSLIMIT SYSINDEXPART ROWS**

Explanation: This is the header message for the SYSINDEXPART section of the REORG TABLESPACE OFFPOSLIMIT output. This section lists the SYSINDEXPART rows for the table space or table space partition as specified in the REORG TABLESPACE input statement. The fields from SYSINDEXPART are:

CREATOR
>   Index creator name

IXNAME
>   Index name

CREATOR
>   Table creator name

TBNAME
>   Table name

PART   Partition number

CARDF
>   Number of rows referenced (floating point)

FAROFFPOSF
>   Number of rows referenced afar (floating point)

NEAROFFPOSF
>   Number of rows referenced nearby (floating point)

STATSTIME
>   TIMESTAMP of RUNSTATS update

Rows which meet the OFFPOSLIMIT criteria are prefaced with '*'.

System action: Processing continues normally

User response: If the TIMESTAMP reported in the STATSTIME column is not recent, the OFFPOSLIMIT criteria might be evaluated with obsolete statistics.

Severity: 0 (informational)

DSNU287I _csect-name_ **REORG TABLESPACE** _dbname.tsname_ **INDREFLIMIT SYSTABLEPART ROWS**

Explanation: This is the header message for the SYSTABLEPART section of the REORG TABLESPACE INDREFLIMIT output. This section lists the SYSTABLEPART rows for the table space or table space partition as specified in the REORG TABLESPACE input statement. The fields from SYSTABLEPART are:

DBNAME
>   Database name

TSNAME
>   Table space name

PART   Partition number

CARD   Number of rows in table space

FARINDREF
>   Number of rows relocated afar

NEARINDREF
>   Number of rows relocated nearby

STATSTIME
>   TIMESTAMP of RUNSTATS update

Rows which meet the INDREFLIMIT criteria are prefaced with '*'.

System action: Processing continues normally

User response: If the TIMESTAMP reported in the STATSTIME column is not recent, the INDREFLIMIT criteria might be evaluated with obsolete statistics.

Severity: 0 (informational)

DSNU288I   *csect-name* **REORG INDEX** *dbname.ixname* **LEAFDISTLIMIT SYSINDEXPART ROWS**

Explanation: This is the header message for the SYSINDEXPART section of the REORG INDEX LEAFDISTLIMIT output. This section lists the SYSINDEXPART rows for the index or index partition as specified in the REORG INDEX input statement. The fields from SYSINDEXPART are:

CREATOR
>   Index creator name

IXNAME
>   Index name

PART   Partition number

LEAFDIST
>   Number of pages between leaf pages multiplied by 100

STATSTIME
>   TIMESTAMP of RUNSTATS update

Rows which meet the LEAFDISTLIMIT criteria are prefaced with '*'.

System action: Processing continues normally

User response: If the TIMESTAMP reported in the STATSTIME column is not recent, the LEAFDISTLIMIT criteria might be evaluated with obsolete statistics.

Severity: 0 (informational)

DSNU289I   *csect-name type* **REORG LIMITS HAVE BEEN MET**

Explanation: The REORG utility is indicating whether any specified limits have been met. For REORG TABLESPACE, this might be the OFFPOSLIMIT or the INDREFLIMIT. For REORG INDEX, this is the LEAFDISTLIMIT. *type* values:

NO   If no limits are met.

blank   If at least one limit is met.

System action: Utility processing continues.

User response: If the REPORTONLY option is specified and the result of this job step is not used to conditionally execute other job steps, examine the utility messages for this table space, partition, or index to determine if a REORG should be performed.

Severity: The return codes are:

1 (informational)
>   If no limit is met.

2 (informational)
>   If at least one limit is met.

DSNU290I   *csect-name type* **REORG WILL BE PERFORMED**

Explanation: The REORG utility is indicating whether any specified limits have been met and whether a REORG will be performed. For REORG TABLESPACE, this might be the OFFPOSLIMIT or the INDREFLIMIT. For REORG INDEX, this is the LEAFDISTLIMIT. *type* values:

NO   If no limits are met.

blank   If at least one limit is met.

System action: Utility processing continues. A REORG is performed if any limits are met.

Severity: 0 (informational)

DSNU291I   MAXIMUM COMPOSITE UNLOAD RECORD SIZE TOO LARGE FOR SORT. SORTDATA KEYWORD IS IGNORED.

Explanation: The SORTDATA keyword was specified for a REORG utility job. However, the largest possible record in the table space, prefixed by the largest key, produces a composite record that is too large to be processed by the SORT component. The SORTDATA keyword is ignored.

System action: Processing continues normally.

User response: If you must use SORTDATA for the table space, try reducing the size of either the largest record or key field by redefining the table. In some cases, changing a field from LONG VARCHAR to VARCHAR with a specified maximum length might help.

Problem determination: This problem should only occur on a table space with a 32KB page size. Analyze the requirement for maximum data length in VARCHAR and LONG VARCHAR columns to see if a shorter length can be specified. It might be necessary to remove the SORTDATA keyword from the utility statement.

Severity: 0

DSNU292I   *csect-name* **- -ALTER UTILITY CANNOT APPLY TO THIS UTILITY, REASON=***c*

Explanation: The -ALTER UTILITY command applies only to a job that is executing REBUILD with SHRLEVEL CHANGE, REORG with SHRLEVEL REFERENCE, or REORG with SHRLEVEL CHANGE. In addition, if the -ALTER UTILITY command specifies MAXRO, LONGLOG, or DELAY, the job must be executing REBUILD with SHRLEVEL CHANGE, or, REORG with SHRLEVEL CHANGE.

The reason code indicates the specific type of error:

1   The job to which the -ALTER UTILITY command applies is not executing REBUILD with SHRLEVEL CHANGE, REORG with SHRLEVEL REFERENCE, or REORG with SHRLEVEL CHANGE.

2   The job to which the -ALTER UTILITY command applies is executing REORG with SHRLEVEL REFERENCE, but the -ALTER UTILITY command specifies MAXRO, LONGLOG, or DELAY.

3   The job to which the -ALTER UTILITY command applies is executing REBUILD with SHRLEVEL change, but the -ALTER UTILITY command specifies DEADLINE.

System action: -ALTER UTILITY does not execute.

User response: Apply -ALTER UTILITY to a job that is executing the proper utility.

Severity: 8 (error)

DSNU293I   *csect-name* **- REORG WITH SHRLEVEL REFERENCE OR CHANGE CANNOT APPLY TO THE** *name* **TABLE SPACE**

Explanation: REORG with SHRLEVEL REFERENCE or CHANGE cannot operate on the following catalog and directory table spaces: DSNDB06.SYSDBASE, DSNDB06.SYSDBAUT, DSNDB06.SYSGROUP, DSNDB06.SYSPLAN, DSNDB06.SYSVIEWS, and DSNDB01.DBD01.

System action: REORG does not execute.

User response: Execute REORG with SHRLEVEL NONE.

Severity: 8 (error)

DSNU294I   *csect-name* **- LARGEST COMPOSITE RECORD IS TOO BIG FOR REORG WITH SHRLEVEL CHANGE**

Explanation: SHRLEVEL CHANGE was specified for a REORG utility job, and the table has a clustering index. However, the largest possible record in the table space, prefixed by the largest key, produces a composite record that is too large to be processed by the SORT component. If the largest possible composite record to be sorted exceeds 32760 bytes in length, which is the maximum record size for a BSAM data set, REORG cannot operate if SHRLEVEL CHANGE is specified.

System action: REORG does not execute.

DSNU295I • DSNU296I

User response: Execute REORG with SHRLEVEL NONE or REFERENCE and specify SORTDATA NO. If you must use SHRLEVEL CHANGE for the table space, try reducing the size of either the largest record or key field by redefining the table. In some cases, changing a field from LONG VARCHAR to VARCHAR with a specified maximum length might help.

Problem determination: This problem should only occur on a table space with a 32KB page size. Analyze the requirement for maximum data length in VARCHAR and LONG VARCHAR columns to see whether a shorter length can be specified.

Severity: 8 (error)

Related reference:

▶ Data sets that REORG INDEX uses (DB2 Utilities)

▶ Data sets that REORG TABLESPACE uses (DB2 Utilities)

DSNU295I *csect-name* **- IN UTILTERM PHASE OF REORG WITH SHRLEVEL REFERENCE OR CHANGE, DATA SET** *name* **COULD NOT BE DELETED**

Explanation: For the area (table space or partition) being reorganized, REORG with SHRLEVEL REFERENCE or CHANGE unloads data from the original data sets and reloads data into shadow data sets. REORG then exchanges the names of the original and shadow data sets. The original name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.I0001.A*nnn*". The shadow name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.J0001.A*nnn*". REORG uses such data sets for indexes as well as data.

If a data set is DB2-managed, the utilterm phase of REORG deletes the shadow data set, which was the original data set. This message indicates that the shadow data set *name* could not be deleted.

System action: REORG continues execution, but it does not delete the specified data set.

User response: Try to delete the data set. If you cannot delete it, notify the system programmer.

Severity: 4 (warning)

DSNU296I *csect-name* **- IN REORG WITH SHRLEVEL REFERENCE OR CHANGE, DATA SET** *name* **DOES NOT EXIST, CONTEXT=***c*

Explanation: For the area (table space or partition) being reorganized, REORG with SHRLEVEL REFERENCE or CHANGE unloads data from the original data sets and reloads data into shadow data sets. REORG then exchanges the names of the original and shadow data sets, using temporary names during the renaming. The original name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.I0001.A*nnn*". The shadow name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.J0001.A*nnn*". The temporary name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.T0001.A*nnn*". REORG uses such data sets for indexes as well as data.

During its initialization, REORG cannot execute if:
- Data sets with the temporary names already exist
- DB2-managed data sets with the shadow names already exist
- User-managed data sets with the shadow names do not already exist

To exchange the names of the original and shadow data sets, REORG renames in the following ways:

1   In the switch phase, it renames the original data set from the original name to the temporary name.

2   In the switch phase, it renames the shadow data set from the shadow name to the original name.

3   In the utilterm phase, it renames the original data set from the temporary name to the shadow name (for user-managed data sets). It deletes the temporary name for DB2-managed data sets.

If either a -TERM UTILITY command is issued, or the REORG switch phase is restarted after REORG has begun renaming data sets, the names revert back to their original form prior to the utilterm phase of REORG:

The context code indicates the specific type of error:

1

During the switch phase of REORG, on rename 1, REORG detected that a data set with an original name does not exist. This error condition can arise for a DB2-managed or user-managed data set.

During the switch phase restart or a -TERM UTILITY during the switch phase, on rename 4, REORG detected that a data set with an original name does not exist. This error condition can arise for a DB2-managed or user-managed data set.

2

During the switch phase of REORG, on rename 2, REORG detected that a data set with a shadow name does not exist. This error condition can arise for a DB2-managed or user-managed data set.

During the switch phase restart or a -TERM UTILITY during the switch phase, on rename 5, REORG detected that a data set with a shadow name does not exist. This error condition can arise for a DB2-managed or user-managed data set.

3   During the utilterm phase of REORG, on rename 3, REORG detected that a data set with a temporary name does not exist. This error condition can arise for a user-managed data set.

7   During initialization of REORG, REORG detected that the shadow data set did not exist. This error can only occur with a table space, index, or partition that is stored in user-managed data sets.

System action: For context codes 1 or 2, utility processing is abended. ABEND reason code 00E4031B is issued. The object is left in a recovery pending state.

For context code 3, the utility job step that caused the function to be requested continues with no renaming. The utility job is placed in the stopped state. The object is left in a recovery pending state, and a message will be issued.

For context code 7, REORG does not execute.

For context codes 1, 2, or 3, the following messages are issued:
- If the data set is part of a nonpartitioned table space, the object is the table space, and the message is DSNU562I.
- If the data set is part of a partition of a table space, the object is the partition, and the message is DSNU560I.
- If the data set is part of a nonpartitioned index, the object is the index, and the message is DSNU557I.
- If the data set is part of a partition of a partitioned index, the object is the partition of the index, and the message is DSNU559I.

User response: For context codes 1 or 2, execute the RECOVER TABLESPACE or RECOVER or REBUILD utilities.

For context code 3, do nothing.

For context code 7, define all shadow objects prior to running REORG.

Severity: 4 (warning) or 8 (error)

DSNU297I   *csect-name* **COMPOSITE RECORD SIZE TOO LARGE FOR** *table-name*

Explanation: The size of the record that would be unloaded for REORG DISCARD or REORG UNLOAD EXTERNAL is greater than the largest allowed data set logical record length (32760 bytes).

System action: The utility will terminate with a return code of 8.

User response: If you are attempting to discard records during REORG, you will have to use SQL to DELETE the records prior to the REORG. If you are attempting to unload records in an external format, you must use an application instead. If you are using the SORTDATA option with REORG UNLOAD EXTERNAL or REORG DISCARD, try removing the SORTDATA keyword.

Problem determination: This problem will most likely occur only on a table with BLOB, CLOB, or DBCLOB columns.

Severity: 8 (error)

DSNU298I   *csect-name* **- RESTART NOT ALLOWED IN** *phase-name* **PHASE OF REORG WITH SHRLEVEL REFERENCE OR CHANGE**

Explanation: An attempt was made to restart the REORG utility with SHRLEVEL REFERENCE or SHRLEVEL CHANGE in the *phase-name* phase. DB2 does not allow you to restart the REORG utility with SHRLEVEL REFERENCE or SHRLEVEL CHANGE in that phase.

DSNU299I

System action: Utility processing abends. Abend reason code 00E40018 or 00E40310 might be issued.

System programmer response: If necessary, take actions to prevent access to the table space or partition before terminating the utility.

User response: Issue the -TERM UTILITY command to terminate the utility job; then rerun the job without the RESTART parameter.

Severity: 8 (error)

DSNU299I   *csect-name* **CHARACTER CONVERSION FROM CCSID** *from-ccsid* **TO CCSID** *to-ccsid* **FAILED WITH ERROR CODE** *error-code* **FOR** *object-type object-qualifier.object-name additional-information***.**

Explanation: A conversion error occurred during the conversion of a string to a different coded character set. The error occurred during the processing of the utility statement.

csect-name
> The name of the control section that issued the message.

from-ccsid
> The coded character set of the string to be converted.

to-ccsid   The coded character set to which the string must be converted.

error-code
> The type of error:

> | | |
> |---|---|
> | **4** | Conversion resulted in the use of incompatible substitution characters in the target of the conversion. |
> | **8** | Length exception, such as expansion required for PC MIXED data that exceeds the maximum length of the string. |
> | **12** | Invalid code point. |
> | **16** | Form exception, such as invalid MIXED data. |
> | **20** | Conversion procedure error, such as an error during Unicode/390 stage 2 conversions or an exit that set the length control field of the string to an invalid value. |
> | **24** | Conversion services have not been setup correctly or the conversion is not supported. |

object-type
> The type of the failed object.

> An *object-type* of UNKNOWN indicates that a failure has occurred in the conversion service. DB2 reason code 00E400D0 is issued, and provides additional information.

object-qualifier.object-name
> The table creator ID followed by the table name. If the object is a user security label or row security label, only *object-name* is shown.

additional-information
> Additional information about the table, which is provided only when *object-type* is TABLE.

> **COLUMN** *column-name*
> > The column name of the failing object. This column could be either the target or the source of the conversion.

> **blank**   No additional information is provided for other values of *object-type*.

System action: The utility job is terminated.

User response: Generate a dump by adding the DIAGNOSE ABEND statement in the utility job and re-running the utility. If the error can be fixed, correct the error and resubmit the utility. Otherwise, issue the TERM utility command to terminate the utility.

If the error is related to security labels, use only security labels that characters that are not substituted during conversion from one CCSID to another. For example, the following national characters might be substituted:
- @ (X'7C')
- # (X'7B')

- $ (X'5B')

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

12 (severe error)

Related information:

↪ 00E400D0 (DB2 Codes)

DSNU300I   *csect-name* **(RE)LOAD PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The reload phase of the REORG or LOAD utility has completed normally. The elapsed time for the execution was 'hh' hours, 'mm' minutes, and 'ss' seconds.

DSNU301I   *csect-name* **- KEYWORD 'SORTDATA' SPECIFIED AND/OR KEYWORD 'NOSYSREC' WAS SPECIFIED BUT NO CLUSTERINGINDEX EXISTS, KEYWORD IS IGNORED**

Explanation: The 'SORTDATA' keyword was specified or the 'NOSYSREC' keyword was specified for a REORG utility job, or implied by the use of SHRLEVEL CHANGE. But it was found that no clustering index existed for the table space being reorganized. The keyword is ignored and processing continues normally.

System action: Processing continues normally.

User response: If a clustering index was expected to be found for the table space being reorganized, determine the reason the index could not be found.

Problem determination: Probably a user error. Determine if a cluster index should be defined on one of the tables in the table space being reorganized. If so, define the cluster index; otherwise, remove the 'SORTDATA' and/or 'NOSYSREC' keyword from the utility statement if necessary.

Severity: 0

DSNU302I   *csect-name* **(RE)LOAD PHASE STATISTICS - NUMBER OF INPUT RECORDS PROCESSED =** *nnnnnnnn*

Explanation: This message is issued at the normal completion of the RELOAD phase of the REORG utility and the LOAD phase of the LOAD utility.

It indicates the number of records processed from the input record data set (that is, the data set identified either by the JCL DD statement identified in the UNLDDN option for REORG or the INDDN option for LOAD).

System action: The LOAD or REORG utility completes the phase in which the records are (re-)introduced into the DB2 database. This is also the last phase of the utility unless there are some indexes that need to be built and/or updated.

User response: The user can verify that the number of records (re-)loaded is consistent with expectations. For example, in REORG it matches the total number of records unloaded. For LOAD, it matches the number of records submitted minus the number of records reported rejected.

Severity: 0 (informational)

DSNU303I   *csect-name* **(RE)LOAD PHASE STATISTICS - NUMBER OF RECORDS=** *nnnnnn* **FOR TABLE** *table-name* **PART=***part-number*

Explanation: This message is issued at the normal completion of the RELOAD phase of the REORG utility and the LOAD phase of the LOAD utility.

It indicates the number of records loaded into PART 'part-number' of 'table-name' from the source record data set. This data set is identified by the JCL DD statement in the UNLDDN option for REORG, or the INDDN option for LOAD.

This message is repeated for each INTO TABLE PART clause in the LOAD utility statement or each TABLESPACE PART clause in the REORG utility statement.

System action: The LOAD or RELOAD phase maintains a separate count for each table part. Rejected rows are not included in the count. Each count is reported as a separate message at the end of the LOAD or RELOAD phase.

The user can verify the number of records loaded or reloaded into the table part. In REORG, it should match the number of records unloaded for that table part. In LOAD, it should match the number of records processed.

Severity: 0 (informational)

DSNU304I   *csect-name* **(RE)LOAD PHASE STATISTICS - NUMBER OF RECORDS =** *nnnnnn* **FOR TABLE** *table-name*

Explanation: This message is issued at the normal completion of the RELOAD phase of the REORG utility and the LOAD phase of the LOAD utility.

It indicates the number of records loaded into TABLE 'table-name' from the source record data set (that is, the data set identified by the JCL DD statement identified in the UNLDDN option for REORG, or the INDDN option for LOAD).

This message is repeated for each TABLE loaded by the current REORG or LOAD utility statement.

If the LOAD or REORG was restarted in a LOAD or RELOAD phase from the last commit point (i.e., the 'RESTART' parm was specified), 'nnnnnn' indicates the number of records loaded after the restart.

System action: The LOAD or RELOAD phase maintains a separate count for each table populated by this phase. Rejected rows are not included in the count. Each count is reported as a separate message at the end of the LOAD or RELOAD phase.

The user can verify that the number of records (re-)loaded into the table is consistent with expectations. For example, in REORG it matches the number of records unloaded for that table. For LOAD, it matches the number of records processed.

Severity: 0 (informational)

DSNU305I   *csect-name* **- TABLE=***table-name* **HAS AN INCOMPLETE TABLE DEFINITION**

Explanation: Table 'table-name' does not have an index defined to enforce the uniqueness of its primary key or unique key.

System action: Processing stops. No table in the table space is loaded.

User response: The user can resubmit the LOAD job after performing one of the following:
- Creating a unique index over the table's primary key
- Creating a unique index over the table's unique key(s)
- Dropping the definition of the table's primary key
- Dropping the definition of the table's unique key(s)

Problem determination: Issue SQL SELECT against the catalog to determine which indexes and relations exist against 'table-name'. Determine whether uniqueness of the primary key or unique key has been guaranteed through omission or through faulty definition of an index.

Severity: 8 (error)

DSNU306I   *csect-name* **- UNABLE TO DRAIN WRITERS FROM** *object-type object-name*

Explanation: The utility was unable to drain the writers from an object because the resource was unavailable. The object is identified by type and name. Message DSNT500I tells why the resource was unavailable. If the QUIESCE utility issues this message, DSNT500I is not issued, but message DSNT501I might be issued.

System action: Utility processing terminates.

Problem determination: Refer to message DSNT500I or DSNT501I.

Severity: 8 (error)

DSNU307I *csect-name* **PARENT TABLE** *parent-table-name* **OF** *table-name* **IN RELATIONSHIP** *relationship-name* **HAS AN INCOMPLETE TABLE DEFINITION**

Explanation: The 'table-name' identifies the target table of the job for the LOAD utility or the dependent table to be checked by the CHECK DATA utility. The 'parent-table-name' identifies the parent table of the target table for the LOAD utility or the dependent table for the CHECK DATA utility in RELATIONSHIP 'relationship-name'. The primary index of PARENT TABLE 'parent-table-name' has been dropped.

This message is repeated for each incompletely defined parent table in each relationship where the target table is a child.

System action: The LOAD and CHECK DATA utilities stop further processing. No data is loaded by the LOAD utility and no relationships are checked by the CHECK DATA utility.

User response: Check the existence of the primary index and its parent tables identified by RELATIONSHIP 'relationship-name'.

Problem determination: If the incompletely defined tables were targets of the LOAD job, message DSNU305I has been issued for each such table. Refer to the catalog to determine if any tables in the table space that were not targets of the LOAD job are incompletely defined.

Severity: 8 (error)

DSNU308I *csect-name* **INDEX KEY LIST RECORD LENGTH EXCEEDS MAXIMUM KEY BUFFER SIZE**

Explanation: The length of the index key list that will be built in the Data Base Services address space while loading a data record may be larger than the maximum key buffer size. The LOAD statement may be too complex.

System action: Utility processing is terminated.

User response: Examine the preceding DSNU650I INTO TABLE statements. The key size and number of indexes affected by each INTO TABLE statement are accumulative. The combination of these elements may produce a list of index keys, including header bytes, that is too large to be contained within the maximum size buffer area during the load of each input data record. It may be necessary to reduce the number of INTO TABLE statements or to drop some of the indexes that are affected by the load.

Severity: 8 (error)

DSNU309I *csect-name* **NOT ALL REQUIRED UNIQUE INDEXES HAVE BEEN DEFINED FOR TABLE** *table-name*

Explanation: The table 'table-name' was created with unique index requirements. Not all of these indexes have been defined.

System action: Processing stops. No table in the table space is loaded.

User response: The user may resubmit the LOAD job after either:
• Creating all the required unique indexes, or
• Dropping the table and re-creating it without the unique index requirements.

Problem determination: Determine which unique indexes need to be created. Determine if the uniqueness requirement for the table is valid.

Severity: 8

DSNU310I *csect-name* **RECORD 'n' WILL BE DISCARDED DUE TO** *field-name* **CONVERSION ERROR FOR** *table-name* **ERROR CODE** *cc*

Explanation: The load utility found the input data provided for a field to be incorrect. The corresponding input data set record 'n' will be placed in the DISCARDS data set.

If the NOSUBS keyword was specified, it is possible that the record was discarded because the input contained a code point that when converted from the source CCSID to the target CCSID resulted in a substitution character being placed in the target string.

The error code, *cc*, identifies the problem. See DSNU334I for the list of error codes and their meaning.

System action: The input data set record is noted and is discarded in the DISCARD phase. The LOAD utility

continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set.

Severity: 4 (warning)

Related reference:

▶ Syntax and options of the LOAD control statement (DB2 Utilities)

Related information:
"DSNU334I" on page 674

DSNU311I *csect-name* **RECORD '*n*' WILL BE DISCARDED DUE TO INVALID** *field-name* **COLUMN SPECIFICATION**

Explanation: The load utility calculated the location of an input data value using the field specifications supplied and VARCHAR length fields present within the input data set record. The calculated position for the field was determined to be incorrect.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data, if one exists. Otherwise, correct the input field specification. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

▶ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU312I *csect-name* **RECORD '*n*' WILL BE DISCARDED DUE TO ERROR = X'***reason-code***' FROM EXIT ROUTINE** *procedure-name* **FOR** *table-name*

Explanation: The exit routine provided by your installation blocked the load of record 'n' into the table indicated. The reason-code was set by the exit routine.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. error is placed in the DISCARDS data set. The LOAD utility continues to process additional records in the input data set.

System programmer response: Verify that the validation routine is correct.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data, if one exists. Ensure that the exit routine provided is correct. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

▶ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU313I *csect-name* **RECORD '*n*' WILL BE DISCARDED DUE TO PART OR HIGH KEY VALUE SPECIFICATION(S)**

Explanation: The discarded record does not belong to any of the partitions being loaded. For a LARGE table space, this might be because the key is higher than the high key value specified for the last partition.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

System programmer response: Verify that the validation routine is correct.

User response: Following the completion of the current LOAD utility invocation, examine the record in the

DISCARDS data set. Correct the error in the discarded input data, if one exists. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set.

Severity: 4 (warning)

Related reference:

 LOAD (DB2 Utilities)

 Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU314I   *csect-name* **RECORD 'n' WILL BE DISCARDED DUE TO WHEN SPECIFICATION(S)**

Explanation: The discarded record satisfies none of the WHEN selection expressions provided.

System action: The input data set record is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data, if one exists. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set. If FORMAT UNLOAD was specified on the load statement, the WHEN specifications are built internally. This specification ensures that the rows are loaded back into the exact same tables from which the data was unloaded.

Severity: 4 (warning)

Related reference:

 Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU315I   *csect-name* **RECORD '*n*' WILL BE DISCARDED DUE TO UNEXPECTED ERROR**

Explanation: The LOAD of input data set record number *n* was attempted but was not successful due to an unexpected error.

System action: If the error was caused because a resource was not available, a DSNT500I message will be issued.

The utility step will be abended with a system X'04E' and an abend reason code of '00E40318'. If a DSNT500I message is not issued, an SVC dump will be requested. A SYSABEND dump of batch storage is also requested.

Operator response: An operator with SYSOPR authority can issue a TERM UTIL to free any held resources if that is necessary.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: If a DSNT500I message was issued, refer to the description under that message number. If the unavailable resource becomes available, the utility step is restartable.

If there is no DSNT500I message, the error is an unexpected internal error and you should consult your system programmer.

Severity: 8 (error)

Related reference:

 -TERM UTILITY (DB2) (DB2 Commands)

Related information:

"DSNT500I" on page 555

 00E40318 (DB2 Codes)

DSNU316I   *csect-name* **- UNABLE TO DRAIN WRITERS FROM** *object-type object-name* **PART** *part-num*

Explanation: The utility was unable to drain the class of writers from a partition of an object because the resource was unavailable. The partition is identified by object type (object-type), object name (object-name), and partition number (part-num). Message DSNT500I tells why the resource was unavailable.

System action: Utility processing terminates.

Problem determination: Refer to message DSNT500I.

Severity: 8 (error)

DSNU317I *csect-name* **DISCARD LIMIT HAS BEEN REACHED**

Explanation: The DISCARDS limit specified on the LOAD utility statement was reached.

System action: Processing abends to allow users to increase the discard limit and restart the utility.

User response: Increase the discard limit and restart the utility. Otherwise -TERM the utility. Review the file of input records provided to LOAD and review table definitions to determine why the number of records discarded exceeds expectations.

Problem determination: See user response.

Severity: 8 (error)

DSNU318I *csect-name* **RECORD '*n*' WILL BE DISCARDED DUE TO INVALID** *field-name* **COLUMN SPECIFICATION FOR** *table-name*

Explanation: The load utility calculated the location of an input data value using the field specifications supplied and VARCHAR length fields present within the input data set record 'n'. The calculated position for the field was determined to be incorrect.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data, if one exists. Otherwise, correct the input field specification. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

↪ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU319I *csect-name* **RESUME(YES) WAS SPECIFIED FOR EMPTY PARTITION** *nn*. **RESUME(NO) IS ASSUMED**

Explanation: This message is issued in response to a LOAD utility request containing a partition level RESUME(YES) option. The RESUME(YES) option implies that the specified partition 'nn' is populated, but in this case it was not. The message alerts the invoker to this fact, and continues as if RESUME(YES) was not specified.

System action: The LOAD utility, in attempting to honor the RESUME(YES), found the partition was empty, ignored the option, and continued as if RESUME(NO) was specified.

User response: The user is alerted to the fact that the requested partition was empty rather than populated as the RESUME LOAD option implied.

Severity: 4 (warning)

DSNU320I *csect-name* **RESUME(YES) WAS SPECIFIED FOR AN EMPTY TABLESPACE**

Explanation: RESUME(YES) was specified within the LOAD utility statement. However, the RESUME(YES) option implies that the specified table space is populated.

System action: Processing continues. The LOAD utility, in attempting to honor the RESUME(YES), found the table space empty, so ignored the option, and proceeded as if RESUME(NO) was specified. If this table is currently defined as COMPRESS YES, a dictionary is not built, unless the tablespace is empty, in which case a compression dictionary will be built.

User response: The user is alerted to the fact that the requested table space was empty rather than populated as the RESUME LOAD option implied. Use the RESUME(NO) option if you want a compression dictionary built.

Severity: 4 (warning)

DSNU321I *csect-name* **CHECK PENDING ON TABLESPACE** *tablespace-name* **PROHIBITS PROCESSING**

Explanation: The requested utility cannot be run against a tablespace or tablespace partition in a check pending state.

System action: Processing terminates after the check pending states of all target tablespaces or partitions of the utility are analyzed.

User response: Remove the named tablespace or partition from its check pending state by using the CHECK DATA or REPAIR utility. If you want more information on the extent of questionable data in the table space, query the SYSTABLES and SYSTABLEPART catalog tables.

Severity: 8 (error)

DSNU322I *csect-name* **DEFERRED STATUS OF TABLE SPACE** *database-name.table-space-name* **PROHIBITS PROCESSING**

Explanation: The requested utility cannot be run against a table space or table space partition in a deferred state.

System action: Processing terminates after the deferred state of all target table spaces or table space partitions of the utility are analyzed.

User response: To remove the deferred state, use the RECOVER TABLESPACE, LOAD REPLACE utility, or -START DATABASE command with ACCESS(RW) or ACCESS(RO). If you want more information on the status of the table space, use the -DISPLAY DATABASE command.

Severity: 8 (error)

DSNU323I *csect-name* **COLUMN** *ccc* **IS OMITTED**

Explanation: The column 'ccc' is a nonnullable column and is not being loaded. Field specifications must include all nonnullable columns. The identified column name was missing from the field specification.

System action: Further utility processing is terminated.

System programmer response: Corrective procedures may require that a table be dropped and created again.

User response: Determine if the input data contains the nonnullable column as a field. If it does, change the field specification to include it. If the input data does not contain the nonnullable column as a field, determine which is in error, the input data or the column specification. Make the corresponding correction. Resubmit the job if that is still appropriate.

Severity: 8 (error)

Related tasks:

↪ Changing data types by dropping and re-creating the table (DB2 Administration Guide)

Related reference:

↪ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU324I *csect-name* **DEFERRED STATUS OF INDEX SPACE** *creator-name.index-name* **PROHIBITS PROCESSING**

Explanation: The requested utility cannot be run against an index space or index space partition in a deferred state.

System action: Processing terminates after the deferred state of all target index spaces or index partitions of the utility are analyzed.

User response: The deferred state can be removed through recovery or rebuild of the index space or the physical index partition. It can also be removed by the -START DATABASE command with ACCESS(RW) or ACCESS(RO). The RECOVER or REBUILD INDEX utility can be run at either the index space or index partition level. If more information on the status of the index space is required, use the -DISPLAY DATABASE command.

Severity: 8 (error)

DSNU325I   *csect-name* **- AN INVALID SQL/DS FORMAT RECORD WAS ENCOUNTERED**

Explanation: The LOAD utility detected an invalid SQL/DS format record type.

System action: Utility execution is terminated.

User response: Correct the input data and resubmit the job. Data that has been transmitted from an SQL/DS unload disk data set in a VM/370 system may have extra block descriptors and record descriptors at the beginning of each record. Ensure that this situation does not exist.

Severity: 8 (error)

Related concepts:

▶ What to do about utilities that were in progress at time of failure (DB2 Administration Guide)

▶ Restart of LOAD (DB2 Utilities)

Related reference:

▶ LOAD (DB2 Utilities)

▶ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU326I   *csect-name* **- USER NOT AUTHORIZED FOR LOAD REPLACE ON TABLESPACE**

Explanation: The user attempted to use the REPLACE option of LOAD, and it will reset to empty all tables in the table space before loading data. This option requires load authorization on each table in the table space, but the user did not have authority for one or more tables. In addition, if a table in the table space has a security label column, the user must have the write-down privilege when write-down is in effect.

System action: When LOAD REPLACE validation encounters the first non-authorized table in the table space, the job step is terminated.

User response: Obtaining the LOAD on DATABASE privilege for the database will allow you to proceed using the REPLACE option. If the table space is an implicit database, then obtain the appropriate authorization privileges on the implicit database or DSNDB04. If a table in the table space has a security label column, ensure that the user has write-down privilege when write-down is in effect before using the REPLACE option. The other alternative is to first delete all rows from the tables which are to be loaded, and then use the RESUME option instead of the REPLACE option.

Severity: 12 (severe error)

DSNU327I   *csect-name* **TABLE** *b* **IS NOT IN SAME TABLESPACE AS TABLE** *a*

Explanation: All tables that are the subject of a single LOAD utility statement must belong to the same table space.

System action: Further utility processing is terminated.

User response: Remove the INTO TABLE control statement that specifies TABLE 'b' from the current LOAD request. If the data is to be loaded into TABLE 'b', create another LOAD utility statement that contains the INTO TABLE control statement for TABLE 'b' and resubmit the job.

Severity: 8 (error)

DSNU328I   *csect-name* **CANNOT PROCESS THE NON-EMPTY PARTITION** *nn* **WITH RESUME(NO). PROCESSING TERMINATES**

Explanation: The partition level RESUME keyword either specified or defaulted to a value of NO, which requires the partition to be empty. However, the LOAD command named partition 'nn', which was not empty.

System action: Further utility processing is terminated.

System programmer response: Determine whether the partition should be empty. If it should, use SQL DELETE to empty it. Otherwise, instruct the user to use the RESUME(YES) option for that partition.

User response: Determine why the assumption of an empty partition was wrong. This may require consultation with the system programmer. Either remove the current partition's rows using an SQL DELETE, or change the LOAD option to RESUME YES. Resubmit the job.

Severity: 8 (error)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU329I   *csect-name* **- FIELD** *field-name* **IS NOT DEFAULTABLE**

Explanation: You specified a NULLIF or DEFAULTIF field specification option for:

- A column defined to be not nullable or not defaultable
- A field with no column counterpart.

System action: The utility abends to allow restart.

User response: Determine whether the field specification refers to the correct column. If it does, remove the NULLIF option from the field specification, and restart the job. Alternatively, TERM the utility.

Severity: 8 (error)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

➡ -TERM UTILITY (DB2) (DB2 Commands)

DSNU330I   *csect-name* **TABLESPACE IS NOT EMPTY**

Explanation: The RESUME keyword either specified or defaulted to a value of 'NO', which requires an empty table space. However, the LOAD command named a table space that was not empty.

System action: Further utility processing is terminated.

System programmer response: If the table space is used by more than one application, determine whether the table space should be empty. If it should be, use SQL DELETE to make it empty. Otherwise, instruct the user to use the RESUME(YES) option.

User response: Determine why the assumption of an empty table space was wrong. This may require consultation with the system programmer. Either remove the current table(s) rows using an SQL DELETE, or change the LOAD option to RESUME YES. Resubmit the job.

Severity: 8 (error)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

➡ DELETE (DB2 SQL)

DSNU331I   *csect-name* **FIELD LISTS ARE NOT ALLOWED WITH** *keyword* **KEYWORD**

Explanation: Field lists are not allowed with FORMAT UNLOAD or FORMAT SQL/DS keywords.

System action: Further utility processing is terminated.

User response: Determine whether the field specifications, the FORMAT, or INCURSOR option is correct. Eliminate whichever is incorrect and resubmit the job.

Severity: 8 (error)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU332I   *csect-name* **- INPUT VALUE X'**hex-data**' OUT OF RANGE FOR COLUMN '**column-name**'**

Explanation: The LOAD utility detected a numeric value outside of the range of permissible values for the column in the DB2 table.

csect-name
    The name of the control section that issued the message.

DSNU333I • DSNU334I

hex-data
> The input value that is outside of the range of the column.

column-name
> The name of the column.

A loss of significant digits results if the LOAD process continues. The input data type is SMALL INTEGER, INTEGER, DECIMAL, or FLOAT and is displayed in hexadecimal.

System action: Further utility processing is terminated unless DISCARD processing is active.

User response: Correct the input data and resubmit the job. Make sure that the 'field specification' on the LOAD statement is consistent with the table definition and the input data. This might mean changing any of the following items:

- The table definition
- The LOAD statement
- The input data.

Severity: 8 (error)

Related concepts:

▶ Restart of an online utility (DB2 Utilities)

▶ Numbers (DB2 SQL)

Related reference:

▶ LOAD (DB2 Utilities)

DSNU333I *csect-name* **- INPUT VALUE '***ebcdic-data***' OUT OF RANGE FOR COLUMN '***column-name***'**

Explanation: The LOAD utility detected a numeric value outside the range of permissible values for the column in the DB2 table. A loss of significant digits results if the LOAD process continues. The input data type is INTEGER EXTERNAL, DECIMAL EXTERNAL, or FLOAT EXTERNAL.

System action: Further utility processing is terminated.

User response: Correct the input data and resubmit the job. Make sure that the 'field specification' on the LOAD statement is consistent with the table definition and the input data. This might mean changing:

- The table definition
- The LOAD statement
- The input data.

Severity: 8 (error)

Related concepts:

▶ Restart of an online utility (DB2 Utilities)

Related reference:

▶ LOAD (DB2 Utilities)

DSNU334I *csect-name* **INPUT FIELD** *field-name* **INVALID FOR** *table-name*, **ERROR CODE** *error-code*

Explanation: The LOAD utility found incorrect input data.

csect-name
> The name of the control section that issued the message.

field-name
> The name of the field for which incorrect data was provided.

table-name
> The name of the table.

error-code
Description of the problem. Possible error codes are:

01	Input numeric value is out of the range for the table column.
02	Input numeric field contains invalid numeric data.
03	An error occurred when converting input numeric data to floating-point data.
04	Input packed decimal data is in error.
05	Input is too long for a date column.
06	Input is too short for a date column.
07	Input is too long for a time column.
08	Input is too short for a time column.
09	Input is too long for a timestamp column.
10	Input is too short for a timestamp column.
11	Input character field is too long for the table column.
12	Input variable character field is too long for the table column.
13	Input is out of range for the date/time column.
14	Input is invalid for the date/time column.
15	Input length is incorrect for numeric column.
16	Input contains a blank character.
17	Input is too long for a graphic column.
18	Input is invalid for a graphic column.
19	Input is too long for the table column after a required CCSID conversion.
20	Input contained an invalid code point for a required CCSID conversion.
21	Input does not conform to the rules for valid data for a required CCSID conversion.
22	Input causes an error during CCSID conversion. An example of this type of error would be an error in a user exit routine.
23	Input is invalid for a ROWID column.
24	Input contained a code point that when translated from the source CCSID to the target CCSID resulted in a substitution character being placed in the target string. This condition can occur when the NOSUBS keyword was specified and no discard data set was specified.
25	The range of values for an identity column is exhausted.
26	A null value has been supplied for a not null column.
27	Input is invalid for a security label column.
28	Input is a valid security label, but the SECLABEL and the user's SECLABEL are disjoint. The user does not have the authority to load this SECLABEL.
29	Decimal floating point conversion error.
31	XML is not well formed.

System action: Further utility processing is terminated.

User response: Correct the input data and resubmit the job. Make sure that the field specification on the LOAD statement is consistent with the table definition and the input data. Correcting this error might mean changing:

- The table definition
- The LOAD statement
- The input data

Severity: 8 (error)

Related reference:

⏏ LOAD (DB2 Utilities)

DSNU335I   *csect-name* **INPUT FIELD 'ffffffff' NOT ENTIRELY WITHIN INPUT RECORD**

Explanation: The LOAD utility found that the input field 'ffffffff' extends beyond the end of the input record.

If DISCARDS processing is active, the message severity is 4 (warning).

System action: For fixed length records, with no VARCHARs included, the length needed by the LOAD utility can be calculated before processing. In this case, when the LOAD utility determines that no records can be loaded because ALL of the INTO TABLE statements require input longer than the defined lengths, the LOAD job terminates on the first record without discarding any records. Further utility processing with mixed or non-fixed length records is terminated unless DISCARDS processing is active.

User response: Correct the input data and resubmit the job. Make sure that the 'field specification' on the LOAD statement is consistent with the table definition and the input data. This might mean changing:
- The table definition
- The LOAD statement
- The input data.

Severity: 4 (warning) or 8 (error)

Related concepts:

⏏ Restart of an online utility (DB2 Utilities)

Related reference:

⏏ LOAD (DB2 Utilities)

DSNU336I   *csect-name* **INVALID** *nullif* **SPECIFICATION FOR INPUT FIELD** *ffffffff*

Explanation: The LOAD utility found the NULLIF field specification to be incorrect.

System action: Further utility processing is terminated.

User response: Correct the NULLIF field specification and resubmit the job.

Severity: 8 (error)

Related concepts:

⏏ Restart of LOAD (DB2 Utilities)

Related reference:

⏏ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU337I   *csect-name* **RECORD(n), CONCATENATED CONTINUEIF RECORD IS TOO LARGE**

Explanation: The CONTINUEIF option was specified during LOAD to concatenate multiple physical records into one logical record for loading. The resultant concatenated record is too large to be processed. The maximum length allowed is 32767 bytes.

System action: When discard processing is specified, the input data set logical record is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set, and the severity level is 4. When discard process is not specified, the severity is 8 and the LOAD utility will terminate.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: When discard processing is specified, following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set. Correct the error in the discarded input data, and submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set. When discard processing is not specified: correct the error in the input data set and rerun the job.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 8

DSNU338I   *csect-name* **INVALID COLUMN SPECIFICATION FOR** *column name*

Explanation: The LOAD utility found the dynamically determined column location for an input field to be incorrect.

System action: The utility abends to allow restart.

User response: Correct the input field specification or the input data and restart the job. Alternatively, TERM the utility.

Severity: 8 (error)

DSNU339I   *csect-name* **INVALID CONTINUEIF SPECIFICATION**

Explanation: The CONTINUEIF clause of a LOAD utility extends beyond the end of the input record.

System action: The LOAD utility is terminated.

User response: Correct the CONTINUEIF clause. The column specification must be reduced and/or the value being compared must be shortened. The value being compared must be entirely within the input record.

Severity: 8 (error)

DSNU340I   *csect-name* **- ERROR LOADING INDEX, DUPLICATE KEY, INDEX=***index-name***, TABLE=***table-name***, RID OF INDEXED ROW=***indexed-row***, RID OF NON-INDEXED ROW=***nonindexed-row*

Explanation: A duplicate key was found during the BUILD phase of REORG INDEX or REBUILD INDEX, or a duplicate hash key was found during the BUILD phase of REORG INDEX. The duplicate key violates the uniqueness rule for the index or the hash key.

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index.

table-name
> The name of the table.

indexed-row
> The record ID (RID) of the indexed row.

nonindexed-row
> The record ID (RID) of the non-indexed row.

System action: All duplicate index entries of the indicated index are reported and the index is placed in RECOVERY PENDING status. Duplicate rows are present in the indexed table.

Utility processing continues with any other utility functions within the utility job step.

Operator response: Notify the system programmer.

System programmer response: Take steps to inhibit any further processing until the index is repaired. Applications that are dependent upon the uniqueness of the table rows should not be initiated until corrective action has been taken.

If the nonunique keys are correct, drop and re-create the index as nonunique.

If the records with duplicate keys should not be in the table, use the REPAIR utility with the DELETE option to remove the incorrect rows. Use the RID values provided in the message as input to the REPAIR utility.

After the duplicate data rows are removed or the unique index has been dropped, issue the DISPLAY DATABASE

command with the RESTRICT option against the target database. Use the REBUILD INDEX utility to recover any indexes that are in the REBUILD pending state. Optionally, use the CHECK INDEX utility to verify that the data and indexes are consistent.

User response: Notify the system programmer.

Problem determination: Use the REPAIR utility with LOCATE RID and DUMP statements to view the records with duplicate keys.

The inconsistent index can cause SQLSELECT to fail. Use CHECK INDEX to check the validity of the index.

Severity: 8 (error)

Related reference:

➦ -DISPLAY DATABASE (DB2) (DB2 Commands)

➦ CHECK INDEX (DB2 Utilities)

➦ REORG INDEX (DB2 Utilities)

➦ REPAIR (DB2 Utilities)

DSNU341I *csect-name* **- IMAGE COPY MUST BE TAKEN BEFORE REORG CAN BE RUN ON A DB2 CATALOG OR DIRECTORY TABLE SPACE:** *database-name.tablespace-name*

Explanation: The DB2 catalog or directory table space being reorganized does not have an image copy that the RECOVER utility can use to recover the table space to currency. Before running the REORG utility, you must run the COPY utility to create an image copy for the indicated table space. This action prevents having to recover the DB2 catalog or directory table space to a prior point in time if a problem occurs during REORG processing.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the directory or catalog database.

tablespace-name
> The name of the directory or catalog table space.

System action: Processing terminates.

User response: Run the COPY utility to create an image copy of the DB2 catalog or directory table space. If more than one catalog or directory table space is to be reorganized, consider using the QUIESCE and COPY utilities for the set of table spaces to be reorganized.

Use the REPORT utility to view the recovery information related to the table space. The output from the REPORT utility should indicate either the absence of an image copy, or the presence of a REORG LOG NO SYSCOPY record.

Severity: 8 (error)

Related reference:

➦ COPY (DB2 Utilities)

➦ QUIESCE (DB2 Utilities)

➦ REPORT (DB2 Utilities)

DSNU342I *csect-name* **ERROR LOADING INDEX, ERROR = X**reason-code, **INDEX =** *index-name*

Explanation: The LOAD, REORG or REBUILD INDEX utility encountered a processing error, which it could not correct, while extending the index identified by *index-name*.

System action: The building of index entries for the indicated index is not completed. An SVC dump of the database services address space is requested for diagnostic purposes. The extension of other indexes associated with the tables being loaded is attempted. Any subsequent attempt to use the index directly or indirectly will cause the invoker to abend.

Operator response: The table space in question might need to be stopped and started for utility-only processing to inhibit attempted application use of the defective index.

System programmer response: If the index is used by more than one application, applications that either insert rows into, or delete rows from, the indexed table or use the index to access rows must be inhibited from executing. One way to accomplish this is to place the table space in question under utility-only processing control by using the START DATABASE and STOP DATABASE commands until the REBUILD INDEX utility is executed.

Alternatively, the index can be dropped. Application execution can then proceed, and the index can be created again (unless the index was the partitioning index).

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: If the indexed table is used by more than one application, the system programmer should be informed so that applications that insert rows into, or delete rows from, the table or use the index directly are inhibited until corrective action is taken.

Normal corrective action would be to use the REBUILD INDEX utility option to reconstruct the index after the processing problem identified by the reason code has been resolved. Alternatively, the index can be dropped and then created again, unless the index in question is the partitioning index.

Severity: 8 (error)

Related concepts:

DB2 reason codes (DB2 Codes)

Related reference:

REBUILD INDEX (DB2 Utilities)

LOAD (DB2 Utilities)

REORG INDEX (DB2 Utilities)

REORG TABLESPACE (DB2 Utilities)

DSNU343I *csect-name* **- BUILD PHASE STATISTICS.** *nnn* **DUPLICATE KEY ERRORS ENCOUNTERED**

Explanation: The count of duplicate key errors encountered during processing is 'nnn'.

System action: Processing continues. For the LOAD utility, the records causing the duplicate key errors will be deleted in a subsequent phase.

Severity: 4 (warning) or 8 (error)

DSNU344I *csect-name* **- UNIQUE INDEX KEY DUPLICATES KEY OF INDEXED ROW AT RID** *record-id* **INDEX = *index-name* TABLE = *table-name* RECNO = *record-number* RID = *duplicate-id***

Explanation: The indicated input data set record was loaded into a table, but it duplicates the key value of an existing record or a hash key, which causes a unique index violation

csect-name
> The name of the control section that issued the message.

record-id
> The record ID where the input data set was loaded.

index-name
> The name of the index.

table-name
> The name of the table.

record-number
> The number for the record in the input data set to load.

duplicate-id
> The record ID of the duplicate key value.

System action: Processing continues.

No index entry is created for the record at RID *duplicate-id*. The record will be deleted in a subsequent phase of processing.

Severity: 4 (warning)

DSNU345I   *csect-name* **UNIQUE INDEX KEY DUPLICATES KEY FROM INPUT DATASET RECORD** *iiii* **LOADED AT RID** *rrrr* **INDEX** = *index-name* **TABLE** = *table-name* **RECNO** = *recno* **RID** = *duprid*

Explanation: Input data set record *recno* was loaded into table *table-name* at RID *duprid*, but it causes a unique index violation for index *index-name*. It duplicates the key value of another record in the load job: the record loaded at RID *rrrr* from input data set record *iiii*.

System action: Processing continues. Index entries are not created for either record. Both records will be deleted in a subsequent phase of processing.

Severity: 4 (warning)

DSNU346I   *csect-name* **RECORD(n), CONCATENATED SQL/DS RECORD IS TOO LARGE**

Explanation: A LOAD utility was executing with FORMAT SQL/DS input data. Multiple spanned segments of the input data were concatenated into one logical record for loading. The resultant concatenated record is too large to be processed. The maximum length allowed is 32764 bytes.

System action: When discard processing is specified, the input data set logical record is noted and is discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set and the severity is 4. When discard processing is not specified, the severity is 8 and the LOAD utility terminates.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: When discard processing is specified, examine the records in the DISCARDS data set after the LOAD utility completes. Correct the error in the discarded input data. Then run the LOAD utility with the RESUME YES option using the DISCARDS data set as input. When discard processing is not specified, correct the error in the input data set and rerun the job.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 4 or 8

DSNU347I   *csect-name* **DEADLINE** = *yyyy-mm-dd-hh.mm.ss.mmmmmmm*

Explanation: This message is displayed in one of the following circumstances:

- As part of the output from the DISPLAY UTILITY command
- When the REORG INDEX or REORG TABLESPACE utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option

In either case, this message is described as part of the DISPLAY UTILITY output in "DSNU100I" on page 620.

DSNU348I   *csect-name* **BUILD PHASE STATISTICS. NUMBER OF KEYS** = *nnnnnnnn* **FOR INDEX** *index-name* **PART** *part-num*

Explanation: This message is issued by the BUILD phase of the REBUILD INDEX utility to report the total number of index key entries used to build the indicated index partition.

System action: This message indicates the number of index key entries submitted to the DB2 data manager to build the indicated index partition. Both unique and nonunique key entries are counted. Rejected key entries are not counted.

User response: The user can verify that the number reported is consistent with expectations.

Severity: 0 (informational)

DSNU349I   *csect-name* **BUILD PHASE STATISTICS. NUMBER OF KEYS =** *nnnnnnnn* **FOR** *index-name*

Explanation: This message is issued by the BUILD phase of the LOAD, REORG or REBUILD INDEX utility. The message is repeated for each index built by the LOAD, REORG or REBUILD INDEX utility. It indicates the total number of index key entries used to build the indicated index.

System action: This message indicates the number of index key entries that were submitted to the DB2 data manager to build the indicated index. Both unique and nonunique key entries are counted. Rejected key entries are not counted.

User response: The user can verify that the number reported is consistent with expectations. For example, the user can determine if it matches the number of records loaded for the table indexed by the indicated index.

Severity: 0 (informational)

DSNU350I   *csect-name* **- EXISTING RECORDS DELETED FROM TABLESPACE**

Explanation: A LOAD utility execution, in which REPLACE was specified, has deleted all records from the specified table space.

System action: Processing continues.

Severity: 0 (informational)

DSNU351I   *csect-name* **- '***reorg-option***' OPTION IS NOT APPLICABLE FOR A DB2 CATALOG OR DIRECTORY OBJECT. THE OPTION IS IGNORED.**

Explanation: The identified option does not apply for this DB2 catalog or directory object. Table spaces and indexes within the DB2 catalog and directory may be processed differently than other table spaces and indexes. Consequently, these table spaces and indexes may not make use of reorg option *reorg-option*.

System action: The option is ignored and processing continues.

User response: None is required. However, you can remove the option from the utility input statement.

Severity: 4 (warning)

DSNU352I   **EXISTING RECORDS DELETED FROM TABLESPACE PARTITION x**

Explanation: A LOAD utility execution, in which PART 'x' REPLACE was specified, has deleted all data records from partition 'x'. Index records over partition 'x' will be deleted in a subsequent phase of processing.

System action: Processing continues.

Severity: 0 (informational)

DSNU353I   *csect-name* **RECORD '***n***' WILL BE DISCARDED DUE TO CHECK CONSTRAINT** *constraint-name* **VIOLATION ON TABLE** *table-name*

Explanation: The discarded record violates the check constraint named *constraint-name* defined on table *table-name*. The record is not loaded and is discarded during the DISCARD phase.

System action: The LOAD utility continues to process additional records in the input data set.

User response: After the LOAD utility completes, examine the record in the DISCARDS data set. Correct the error in the discarded input data. Submit a RESUME(YES) LOAD job with the DISCARDS data set as the input data set.

Severity: 4 (warning)

DSNU354I • DSNU355I

DSNU354I *csect-name* **RECORD '*n*' VIOLATES CHECK CONSTRAINT** *constraint-name* **ON TABLE** *table-name*

Explanation: The LOAD utility cannot load record *n* into table *table-name* because the check constraint *constraint-name* would be violated. The record is not loaded.

System action: Processing terminates. An abend is requested so that the LOAD can be restarted after the error condition is corrected.

User response: Find the failing record in the input data set, correct the condition that caused the check constraint to be violated, and restart the LOAD utility. You can optionally include a SYSDISC DD statement in the JCL input to hold copies of records that are not loaded.

Severity: 8 (error)

DSNU355I *csect-name* **INDEXVAL PHASE STATISTICS -** *error-count* **DUPLICATE KEY ERRORS, KEY EVALUATION ERRORS, XML ERRORS OR LOB ERRORS WERE CORRECTED BY DELETING** *deleted-count* **DATA ROWS**

Explanation: This message reports the results of the INDEXVAL phase of the LOAD utility. Data rows that were added during the LOAD or RELOAD phase of this LOAD utility invocation were deleted because the data rows caused one of the following conditions:

- Unique index violations in the BUILD phase
- Key evaluation errors during the build of an index on an expression-based index
- XML document errors or LOB errors in the BUILD phase

csect-name
> The name of the control section that issued the message.

error-count
> The number of unique index violations or key evaluation errors.

deleted-count
> The number of data rows that were deleted to correct the errors.

Unique index violations are counted in one of two ways. An input data set record which has a value for a uniquely indexed column that duplicates a value that existed in the index prior to the LOAD job is counted as one duplicate key error. Two input data set records which have like values for a uniquely indexed column are counted as two unique index errors (that is, each of the two input data set records is in error).

System action: Processing continues.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Examine the error summary from the LOAD job. Determine which of the discarded records need to be corrected and loaded into the table space.

Compare the number of duplicate key errors or key evaluation errors that were corrected in the INDEXVAL phase to the number of duplicate key errors that were detected in the BUILD phase (reported by message DSNU343I).

If the counts are unequal, notify the system programmer.

If the counts are equal, following the LOAD utility execution, examine the discards file or the LOAD error summary to determine which input data set records caused the unique index violations or key evaluation errors. Revise the input records so they will not introduce duplicate values into unique indexes and reapply the input records to the table space. Alternatively, if appropriate, make the index non-unique and reapply the rejected records that caused the duplicate key errors to the table space.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 4 (warning)

Related information:
"DSNU343I" on page 679

DSNU356I   *csect-name* **INDEXVAL PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The INDEXVAL phase of the LOAD utility has completed normally. The elapsed time for the execution was 'hh' hours, 'mm' minutes, and 'ss' seconds.

Severity: 0 (informational)

DSNU357I   *csect-name* **- 'LOG NO' OPTION REQUIRED FOR DB2 CATALOG AND DIRECTORY TABLE SPACES**

Explanation: The LOG NO option must be specified when reorganizing a DB2 catalog or directory table space.

System action: Processing terminates.

User response: Add the LOG NO option to the utility input statement.

Severity: 8 (error)

DSNU358I   *csect-name* **- 'UNLOAD ONLY' OPTION NOT VALID FOR DB2 CATALOG OR DIRECTORY TABLE SPACE**

Explanation: The UNLOAD ONLY option is not permitted for DB2 catalog or directory table spaces.

System action: Processing terminates.

User response: Determine the purpose for using the UNLOAD ONLY option. There might be alternative ways of accomplishing the required function (for example, the UNLOAD PAUSE option).

Severity: 8

DSNU359I   **KEYWORD 'SORTKEYS' SPECIFIED BUT NO INDEX OR FOREIGN KEYS EXIST, KEYWORD IS IGNORED**

Explanation: The 'SORTKEYS' keyword was specified for a LOAD or REORG utility job; however, it was found that no index and no foreign keys were defined for any of the tables being loaded or reorganized. The keyword is ignored and processing continues normally.

System action: Processing continues normally.

User response: If an index or foreign keys were expected to be found for the table or tables being loaded or reorganized, determine the reason the index or foreign keys could not be found.

Problem determination: Probably a user error. Determine if an index should be defined on any of the tables in the table space, or if foreign keys should be defined. If so, define the index or foreign keys; otherwise, remove the 'SORTKEYS' keyword from the utility statement if necessary.

Severity: 0 (informational)

DSNU360I   *csect-name* **- IN REORG WITH SHRLEVEL REFERENCE OR CHANGE, DATA SET** *name* **EXISTS, CONTEXT=***c*

Explanation: For the area (table space or partition) being reorganized, REORG with SHRLEVEL REFERENCE or CHANGE unloads data from the original data sets and reloads data into shadow data sets. REORG then exchanges the names of the original and shadow data sets, using temporary names during the renaming. The original name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.I0001.A*nnn*". The shadow name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.J0001.A*nnn*". The temporary name of a data set is in the form "*catname*.DSNDB*x*.*dbname*.*psname*.T0001.A*nnn*". REORG uses such data sets for indexes as well as data.

During its initialization, REORG cannot execute if:
- Data sets with the temporary names already exist.
- DB2-managed data sets with the shadow names already exist.
- User-managed data sets with the shadow names do **not** already exist.

DSNU360I

To exchange the names of the original and shadow data sets, REORG performs the following renaming:

1   In the switch phase, it renames the original data set from the original name to the temporary name

2   In the switch phase, it renames the shadow data set from the shadow name to the original name

3   In the utilterm phase, it renames the original data set from the temporary name to the shadow name (for user-managed data sets). It deletes the temporary for DB2-managed data sets.

If a -TERM UTILITY command is issued after any renaming, or a restart of the REORG switch phase occurs after any renaming, the data set names revert back to their original form prior to the renaming:

4   The original data set from the shadow name to the temporary name

5   The shadow data set from the original name to the shadow name

6   The original data set from the temporary name to the original name

The context code indicates the specific type of error:

1   During the switch phase of REORG, on rename 1, REORG detected that a data set with a temporary name already exists. This error condition can arise for a DB2-managed or user-managed data set.

2   During the switch phase of REORG, on rename 2, REORG detected that a data set with a original name already exists. This error condition can arise for a DB2-managed or user-managed data set.

3   During the utilterm phase of REORG, on rename 3, REORG detected that a data set with a shadow name already exists. This error condition can arise for a user-managed data set.

4   During switch phase restart or a -TERM UTILITY during the switch phase, on rename 4 (while reverting back to the original name), -TERM UTILITY detected that a data set with a temporary name already exists. This error condition can arise for a DB2-managed or user-managed data set.

5   During switch phase restart or a -TERM UTILITY during the switch phase, on rename 5 (while reverting back to the original name), -TERM UTILITY detected that a data set with a shadow name already exists. This error condition can arise for a DB2-managed or user-managed data set.

6   During switch phase restart or a -TERM UTILITY during the switch phase, on rename 6 (while reverting back to the original name), -TERM UTILITY detected that a data set with a original name already exists. This error condition can arise for a DB2-managed or user-managed data set.

8   During initialization of REORG, REORG detected that a data set with a temporary name already exists. This error condition can arise for a DB2-managed or user-managed data set.

System action: For context code 1, or 2, the utility job step that caused the function to be requested abends with code 00E4031B. The utility job is placed in the stopped state.

For context code 3, the utility job step that caused the function to be requested continues without performing this renaming.

For context code 4, 5, or 6, the object is left in a recovery pending state, -TERM UTILITY continues to undo renaming, and a message is issued. The objects and messages are as follows:

- If the data set is part of a nonpartitioned table space, the object is the table space, and the message is DSNU562I.
- If the data set is part of a partition of a table space, the object is the partition, and the message is DSNU560I.
- If the data set is part of a nonpartitioned index, the object is the index, and the message is DSNU557I.
- If the data set is part of a partition of a partitioned index, the object is the partition of the index, and the message is DSNU559I.

For context code 8, REORG does not execute.

User response: For context code 1 or 2, you can choose between these actions:

- Delete the data set; then RESTART the utility.
- TERM the utility. Delete the data set before trying to execute REORG again with SHRLEVEL REFERENCE or CHANGE.

For context code 3, if you want to perform the renaming, delete the data set with the shadow name and then perform the renaming.

For context code 4, 5, or 6, execute the RECOVER or REBUILD INDEX utility; then delete the shadow and temporary data sets (if they still exist).

For context code 8, delete the data set and then execute REORG.

Severity: 8 (error)

DSNU361I   *csect-name* **FOREIGN KEY HAS NO PRIMARY KEY FOR RELATIONSHIP=***r1* **INPUT RECORD NUMBER=***nnnn***, RID=X***rrrr*

Explanation: Input record *nnnn* caused a row to be loaded into table *table* at RID *rrrr* with no primary key for the relationship with the name *r1*.

System action: Processing continues.

User response: None required. The constraint violation described will be rectified later in processing.

Severity: 4 (warning)

DSNU362I   *csect-name* **- IN** *utility* **WITH SHRLEVEL CHANGE, LOG HAS BECOME SHORT, BUT MAXRO IS DEFER, MEMBER=***member-name***, UTILID=***utilid*

Explanation: When you execute *utility* with SHRLEVEL CHANGE, DB2 will add a five second pause to the next iteration, if the value of MAXRO is DEFER, and the actual time for an iteration and the estimated time for the next iteration are both less than five seconds. This pause limits the utility's consumption of processor time.

Within an execution of the utility, the first time that this situation occurs, DB2 produces this message, which announces the pause. On all later occurrences of this situation, DB2 still adds the pause, but DB2 sends the message only if DB2 has *not* sent the message (for this execution of the utility) within the past 30 minutes.

WTO   This message is sent to the console, but it is not an action message. It is not sent to SYSPRINT.

member-name
        In a data sharing environment, *member-name* identifies the DB2 member. In a non-sharing environment, *member-name* is blank. The variable of *utilid* identifies the utility job step.

System action: The utility continues execution, with a five second pause. If the conditions in the explanation above continue to be true, DB2 continues to add five second pauses.

Operator response: You are not required to take any action at this time, but if you were planning to wait for an appropriate time to execute the -ALTER UTILITY command to increase the value of MAXRO and thus cause the utility to finish, now might be an appropriate time to execute.

User response: You are not required to take any action at this time, but if you were planning to wait for an appropriate time to execute the -ALTER UTILITY command to choose the value of MAXRO, and thus cause REORG to finish, now might be an appropriate time to execute.

Severity: 0 (informational)

DSNU363I   *csect-name* **- ENFORCE PHASE STATISTICS - RELATIONSHIP** *r1* **FOREIGN KEYS CHECKED=***n* **CONSTRAINT VIOLATIONS ENCOUNTERED=***m*

Explanation: This message provides statistics for the enforce phase of the load utility. For relationship 'r1', the number of foreign keys checked is 'n'. The number of constraint violations found is 'm'.

System action: Processing continues.

Severity: 0 (informational)

DSNU364I   **csect-name PARTITIONS WILL BE LOADED IN PARALLEL, NUMBER OF TASKS = nnnn**

Explanation: This message is issued by the RELOAD phase of the LOAD utility. The message indicates that the LOAD utility is able to load partitions in parallel. The number of utility tasks used to load partitions is indicated by *nnnn*.

System action: Processing of the RELOAD phase continues.

DSNU365I • DSNU368I

User response: You can verify that the number of tasks reported is consistent with your expectations. For example, if 4 partitions are to be loaded, you might expect to see the number of tasks = 4.

Severity: 0

DSNU365I   *csect-name* **- USER** *auth-id* **DOES NOT HAVE AUTHORITY ON REORG MAPPING TABLE**

Explanation: The REORG utility was run with SHRLEVEL CHANGE and a mapping table, but user *auth-id* does not have authority to delete, insert, select, and update in the mapping table specified in the utility statement.

System action: Further utility processing is terminated.

User response: Obtain authorization to delete, insert, select, and update in the mapping table specified in the utility statement.

Severity: 8 (error)

DSNU366I   *csect-name* **- REORG MAPPING TABLE HAS WRONG COLUMNS**

Explanation: A mapping table was specified for a utility job executing REORG with SHRLEVEL CHANGE. However, the table has the wrong set of columns. This message results from any CREATE table parameters that are missing from, added to or different than the CREATE TABLE statement for the mapping table.

System action: REORG does not execute.

User response: Correct the table columns in the mapping table, and run the REORG utility again.

Severity: 8 (error)

Related concepts:

Before running REORG TABLESPACE (DB2 Utilities)

Access with REORG TABLESPACE SHRLEVEL (DB2 Utilities)

Related reference:

Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU367I   *csect-name* **- ENFORCE PHASE STATISTICS -** *n* **ROWS DELETED FROM TABLE** *t*

Explanation: This message reports the results of the ENFORCE phase of the LOAD utility. Data rows added during the RELOAD phase of the LOAD utility have been deleted because they contained foreign keys which did not have a corresponding primary key. A total of *n* data rows were deleted from table *t*.

Primary errors may cascade and remove additional records. The count is the sum of both primary and secondary constraint violations.

System action: Processing continues.

User response: Determine why no primary key exists for the deleted record.

Problem determination: All primary errors are accompanied by a DSNU361I message which identifies the relationship and input record which does not have a primary key value. The report phase will report all secondary errors and indicate the primary error record which caused this record to be deleted.

The relevant parent table can be found by querying the SYSIBM.SYSRELS catalog table. The relevant columns can be found by querying the SYSIBM.SYSFOREIGNKEYS catalog table.

Severity: 4 (warning)

DSNU368I   *csect-name* **- REORG MAPPING TABLE HAS WRONG INDEX**

Explanation: A mapping table was specified for a REORG utility job that specifies SHRLEVEL CHANGE. However, the table contains the wrong index. This message might result from any CREATE INDEX parameters that are missing from, added to, or different from the CREATE INDEX statement in the mapping table. It might also result from the existence of any other indexes on the mapping table.

System action: The REORG utility does not execute.

User response: Create a mapping table with the correct index and no other indexes, and run the REORG utility.

Severity: 8 (error)

Related concepts:

➡ Before running REORG TABLESPACE (DB2 Utilities)

➡ Access with REORG TABLESPACE SHRLEVEL (DB2 Utilities)

Related reference:

➡ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU369I   *csect-name* **- ENFORCE PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The ENFORCE phase of the LOAD utility has completed normally. The elapsed time for the execution was 'hh' hours, 'mm' minutes, and 'ss' seconds.

Severity: 0 (informational)

DSNU370I   *csect-name* **- THE LOCAL FORMAT OPTION HAS BEEN USED WITH A DATE OR TIME AND NO LOCAL EXIT HAS BEEN INSTALLED.**

Explanation: The Installation Application Programming Defaults Panel indicated that an exit for date or time was supplied, but the exit supplied by DB2 was not replaced.

System action: The statement cannot be executed.

System programmer response: To replace the DB2-supplied exit routine, an SMP USERMOD must be applied. If an SMP USERMOD is undesirable, the replacement module may be linked into a user library either included in the MVS LINKLIST or concatenated to the STEPLIB in the started task JCL for *ssnm*DBM1 (*ssnm* is the subsystem name).

User response: Contact the system programmer about installation of the date or time exit.

Severity: 8 (error)

DSNU371I   *csect-name* **- REORG MAPPING TABLE'S TABLE SPACE MUST BE SEGMENTED**

Explanation: A REORG TABLESPACE utility statement with the SHRLEVEL CHANGE option specified an invalid mapping table. The mapping table must be defined in a segmented table space or a partition-by-growth table space.

csect-name
        The name of the control section that issued the message.

System action: The REORG TABLESPACE utility does not execute.

User response: Specify a mapping table that is defined in a segmented table space or a partition-by-growth table space.

Severity: 8 (error)

Related concepts:

➡ Before running REORG TABLESPACE (DB2 Utilities)

➡ Partition-by-growth table spaces (Introduction to DB2 for z/OS)

➡ Segmented (non-universal) table spaces (Introduction to DB2 for z/OS)

Related reference:

➡ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

➡ CREATE TABLESPACE (DB2 SQL)

DSNU372I   *csect-name* **- CANNOT REORGANIZE REORG MAPPING TABLE'S TABLE SPACE**

Explanation: REORG with SHRLEVEL CHANGE specified a mapping table that resides in the table space that is being reorganized. The table space that contains the mapping table must not be the table space that is being reorganized.

System action: REORG does not execute.

User response: Specify a mapping table that is not in the table space that is being reorganized.

Severity: 8 (error)

DSNU373I   *csect-name* **- IN -ALTER UTILITY OR REORG WITH SHRLEVEL REFERENCE OR CHANGE, DEADLINE HAS ALREADY OCCURRED, CONTEXT=***c*

Explanation: If a deadline is specified in the -ALTER UTILITY command or the REORG utility with SHRLEVEL REFERENCE or CHANGE, the deadline cannot occur before the time of execution of the -ALTER UTILITY or REORG that specified the deadline. Context codes are:

1   The error occurred on -ALTER UTILITY.

2   The error occurred on REORG.

System action: -ALTER UTILITY or REORG does not execute.

User response: Execute -ALTER UTILITY or REORG with no deadline or with a deadline that has not already occurred. If you want to terminate an execution of REORG, use the -TERM UTILITY command.

Severity: 8 (error)

DSNU374I   *csect-name* **-** *utility-name* **WITH SHRLEVEL REFERENCE OR CHANGE HAS BEEN TERMINATED, REASON=***reason-code*

Explanation: The utility has terminated.

csect-name
          The name of the control section that issued the message.

utility-name
          The name of the utility.

reason-code
          A numeric value that indicates the reason for the failure.

          1   The utility with SHRLEVEL CHANGE specified LONGLOG TERM, and at the end of an iteration of log processing, the number of log records that the next iteration would process is not sufficiently lower than the number that the previous iterations processed.

          2   DB2 estimated that the switch phase of REORG with SHRLEVEL REFERENCE or CHANGE would not start by the deadline specified in the DEADLINE phrase.

          3   REORG terminated because the FlashCopy® failed and no inline copy was made.

System action: The utility terminates.

User response: Take the action that corresponds to the *reason-code* value:

1   Attempt a new invocation of the utility later, perhaps when you expect that the application workload is lower or the maximum needed amount of time for applications to have read-only access is longer.

2

          • On SHRLEVEL CHANGE, attempt a new invocation of reorganization later, perhaps when you expect to have a longer time available, you expect that the application workload is lower, or you expect that the maximum needed amount of time for applications to have read-only access is longer.

          • On SHRLEVEL REFERENCE, attempt a new invocation of reorganization later, perhaps when you expect to have more time available.

3   Correct the problem with FlashCopy and run the utility again.

Severity: 8 (error)

DSNU375I   *csect-name* **DISCARD PHASE STATISTICS -** *nnn* **INPUT DATA SET RECORDS DISCARDED**

Explanation: Input data set records were rejected by the LOAD utility for one or several reasons. The rejected records have been placed in a discards data set. The number of records rejected is 'nnn'. Message DSNU399I summarizes the errors causing the records to be rejected.

System action: Processing continues.

User response: Examine the LOAD error summary and decide which of the discarded records, if any, should be corrected and reapplied to the table space.

Severity: 4 (warning)

DSNU376I   *csect-name* **DISCARD PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The DISCARD phase of the LOAD utility has completed normally. The elapsed time for the execution was 'hh' hours, 'mm' minutes, and 'ss' seconds.

Severity: 0 (informational)

DSNU377I   *csect-name* **- IN** *utility* **WITH SHRLEVEL CHANGE, THE LOG IS BECOMING LONG,**
            MEMBER=*member-name***, UTILID=***utilid*

Explanation: During log processing for REORG with SHRLEVEL CHANGE, the number of log records that the next iteration will process is not sufficiently lower than the number that the previous iterations processed. This situation means that during reorganization, the log is not being read quickly enough to catch up to the application writing the log. This is a WTO message; it is sent to the console, but it is not an action message. It is also sent to SYSPRINT.

During log processing for *utility* with SHRLEVEL CHANGE, the number of log records that the next iteration is to process is not sufficiently lower than the number that the previous iterations processed. When you executed the utility, the log was not read quickly enough to catch up to the application that is writing the log. This is a WTO message; it is sent to the console, but it is does not require any user action. The message is also sent to SYSPRINT.

DB2 refrains from taking the specified action in the LONGLOG phrase if any one of the following events occur for this execution of the utility before the delay expires:

- You issue an -ALTER UTILITY command.
- You issue a -TERM UTILITY command.
- DB2 estimates that the time to perform the next iteration is less than or equal to the time specified in the MAXRO phrase.
- The utility terminates for any reason (including the deadline).

member-name
        Identifies the DB2 member in a data sharing environment. It is blank in a non-data-sharing environment.

utilid   Identifies the utility job step.

System action: After a delay (specified by the DELAY phrase of the *utility* or the -ALTER UTILITY command), the REORG utility performs the action specified by the LONGLOG phrase of the REORG utility or the -ALTER UTILITY command.

Operator response: No action is required. However, if the action that is specified by the LONGLOG phrase is not what you wanted, the operator (or a user-written program that is triggered by the message) can take an action.

Severity: 4 (warning)

Related reference:

↪ -TERM UTILITY (DB2) (DB2 Commands)

↪ -ALTER UTILITY (DB2) (DB2 Commands)

DSNU378I   *csect-name* **- REORG WITH SHRLEVEL NONE HAS INAPPROPRIATE NOSYSREC KEYWORD, REASON=***c*

Explanation: REORG with SHRLEVEL NONE allows the following combinations of the SORTDATA and NOSYSREC keywords:

- Neither keyword
- SORTDATA alone (without NOSYSREC)
- SORTDATA NOSYSREC

These combinations do not trigger this message. However, NOSYSREC alone (without SORTDATA) is inappropriate and triggers this message with a reason code of 1.

Also, for REORG with SHRLEVEL NONE, if you specify UNLOAD PAUSE, UNLOAD ONLY, or UNLOAD EXTERNAL, NOSYSREC is inappropriate and triggers this message with a reason code of 2. If the two conditions (omission of SORTDATA and inclusion of UNLOAD PAUSE, ONLY, or EXTERNAL) are both true, NOSYSREC triggers this message with a reason code of 2.

System action: REORG does not execute.

User response: Specify an invocation of REORG with appropriate parameters. Specifically, for reason code 1, if you want to execute REORG with SHRLEVEL NONE, add SORTDATA and/or remove NOSYSREC. For reason code 2, if you want to execute REORG with SHRLEVEL NONE, remove NOSYSREC, UNLOAD PAUSE, UNLOAD ONLY, or UNLOAD EXTERNAL .

Severity: 8 (error)

DSNU379I   *csect-name* **- ALTER UTILITY, UTILID = ***utilid***: SWITCH PHASE IN REORG HAS ALREADY STARTED, NO ALTER IS PERFORMED**

Explanation: The -ALTER UTILITY command was applied to a job that is executing REORG with SHRLEVEL REFERENCE or CHANGE. However, nothing was altered, because REORG had already started its switch phase.

System action: ALTER UTILITY has no effect.

Severity: 4 (warning)

DSNU380I   *csect-name* **TABLESPACE** *database.tablespace-name* **PARTITION** *part-num* **IS IN COPY PENDING**

Explanation: An image copy is required to establish a new base from which the table space partition can be recovered. Otherwise, the results of running the RECOVER utility on that table space partition are unpredictable.

If the RECOVER TABLESPACE utility with the keyword TOCOPY is being used, the START_RBA or START_LRSN value of the associated full image copy might be outside the available recovery information. To verify this, look at the REPORT utility output. The RECOVER TABLESPACE utility will process the TOCOPY keyword and put the recovering object in copy pending status.

User response: Take an image copy of the associated table space partition. If the message was issued after the COPY utility failed, correct all specified errors for the associated table space before attempting another image copy of that table space.

Severity: 4 (warning)

DSNU381I   *csect-name* **TABLESPACE** *database.tablespace-name* **IS IN COPY PENDING**

Explanation: This message is issued after:

- A LOAD or REORG utility with the LOG(NO) option left the associated table space in an unrecoverable state.
- An image copy failed because broken pages were detected or an image copy was terminated.
- A partial recovery finished successfully.
- A MODIFY RECOVERY left the associated table space in an unrecoverable state.

  If the RECOVER TABLESPACE utility with keyword TOCOPY is being used, the START_RBA or START_LRSN value of the associated full image copy might be outside the available recovery information. To verify this, look at the REPORT utility output. The RECOVER TABLESPACE utility will process the TOCOPY keyword and put the recovering object in copy pending status.

An image copy is required to establish a new base from which the table space can be recovered. Otherwise, the results of running the RECOVER utility on that table space are unpredictable.

User response: Take an image copy of the associated table space. If the message was issued after the COPY utility failed, correct all specified errors for the associated table space before attempting another image copy on that table space.

Severity: 4 (warning)

DSNU382I   *csect-name* **-** *utility* **UTILITY, UTILID =** *utilid* **REQUESTED TO ALTER, PHASE =** *utility-phase*

Explanation: When you issue a -ALTER UTILITY command, DB2 issues this message in response. It acknowledges that the request for alteration was received and verified. At the time the -ALTER UTILITY request was received, the *utility* was in the utility job step identified by UTILID *utilid*.

System action: DB2 has accepted the request to alter the indicated utility job step. The utility alters its parameter values.

User response: This message is an acknowledgement that the request has been accepted and is in progress.

Severity: 0 (informational)

DSNU383I   *csect-name* **CURRENT ITERATION NUMBER =** *nnnnnn* **WRITE ACCESS ALLOWED IN CURRENT ITERATION =** *www* **ITERATION BEFORE PREVIOUS ITERATION: ELAPSED TIME =** *ha:ma:sa* **NUMBER OF LOG RECORDS PROCESSED =** *nnnnna* **PREVIOUS ITERATION: ELAPSED TIME =** *hb:mb:sb* **NUMBER OF LOG RECORDS PROCESSED =** *nnnnnb* **CURRENT ITERATION: ESTIMATED ELAPSED TIME =** *hc:mc:sc* **ACTUAL ELAPSED TIME SO FAR =** *hd:md:sd* **ACTUAL NUMBER OF LOG RECORDS BEING PROCESSED =** *nnnnnd* **CURRENT ESTIMATE FOR NEXT ITERATION: ELAPSED TIME =** *he:me:se* **NUMBER OF LOG RECORDS TO BE PROCESSED =** *nnnnne*

Explanation: This message is displayed in one of the following circumstances:

- As part of the output from the DISPLAY UTILITY command
- When the REORG INDEX or REORG TABLESPACE utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option

In either case, this message is described as part of the DISPLAY UTILITY output in "DSNU100I" on page 620.

DSNU384I   *csect-name* **- MAXRO =** *maxro-value* **SECONDS LONGLOG =** *longlog-value* **DELAY =** *delay-value* **SECONDS**

Explanation: This message is displayed in one of the following circumstances:

- As part of the output from the DISPLAY UTILITY command
- When the REORG INDEX or REORG TABLESPACE utility estimates that the SWITCH phase will not start by the deadline that is specified in the DEADLINE option

In either case, this message is described as part of the DISPLAY UTILITY output in "DSNU100I" on page 620.

DSNU385I   *csect-name* **- LOG PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: This message is issued at the normal completion of the LOG phase of the utility. The time that has elapsed between the beginning and the completion of this phase is measured in units of:

hh   Hours

mm   Minutes

ss   Seconds

System action: Processing continues to the next phase of the utility.

Severity: 0 (informational)

DSNU386I • DSNU392I

DSNU386I   *csect-name* **- LOG PHASE STATISTICS. NUMBER OF ITERATIONS =** *iiiiiiii*, **NUMBER OF LOG RECORDS =** *rrrrrrrr*

Explanation: This message is issued at the normal completion of the LOG phase of the utility.

iiiiiiii   The number of iterations of log processing that were performed. This number includes one or more iterations during which applications have read/write access plus one final iteration during which applications have read-only access.

rrrrrrrr   The number of log records that were read.

Application of some of these log records might be unnecessary. Therefore, the LOG phase might apply only a subset of these log records to the shadow copy of the table space or partition.

System action: The LOG phase of the utility is completed.

Severity: 0 (informational)

DSNU387I   *csect-name* **- SWITCH PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: This message is issued at the normal completion of the switch phase of the REORG utility. The time that has elapsed between the beginning and the completion of this phase was:

hh   hours

mm   minutes

ss   seconds

System action: Processing continues to the next phase of the REORG utility.

Severity: 0 (informational)

DSNU390I   *csect-name* **INVALID CONVERSION FOR FIELD** *cccccccc*

Explanation: The input field type for column, *cccccccc*, is incompatible with its data type (as found in the catalog).

System action: Utility processing is terminated.

User response: Correct the error, and resubmit the job.

Severity: 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

➡ DB2 stand-alone utilities (DB2 Utilities)

DSNU391I   *csect-name* **- SORTBLD PHASE STATISTICS. NUMBER OF INDEXES =** *mmm*

Explanation: This message is issued at normal completion of the SORTBLD phase of the LOAD, REORG, or REBUILD INDEX utility. The number of indexes that were built is indicated as *mmm*.

System action: The SORTBLD phase of either the LOAD, REORG, or REBUILD INDEX utility is completed.

Severity: 0 (informational)

DSNU392I   *csect-name* **- SORTBLD PHASE COMPLETE, ELAPSED TIME =** *hh:mm:ss*

Explanation: This message is issued at the normal completion of the SORTBLD phase of the LOAD, REORG, and REBUILD INDEX utility. The time that has elapsed between the beginning of the phase and the completion of this phase was *hh* hours, *mm* minutes, and *ss* seconds.

System action: Processing continues to the next phase of the LOAD, REORG, or REBUILD INDEX utility. If this is the last phase, processing continues to the next utility statement in the utility job step, if any.

Severity: 0 (informational)

DSNU393I   *csect-name* **SORTBLD PHASE STATISTICS - NUMBER OF KEYS =** *nnnnnnnn* **FOR INDEX** *index-name*
         PART *part-num*

Explanation: This message is issued by the SORTBLD phase of the LOAD, REORG, or REBUILD INDEX utility to report the total number of index key entries used to build the indicated index partition.

System action: This message indicates the number of index key entries for the index partition being built. Both unique and nonunique key entries are counted. Rejected key entries are not counted.

User response: You can verify that the number reported is consistent with expectations.

Severity: 0 (informational)

DSNU394I   *csect-name* **SORTBLD PHASE STATISTICS. NUMBER OF KEYS =** *nnnnnnnn* **FOR INDEX** *index-name*

Explanation: This message is issued by the SORTBLD phase of the LOAD, REORG, or REBUILD INDEX utility. This message is repeated for each index that is built by the LOAD, REORG, or REBUILD INDEX utility.

csect-name
         The name of the control section that issued the message.

nnnnnnnn
         The total number of index key entries that are used to build the indicated index. Both unique and non-unique key entries are counted. Rejected key entries are not counted. If the index is a partitioning index, the total equals the sum of the number of index key entries that are reported in prior DSNU393I messages.

index-name
         The name of the index.

System action: Processing continues.

User response: You can verify that the number reported is consistent with expectations. For example, you can determine whether the number matches the number of records that were loaded into the table that is indexed by the indicated index. The number of loaded records is reported in message DSNU302I, DSNU303I, or DSNU304I. The number of mapping index keys that are processed does not need to match the total number of data records that are reloaded.

Severity: 0

Related information:

"DSNU393I"

"DSNU302I" on page 665

"DSNU303I" on page 665

"DSNU304I" on page 666

DSNU395I   *csect-name* **INDEXES WILL BE BUILT IN PARALLEL, NUMBER OF TASKS =** *nnnn*

Explanation: This message is issued by the RELOAD phase of the LOAD or REORG or UNLOAD phase of the CHECK INDEX or REBUILD INDEX utility. The message indicates that the CHECK INDEX, LOAD, REORG, or REBUILD INDEX utility is able to build indexes in parallel. The number of utility tasks used to build the indexes is indicated by *nnnn*.

System action: Processing of the RELOAD phase continues.

User response: You can verify that the number of tasks reported is consistent with expectations. For example, if 3 indexes are to be built, the user might expect to see number of tasks = 6.

Severity: 0 (informational)

DSNU396I   *csect-name* **REPORT PHASE COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: The REPORT phase of the LOAD utility has completed normally. The elapsed time for the execution was 'hh' hours, 'mm' minutes, and 'ss' seconds.

Severity: 0 (informational)

DSNU397I

DSNU397I   *csect-name* **NUMBER OF TASKS CONSTRAINED BY** *resource-type* **TO** *number-of-tasks*

Explanation: This message indicates that while a utility is able to start subtasks to perform operations in parallel, the number of tasks that it will use has been constrained by a lack of some resource.

csect-name
>   The name of the control section that issued the message.

resource-type
>   The constraining resource, which can have the following values and meanings:

>   **VIRTUAL STORAGE**
>>   The utility could start additional tasks, but there is insufficient virtual storage in the address space in which the utility job is running to make it effective to do so.

>   **VIRTUAL STORAGE BELOW**
>>   The utility could start additional tasks, but there is insufficient virtual storage below the line in the address space in which the utility job is running to make it effective to do so.

>   **DD STATEMENTS**
>>   The utility could start additional tasks, but the user has specified DD statements for the data sets to be used by SORT for sort work data sets or for sort message data sets, and fewer sort work data set groups or sort message data sets have been specified than the maximum number of tasks the utility could otherwise start.

>   **CPUS**   The utility could start additional tasks, but it would not be efficient to do so because it would overtax the available CPUs.

>   **SWA=BELOW**
>>   The utility could start additional tasks, but there is insufficient storage below the line in the address space in which the job is running.

>   **PARALLEL N**
>>   The utility could start additional tasks, but the PARALLEL keyword setting for the utility limits the number of tasks that will be started.

>   **ZPARM PARAMDEG_UTIL**
>>   The utility could start additional tasks, but the setting of the PARAMDEG_UTIL subsystem parameter limits the number of tasks that will be started.

number-of-tasks
>   The maximum number of tasks that can be started after applying the constraint for the resource that was identified.

>   This value might be different than what you specified, if the constraint is less than the minimum number of tasks that are needed for the utility to complete.

System action: Processing of the phase continues.

User response: No action is required. However, if you want to increase the number of tasks used by the utility in order to reduce the elapsed time of the utility job, the possible actions depend on the value of *resource-type* in the message:

VIRTUAL STORAGE
>   Increase the amount of virtual storage available to the utility job by specifying a larger region size in the job JCL, altering the JES defaults for region size, or changing the system IEFUSI exit to allow a larger virtual region.

>   If the utility is using BSAM data sets for which the number of buffers have been specified (BUFNO specified in the DCB field of the DD statement for the data set), consider reducing the number of buffers.

VIRTUAL STORAGE BELOW
>   If possible, increase the amount of virtual storage below the line available to the utility job by specifying a larger region size in the job JCL or reducing the number of data sets allocated in the job step. It might not be possible to relieve this constraint.

DD STATEMENTS

Define additional groups of sort work data sets, or eliminate the SWnnWKmm DD statements and specify the SORTDEVT keyword (and, optionally, the SORTNUM keyword) to allow SORT to use dynamic allocation of its work data sets.

If the limitation is due to sort message data sets rather than sort work data sets, define additional sort message DD statements or eliminate the UTPRINnn DD statements and allocate UTPRINT to SYSOUT so that the sort message data sets can be dynamically allocated.

CPUS Submit the job to a member that has more processors.

SWA=BELOW

Move the SWA above the 16 MB line for the JOBCLASS. JES command, $DJOBCLASS(n), will display whether or not SWA is set to BELOW for CLASS n. The $TJOBCLASS(n),SWA=ABOVE command will set SWA to ABOVE the line. The $DSTCCLASS and $TSTCCLASS commands can be used to move SWA to ABOVE for the utilities that are running in a WLM Stored Procedure Environment. Consult with the MVS System Programmer on these changes.

Severity: 0

DSNU398I *csect-name* **UNEXPECTED PROCESSING ERROR, REASON=X00E40318, ON TABLE** *table-name*

Explanation: When attempting to load a record into table *table-name*, an error was encountered.

System action: If the error was caused by a resource not being available, a DSNT500I message is issued.

The utility step is abended with a system X'04E' with an abend reason code of '00E40318'. If a DSNT500I message is not issued, an SVC dump is requested. A SYSABEND dump of batch storage is also requested.

The utility step is left in the stopped state.

Operator response: An operator with SYSOPR authority can issue a TERM UTIL to free any held resources (this may not be necessary).

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: If a DSNT500I message was issued, refer to its description under that message number in this manual. If the unavailable resource becomes available, the utility step is restartable.

If a DSNT500I message is not issued, the error is an internal error. Notify the system programmer.

Severity: 8 (error)

Related reference:

⬐ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

"DSNT500I" on page 555

⬐ 00E40318 (DB2 Codes)

DSNU399I *csect-name* **- LOAD UTILITY ERROR SUMMARY REPORT**

Explanation: This message summarizes errors detected during the load job. The report contains one line for each error. The errors are listed first by error severity and within that by *idsrn*.

The report fields have the following meanings:

sev

The severity of the error reported on this line:

Severity	Type of error	Explanation
1	Primary	Primary referential integrity errors and all nonreferential integrity errors.

DSNU400I

Severity	Type of error	Explanation
2	Secondary	Secondary referential integrity errors (that is, errors that result from a primary referential integrity error in an ancestor).

idsrn   The relative record number in the input data set of the record causing the error reported on this line.

ddsrn   The relative record number in the input data set of the record causing the error reported on this line, if discard processing is in effect. Otherwise, *ddsrn* is blank.

type   A description of the type of error reported on this line, as follows:

- Conversion
- Field specification
- Validation
- Part Specification
- Unexpected
- Column specification
- Referential integrity
- Check constraint
- Duplicate key
- Key evaluation error
- XML document error

name1   The name of the table for which the record causing the error was targeted, if this information is pertinent for the type of error reported on this line. Otherwise, *name1* is blank. This field has nonblank values for the following error types:

- Conversion
- Validation
- Column specification
- Referential integrity (primary).
- Check constraint

name2   The name of the field or fanset involved in the error reported on this line, if this information is pertinent for the type of error reported. Otherwise, *name2* is blank. This field contains a field name for the following types of errors:

- Conversion
- Field specification
- Column specification.

For primary referential integrity errors, *name2* is a relationship name. For check constraint errors, *name2* is a check constraint name.

idsrn2   The input data set record number of a related error. For duplicate key errors, it gives the input data set record number of the record whose key is duplicated. For secondary referential integrity errors, it gives the input data set record number of the record whose primary referential integrity error cascaded to cause this secondary error. For other types of errors, this field is unused.

System action: Processing continues.

Severity: 4 (warning)

DSNU400I   *csect-name* **COPY PROCESSED FOR** *object-type object-qualifier*.*object-name* **NUMBER OF PAGES=**total-pages **AVERAGE PERCENT FREE SPACE PER PAGE =** *free-space* **PERCENT OF CHANGED PAGES =** *changed-pages* **ELAPSED TIME=** *hh:mm:ss*

Explanation: This is an informational message to report the copy statistics. This message is issued by the following utilities:

- COPY

- LOAD, if a full image copy has been requested during a table space load
- REORG, if a full image copy has been requested during a table space reorganization

The statistics apply to pages that are included in the scope of the COPY request: a full table space, a full index space (DSNUM(ALL)), or a partition (DSNUM(integer)). All of the pages in the table space, index space, or partition can be copied (FULL(YES)), or only the changed pages can be copied (FULL(NO)).

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

total-pages
> The number of pages written out. The number of pages written includes valid instances where pages are written more than once, or for incremental copies, pages written which have not changed since the last copy was taken. A slight discrepancy between this value and the actual number of pages is possible and is considered normal.

free-space
> The average percentage of free space within the copied pages, in the format *nn.nn*. This value indicates the amount of change activity that can occur without significant impairment of reference performance.
>
> This value is 0 for sequential copies made from a FlashCopy.

changed-pages
> The percentage of changed pages, to two decimal places (*nn.nn* or 100.00). This value indicates the amount of change activity that occurred since the last image copy or reorganization. The percentage of changed pages is obtained by dividing the number of changed pages by the number of pages in the object. Preformatted pages consisting of all zeros are included in the count of pages scanned. Because these preformatted pages are not copied to the output data set, the number of pages scanned is usually larger than the number of pages copied.
>
> When this message is issued for image copies produced during LOAD or REORG, this percentage will always be reported as 100.00 (100%), because the utility is resetting and reformatting the entire table space.
>
> This value is 0 for sequential copies made from a FlashCopy.

hh:mm:ss
> The elapsed time, shown as the number of hours, minutes, and seconds.

System action: The COPY, LOAD or REORG utility continues processing. Message DSNU428I is issued when the copy processing is complete.

User response: No action is required.

You can use this information to help you make decisions about your strategies, such as:

- Use of REORG or RUNSTATS
- Your overall recovery strategy, including decisions about using incremental copies
- Storage allocation

Related reference:

RUNSTATS (DB2 Utilities)

REORG TABLESPACE (DB2 Utilities)

Related information:

"DSNU428I" on page 704

DSNU401I   *csect-name* **CONCURRENT COPY COMPLETE, ELAPSED TIME=** *hh*:*mm*:*ss*

Explanation: The DFSMS Concurrent Copy utility completed normally.

System action: Processing continues to the next utility statement in the utility job step. If there are no more utility statements, the utility job step is normally terminated, and the resources are released to their prior availability state. The utility identifier (UTILID) of the utility job step becomes available for reuse.

User response: No action is required.

Severity: 0 (informational)

DSNU402I   *csect-name* **INCREMENTAL IMAGE COPY DISALLOWED FOR** *object-type object-qualifier.object-name*
         FULL IMAGE COPY WILL BE TAKEN

Explanation: An incremental image copy for a table space or data set was requested. However, a full image copy is required.

csect-name
        The name of the control section that issued the message.

object-type
        The type of object.

object-qualifier.object-name
        The qualified object name.

Possible reasons why an incremental image copy is not allowed include the following situations:
* The request was for one of the following table spaces that are maintained by DB2:
    – DSNDB01.DBD01
    – DSNDB06.SYSCOPY
    – DSNDB01.SYSUTILX
    – DSNDB01.SYSDBDXA
* The LOAD utility or REORG TABLESPACE utility was run on the table space or data set, but no subsequent full image copy was taken.
* No full image copy was found for the requested sites.
* The requested table space or data set is in COPY-pending status.
* The request was for an entire table space, but the most recent image copy for the table space is at the data set level.
* The previous image copy is a DFSMS concurrent copy.

System action: The request for an incremental image copy is ignored, and a full image copy is taken.

User response: The size of the data set for the image copy might not be sufficient if the size was calculated for an incremental image copy. In that case, terminate the job. Then, specify a data set with the appropriate size for a full image copy, and resubmit the job. Otherwise, no action is necessary.

Severity: 4 (warning)

Related concepts:

⏏ Incremental image copies (DB2 Utilities)

Related reference:

⏏ COPY-pending status (DB2 Utilities)

⏏ COPY (DB2 Utilities)

DSNU403I   *csect-name* **- FULL AND INCREMENTAL COPIES OF THE SAME TABLESPACE ON THE SAME**
         TAPE DISALLOWED DDNAME *ddname*

Explanation: The COPY utility has been invoked to take an incremental image copy of a TABLESPACE. The

specified output data set is on a tape which contains another FULL or INCREMENTAL image copy of the same TABLESPACE. This is not allowed.

System action: The utility will terminate with a return code of 8, and will not take the image copy or update the SYSCOPY table.

User response: Change the VOLSER for the image copy output data set or use the MODIFY RECOVERY utility to delete SYSIBM.SYSCOPY records which are no longer needed for the same TABLESPACE and tape volume.

Problem determination: Use the SQL SELECT command to look at the SYSIBM.SYSCOPY table. If the conflicting image copy data sets on the output tape are no longer needed, use MODIFY RECOVERY to eliminate the related SYSIBM.SYSCOPY records. If the data sets are still needed, you must place the new image copy on a different tape or you must take a FULL image copy.

Severity: 8 (error)

DSNU404I   *csect-name* **LOCAL SITE AND RECOVERY SITE INCREMENTAL IMAGE COPIES ARE NOT SYNCHRONIZED FOR** *obj-type obj-qual.obj-name*

Explanation: No incremental image copy can be made if

1. The incremental image copy is requested only for a site other than the current site.

2. The incremental image copies are requested for both sites but the most recent full image copy was made only for the current site.

3. The incremental image copies are requested for both sites and the most recent full image copies were made for both sites, but between the most recent full image copy and current request, all incremental image copies were made for current site only.

System action: The utility will terminate with a return code of 8, and will not take the image copy or update the SYSCOPY table.

User response: Take another full image copy of the TABLESPACE for both LOCAL site and RECOVERY site or change the request to make incremental image copy only for current site.

Severity: 8(error)

DSNU406I   **FULL IMAGE COPY SHOULD BE TAKEN FOR BOTH LOCAL SITE AND RECOVERY SITE FOR** *obj-type obj-qual.obj-name*

Explanation: The requested full image is for one site only but history shows that there were image copies taken for both local site and recovery site and the table space is in copy pending status, or point in time recover has been done on the table space, or the table space is in copy pending status and the requested full image copy is for a site other than the current site. Be aware of the following situations:

1. If the requested copy is made for current site, then the other site can only do point in time recovery.

2. If the requested copy is for the site other than the current site, then the TABLESPACE will be in the same state as before COPY has been made.

System action: COPY, LOAD, or REORG utility will continue to process the image copy.

User response: Take another full image copy of the TABLESPACE for both LOCAL site and RECOVERY site to guarantee the correct recovery on both sites or check DSNZPARM to ensure that the current site is the site at which the image copy is being taken.

Severity: 4 (warning)

DSNU407I   *csect-name* **- DATA SET ALREADY USED FOR PREVIOUS IMAGE COPY DDNAME** *ddname*

Explanation: A COPY, LOAD, or REORG utility has been invoked to take an image copy to a data set which is a duplicate of one already recorded in SYSCOPY. If the specified data set is cataloged, a matching DSNAME exists. If the specified data set is not cataloged, a matching DSNAME, DSVOLSER, and FILESEQNO exists.

System action: The utility will terminate with a return code of 8, and will not take the image copy or update the SYSCOPY table.

User response: Change the name of the data set or use MODIFY utility to delete the old SYSCOPY record containing the duplicate data set.

DSNU408I • DSNU411I

Severity: 8 (error)

DSNU408I   *csect-name* **-** *dsn* **EXISTS WITH -VOLUME=***vol***, FILE SEQUENCE-NUMBER=***fseq***, RBA=***rba*|*lrsn***,**
       DATE=*date***, TIME=***time*

Explanation: If this message is issued after DSNU403I, it provides information from the SYSCOPY catalog table about the previously created image copy of the table space. If this message is issued after DSNU407I, it provides information from the SYSCOPY table about the duplicate data set that was detected during image copy processing.

For copies to DASD or MSS, *fseq* is 0; for copies of cataloged data sets, *vol* is (CATG).

System action: The utility terminates with return code 8 and does not take the image copy or update the SYSCOPY table.

User response: Use the information provided to change the name of the data set or to locate the duplicate SYSCOPY record for deletion with the MODIFY RECOVERY utility.

Severity: 8 (error)

Related concepts:

➡ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU409I   *csect-name* **NO** *hardware-or-software* **SUPPORT FOR** *obj-type obj-qual.obj-name*

Explanation: This message is issued by the COPY or RECOVER utility for one of the following reasons.
- The COPY utility was invoked to take a DFSMS concurrent copy of a table space or index space that resides on a device with no hardware or software support for the DFSMSdss DUMP command with the CONCURRENT option.
- The RECOVER utility invoked the DFSMSdss RESTORE command without the supporting software.

System action: The RECOVER utility falls back to the previous image copy data set and continues. The COPY utility job terminates.

User response: If a list of objects is specified in the COPY job, remove the object that has no hardware or software support from the list. Resubmit the job or use the DB2 COPY utility instead of the DFSMS Concurrent Copy utility.

Severity: 4 (for RECOVER) or 8 (for COPY)

DSNU410I   *csect-name obj-type obj-qual.obj-name* **CONTAINED NO CHANGED PAGES**

Explanation: An incremental image copy (COPY option FULL(NO)), which copies only changed pages, was requested. However, no changed pages were found.

System action: This message indicates normal completion of the image copy utility. Processing continues to the next utility statement in the utility job step, if any. If there are no more utility statements, the utility job step is normally terminated, and the resources are released to their prior availability state. The UTILID of the utility job step becomes available for reuse.

Because no output was produced, the SYSIBM.SYSCOPY catalog table remains unchanged.

The requester may want to determine why the apparent assumption that a change occurred was not true.

Operator response: The target copy data set identified by the DD statement nominated by the COPY COPYDDN option was not required. It was not tracked in the DB2 catalog table SYSIBM.SYSCOPY. It need not be kept for subsequent re-call.

Severity: 4 (warning)

This message reflects a warning condition that did not prevent subsequent processing.

DSNU411I   *csect-name* **UNABLE TO LOGICALLY COMPLETE** *obj-type obj-qual.obj-name* **RETURN CODE**
       return-code **REASON CODE** *reason-code*

Explanation: The DFSMSdss DUMP command with the CONCURRENT option encountered an error during the initialization of the target object being copied. The return code and reason code are from the DFSMSdss DUMP command.

System action: The utility job terminates.

User response: See the return code and reason code in *z/OS DFSMSdss Storage Administration Reference* to determine the problem. Resubmit the job after resolving the problem.

Severity: 8

DSNU412I   *csect-name* **UNABLE TO PHYSICALLY COMPLETE** *obj-type obj-qual.obj-name* **RETURN CODE** *return-code*

Explanation: The DFSMSdss DUMP command with the CONCURRENT option encountered an error during the backup process. The return code is from the DFSMSdss DUMP command.

System action: The utility job terminates.

User response: See the return code description in *z/OS DFSMSdss Storage Administration Reference* to determine the problem. Resubmit the job after resolving the problem, or use the DB2 Copy utility instead of the DFSMS Concurrent Copy utility for the failed object.

Severity: 8

DSNU413I   *csect-name* **- CONCURRENT COPY SUCCESSFUL FOR** *obj-type obj-qual.obj-name*

Explanation: The DFSMS Concurrent Copy utility completed normally for the named object. This message is repeated for every copied object in the list.

System action: Processing continues to the next object in the list. If there are no more objects, the utility ends normally with message DSNU401I.

User response: No action is required.

Severity: 0 (informational)

DSNU414I   *csect-name* **UNABLE TO RETRIEVE RESTART/REDO LRSN** *obj-type obj-qual.obj-name*

Explanation: This message is issued by the DB2 COPY utility when an error is encountered during the retrieval of a log record sequence number (LRSN).

This error can occur in these situations:
- Another DB2 in a data sharing group terminated abnormally while trying to update the table space or index space.
- A failure occurred while trying to communicate with other DB2s in the data sharing group.

System action: The utility job terminates.

User response: Issue a -DISPLAY DATABASE command with the USE option to determine if any other DB2 systems hold a retained lock on the table space or index space (status = RSTRT). If so, you must restart that DB2 system before the COPY job can be successfully run.

Severity: 8

DSNU415I   *csect-name* **INVALID OUTPUT BLOCK SIZE SPECIFIED DDNAME** *ddname*

Explanation: Valid block sizes for an image copy or mergecopy data set are multiples of 4KB and less than the device limitation.

System action: Utility processing is terminated.

User response: Correct the JCL statement, or use the default. Resubmit the request in a new utility job step.

Severity: 8 (error)

Related concepts:

DB2 online utilities (DB2 Utilities)

DB2 stand-alone utilities (DB2 Utilities)

DSNU416I   *csect-name* **THE OUTPUT IMAGE COPY DATA SET CANNOT BE PARTITIONED. DDNAME**
ddname

Explanation: The use of partitioned image copy data sets is restricted. The COPY, MERGECOPY, LOAD, or REORGutility job that was submitted has a partitioned output image copy data set, specified by the COPYDDN ddname or RECOVERYDDN ddname parameter.

System action: Utility processing is terminated. The image copy will not be taken or the merge will not be done, and the SYSIBM.SYSCOPY table will not be updated.

User response: Correct the utility job by specifying a nonpartitioned data set as the output image copy data set.

Severity: 8 (error)

DSNU417I   *csect-name* **THE NUMBER OF VOLUMES USED BY THE IMAGE COPY DATA SET** *data-set-name*
HAS REACHED THE LIMIT

Explanation: The image copy data set reached the limit of 255 volumes and another volume is required.

System action: The utility job is terminated.

User response: You might want to use the DSNUM option of the COPY utility to identify a partition or data set within the table space to be copied, or specify a partition number to be loaded or reorganized if the copy is being produced from a LOAD or REORG.

Severity: 8 (error)

Related reference:

⇥ Syntax and options of the COPY control statement (DB2 Utilities)

⇥ Syntax and options of the LOAD control statement (DB2 Utilities)

⇥ Syntax and options of the REORG INDEX control statement (DB2 Utilities)

⇥ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

DSNU418I   *csect-name* **CANNOT TAKE IMAGE COPY AT DATASET LEVEL FOR NONPARTITIONED**
TABLESPACE *obj-qual.obj-name* **IN COPY PENDING STATUS**

Explanation: An image copy at the data set level was requested. However, it is not allowed for one of the following reasons:

- The table space is in copy pending status.
- The table space is not suitable for image copy at the data set level.

A copy of the entire table space is required.

System action: The COPY utility completes unsuccessfully with return code 8.

System programmer response: You must take a full image copy of the entire table space by specifying the DSNUM ALL option with the COPY utility.

Severity: 8 (error)

DSNU419I   *csect-name* **UTILITY CANNOT PROCESS** *obj-typeobj-qual.obj-name* **WITH MIXED SPECIFIC AND**
NONSPECIFIC VOLUME IDS.

Explanation: The requested object cannot be processed, because it was defined to use a DB2 storage group with mixed specific and nonspecific volume IDs. A specific volume ID is represented by a string with a maximum of 6 characters. A nonspecific volume ID is represented by an asterisk (*).

csect-name
        The name of the control section that issued the message.

obj-type   The type of object.

obj-qual.obj-name
        The qualified object name.

702   Messages

System action: The utility completes unsuccessfully with return code 8.

System programmer response: Use the ALTER STOGROUP command to change the storage group to contain only specific or nonspecific volume IDs and rerun the job.

Severity: 8 (error)

DSNU420I *csect-name* **INCORRECT SHARE LEVEL SPECIFIED FOR TABLE SPACE** *database-name***.***tablespace-name* **DSNUM** *dsnum*

Explanation: When copying a table space that is not recoverable, the share level (SHRLEVEL) must be REFERENCE.

If the object is an entire table space, 'DSNUM *dsnum*' is not displayed.

System action: Utility processing is terminated.

User response: Correct the SHRLEVEL to REFERENCE, and resubmit the COPY utility in a new utility job step.

Severity: 8 (error)

Related reference:

➡ COPY (DB2 Utilities)

➡ Syntax and options of the COPY control statement (DB2 Utilities)

DSNU421I *csect-name* **- START OF DFSMS MESSAGES**

Explanation: This message indicates the beginning of the DFSMS messages; DSNU422I indicates the end.

System action: No action is required, but subsequent DFSMS messages might require action. See the DFSMS documentation for further information.

DSNU422I *csect-name* **- END OF DFSMS MESSAGE**

Explanation: This message indicates the end of the DFSMS messages; DSNU421I indicates the beginning.

System action: No action is required, but preceding DFSMS messages might require action. See the DFSMS documentation for further information.

DSNU423I *csect name* **CANNOT TAKE CONCURRENT COPY WITH SHRLEVEL CHANGE OPTION FOR** *obj-type obj-qual.obj-name* **WITH GREATER THAN 4KB PAGE SIZE**

Explanation: The DFSMS Concurrent Copy utility is serialized at the extent level. The greater than 4KB page size condition might span more than one extent. This might cause the output data set to contain inconsistent data.

System action: The COPY utility completes unsuccessfully with return code 8.

System programmer response: You must take a concurrent copy with the SHRLEVEL REFERENCE option or a DB2 image copy if you prefer the SHRLEVEL CHANGE option.

Convert the data set attribute Control Interval size of the associated tablespace from the 4K Control Interval into the one that matches the page size. DB2 Utilities REORG Tablespace, LOAD with REPLACE, and RECOVER can be used to convert the table space.

Severity: 8 (error)

DSNU424I *csect-name* **INCOMPATIBLE MAINTENANCE DETECTED. REFERENCE APAR NUMBER** *apar* **DO THE FOLLOWING:**

Explanation: This message indicates that there is a maintenance incompatibility detected across restart. This message is followed by the actions the user should take to resolve the incompatibility.

apar The number of the APAR with closing text that describes the incompatibility.

System action: Utility execution is abended.

System programmer response: Refer to the APAR closing text.

DSNU425I • DSNU428I

DSNU425I *csect-name obj-type obj-qual.obj-name* **DOES NOT HAVE THE COPY YES ATTRIBUTE**

Explanation: The utility cannot process the identified index or index space because it was not defined with the COPY YES attribute. Without the COPY YES attribute, the index cannot be specified as a target for the RECOVER, COPY, or REPORT utilities. If the index needs to be rebuilt from the data in the table space, use the REBUILD INDEX utility instead.

System action: The RECOVER utility terminates processing with a return code of 8. The COPY utility skips this object but continues processing the other objects in the list. The final COPY utility return code will be 8. The REPORT utility issues this message with a return code of 4 and continues processing other objects (if any).

User response: Determine whether the index should be altered to have the COPY YES attribute. If so, alter the index then rerun the utility job. Otherwise remove the index from the utility statement.

Severity: 4 (warning) or 8 (error)

DSNU426I *csect-name* **CANNOT COPY ONE PIECE OF NON-PARTITIONING** *obj-type obj-qual.obj-name*

Explanation: The copy of a nonpartitioning index at the data set level is not allowed.

System action: The COPY utility skips this object but continues processing the other objects in the list. The final COPY utility return code will be 8.

User response: Either remove the DSNUM option so that it defaults to DSNUM(ALL) or modify your DSNUM option to DSNUM(ALL).

Severity: 8 (error)

DSNU427I *csect-name* **OBJECTS WILL BE PROCESSED IN PARALLEL, NUMBER OF OBJECTS =** *maximum-objects*

Explanation: This message indicates the maximum degree of parallelism that the COPY or RECOVER utility is able to process.

csect-name
> The name of the control section that issued the message.

maximum-objects
> The maximum number of objects that can be processed in parallel, based on available resources. This value can differ from the actual number of objects that are processed in parallel. For example, if 100 objects are to be recovered and the image copies are on a single tape, then only one object can be processed at a time.

System action: Processing continues.

User response: No action is required.

You can verify that the number of objects processed is consistent with expectations.

Severity: 0

Related reference:

➡ COPY (DB2 Utilities)

➡ RECOVER (DB2 Utilities)

DSNU428I *csect-name* **DB2 IMAGE COPY SUCCESSFUL FOR** *obj-type obj-qual.obj-name*

Explanation: The image copy of the object processed by the COPY, LOAD, or REORG utility has completed successfully and the image copy data set information has been recorded in the SYSIBM.SYSCOPY catalog table.

System action: The COPY utility continues its processing on the next object in the list. The LOAD and REORG utilities continue processing in the subsequent phases.

DSNU429I   *csect-name* **RECOVERY OF** *objecttype database-name.tablespace-name* **MUST BE EXECUTED AT THE
PARTITION LEVEL**

Explanation: You must recover the partitioned object at the partition level. You may receive this message if you are
using COPY with the CONCURRENT option. The use of online REORG (SHRLEVEL REFERENCE or CHANGE) has
created a mixture of 'I0001' and 'J0001' instance nodes (fifth qualifier) for the underlying LDS's for the partitions of
this object.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU430I   *csect-name* **- ERROR ALLOCATING TEMPORARY FILTER FOR** *obj-type obj-qual.obj-name* **DSN=***name*
CODE=X'*reason-code***'**

Explanation: SVC 99 returned an error during the dynamic allocation of a temporary filter data set. This message is
followed by text that explains the reason-code that is shown.

The COPY utility was specified with the CONCURRENT option, and has detected that the number of data sets to be
processed will exceed the DFSMSdss DUMP command limitation of 255. The utility will dynamically allocate a
temporary filter data set for the object and generate a DFSMSdss DUMP command with the FILT option.

System action: Utility processing halts for this object.

The utility job completes with a return code of 8.

If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility
will process the next item in the list.

User response: Correct the error that is preventing dynamic allocation, and resubmit the utility job. In some cases
where dynamic allocation may not be possible, consider using the FILTERDDN option on the COPY utility statement.

Severity: 8 (error)

DSNU431I   *csect-name* **- ERROR ALLOCATING TEMPORARY FILTER FOR** *obj-type obj-qual.obj-name*
DSNUM=*name* **CODE=X'***reason-code***'**

Explanation: An error returned during the dynamic allocation of a temporary filter data set. This message is
followed by text that explains *reason-code*.

The RECOVER utility was run with the CURRENTCOPYONLY option and has detected that the number of data sets
to be processed exceeds the DFSMSdss RESTORE command limitation of 255. DB2 dynamically allocates a temporary
filter data set for the object and generates a DFSMSdss RESTORE command with the FILT option.

System action: If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP)
option, the utility processes the next item in the list. Otherwise, utility processing stops with return code of 8.

User response: Correct the error that prevents dynamic allocation and resubmit the utility job. If dynamic allocation
is not possible, remove be possible, remove the CURRENTCOPYONLY option on the RECOVER utility statement.

Severity: 8 (error)

DSNU432I   *csect-name* **- ERROR DURING DFDSS RESTORE OBJECT FOR** *obj-type obj-qual.obj-name***. THE
OBJECT IS NOT RECOVERED**

Explanation: The RECOVER utility received an error notification from DFdss on object *obj-qual.obj-name*. The object
has not been restored, but recovery of the other objects continues.

System action: RECOVER processes the other objects.

When RECOVER completes, it displays all objects for which CURRENTCOPYONLY was specified.

User response: Remove the CURRENTCOPYONLY keyword from the utility control statements.

Severity: 4 (warning)

DSNU434I *csect-name* **- TEMPLATE** *template-name* **HAS THE STACK YES OPTION ON** *dd-name* **DATA SET WHICH IS NOT SUPPORTED FOR** *utility-name***.**

Explanation: Template processing has detected tape stacking on an output data set which is not supported.

System action: If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility processes the next item in the list. The utility will end with RC8 and subsequent utility statements in the job step are not executed. Otherwise, if EVENT(ITEMERROR,HALT) is specified or defaulted to, the utility processing stops immediately with a return code of 8 and subsequent utility statements in the job step are not executed.

User response: Remove the STACK YES option from the TEMPLATE statement, then resubmit the job.

Severity: 8 (error)

DSNU437I **RESTART OF COPY ON** *object-type object-qualifier.object-name DSNUM n* **IS NOT ALLOWED**

Explanation: Restart of a COPY utility on the identified object is not allowed because a REORG utility reorganized the data.

object-type
> The type of object to be copied.

object-qualifier.object-name
> The qualified name of the identified object in the COPY utility.

DSNUM n
> The data set number of the object. *DSNUM n* is omitted from the message text if:
>
> • DSNUM is not specified on the COPY utility statement.
>
> • DSNUM ALL is specified on the COPY utility statement.

System action: The utility halts processing for this object. If the utility job step includes an OPTIONS control statement with EVENT(ITEMERROR,SKIP) the utility processes the next item in the list. Otherwise, utility processing terminates.

System programmer response: Run the REPORT RECOVERY utility for the object. Examine the REPORT RECOVERY output. Submit another COPY utility to create a backup of the object needed.

Severity: 8 (error)

Related reference:

➡ Concurrency and compatibility for COPY (DB2 Utilities)

DSNU440I *csect-name* **IMAGE COPY INFORMATION**

Explanation: This message gives statistical information about the results of running the COPY utility. The information applies to a table space, partition, or data set, and is based on the information in the space page map at the time the utility is executed.

csect-name
> The name of the control section that issued the message.

This message contains the following information:

DBNAME
> The name of the database to which the table space belongs.

TSNAME
> The table space that is specified in the utility statement.

DSNUM
> The partition or data set number of the table space. The value is ALL if information is about the entire table space.

*nn***KB PAGES**
> The total number of *nn*KB pages in the partition, data set, or entire table space. This value includes preformatted pages that contain all zeros.

EMPTY PAGES

The total number of empty pages in the data set of the segmented table space, or entire segmented table space. Empty pages can result from deletes, or are pages that are not yet formatted. They are not copied by the utility.

CHANGED PAGES

The number of pages in the partition, data set, or table space that were changed since the last time an image copy was created.

PERCENT OF CHANGED PAGES

The percent of changed pages in the table space, data set, or partition specified.

ICTYPE

The type of image copy recommended for the partition, data set, or table space specified. ICTYPE is based on the comparison between the percent of changed pages and the change limit values. ICTYPE values:

I   If the percent of changed pages is greater or equal to the low CHANGELIMIT value and less than the high CHANGELIMIT value. An incremental image copy is recommended, or COPY takes one if the REPORTONLY option is not specified.

F   If the percent of changed pages is equal to or greater than the high CHANGELIMIT value. A full image copy is recommended, and COPY takes one if the REPORTONLY option is not specified.

NONE If no pages were changed. An image copy is not recommended, and COPY does not take one.

blank For the partition or data set if DSNUM ALL is specified, or if CONCURRENT is specified and the percent of changed pages is less than the CHANGELIMIT value.

System action: Utility processing continues

User response: If the message indicates, "INFORMATION CANNOT BE DETERMINED" for a table space, partition, or data set, examine the utility output for the DSNU441I message which lists the page numbers of the broken pages. See message DSNU441I for further instructions on repairing pages.

Severity: 0

Related reference:

➡ COPY (DB2 Utilities)

Related information:

"DSNU441I"

DSNU441I   **csect-name BROKEN PAGE FOUND DBNAME = dbname TSNAME = tsname DSNUM = dsnum PAGE = page ERRORCODE =** *error-code*

Explanation: The COPY utility found a broken space map page.

dbname

The name of the database to which the table space belongs.

tsname   The table space containing the broken page.

dsnum   The partition or data set number of the table space containing the broken page.

page   The hexadecimal page number of the broken page.

error-code

The specific cause of the problem. Only some CSECT modules issue an error code. If the CSECT module does not issue an error code, the message that you receive will not list the ERRORCODE keyword and value.

System action: Utility execution continues.

User response: The data sets identified are in an indeterminate state. Use RECOVER, RECOVER PAGE, or the REPAIR utility to correct the broken page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU442I *csect-name* **CHANGELIMIT REPORT INFORMATION UNAVAILABLE FOR** *obj-type obj-qual.obj-name*
WITH TRACKMOD NO

Explanation: This message is issued in response to a COPY utility. The CHANGELIMIT option cannot produce
statistical information for a table space or a partition that is defined with TRACKMOD NO, or before a copy has
been taken after the TRACKMOD attribute has been changed from NO to YES.

System action: If REPORTONLY is not specified then a full image copy will be taken.

User response: To obtain complete statistical information run CHANGELIMIT REPORTONLY after a copy has been
taken.

Severity: 4 (informational)

DSNU443I *csect-name type* **CHANGE LIMIT MET FOR** *obj-type obj-qual.obj-name*

Explanation: The COPY utility is indicating whether the percent of changed pages in the specified table space,
partition, or data set meets the CHANGELIMIT values. *type* values:

NO If no CHANGELIMIT met.

INCREMENTAL
If the percent of changed pages is greater than the low CHANGELIMIT value and less than the high
CHANGELIMIT value.

FULL If the percent of changed pages is greater than or equal to the the high CHANGELIMIT value.

System action: Utility processing continues.

User response: If the REPORTONLY option is specified and the result of this job step is not used to conditionally
execute other job steps, examine the utility messages for this table space, partition, or data set to determine if it has
changed to such a degree that an image copy or other step should be performed.

Severity: The return codes are:

1 (informational)
If no CHANGELIMIT was met.

2 (informational)
If the percent of changed pages is greater than the low CHANGELIMIT and less than the high
CHANGELIMIT value.

3 (informational)
If the percent of changed pages is greater than or equal to the high CHANGELIMIT value.

DSNU445I *csect-name ictype* **IMAGE COPY WILL BE TAKEN FOR** *obj-type obj-qual.obj-name*

Explanation: The COPY utility is indicating whether the percent of changed pages in the table space meets the
change limit value, and whether an image copy will be taken. *ictype* values:

NO If no CHANGELIMIT value met.

INCREMENTAL
If the percent of changed pages is greater than the low CHANGELIMIT value and less than the high
CHANGELIMIT value.

FULL If the percent of changed pages is greater than or equal to the high CHANGELIMIT value.

System action: Utility processing continues. An image copy is taken if a change limit is met. If the low
CHANGELIMIT value is met, then an incremental copy is taken. If the high CHANGELIMIT value is met, then a full
image copy is taken.

Severity: 0 (informational)

DSNU446I   *csect-name* **- CHANGELIMIT REPORT INFORMATION UNAVAILABLE FOR** *object-type*
*object-qualifier.object-name***, A FULL IMAGE COPY MUST BE TAKEN**

Explanation: The COPY utility was run with the CHANGELIMIT option. However, the required data set statistical information was not available. To use the CHANGELIMIT option, a full image copy must exist for the target object.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The qualified object name.

Possible reasons why the statistical information is not available include:

- The request was for one of the following table spaces that are maintained by DB2: DSNDB01.DBD01, DSNDB06.SYSCOPY, DSNDB01.SYSUTILX, or DSNDB01.SYSDBDXA.
- The LOAD utility or REORG TABLESPACE utility was run on the table space or data set, but no subsequent full image copy was taken.
- No full image copy was found for the requested sites.
- The requested table space or data set is in COPY-pending status.
- The request was for an entire table space, but the most recent image copy for the table space is at the data set level.
- The previous image copy is a DFSMS concurrent copy.
- The previous image copy is a FlashCopy image copy.
- The previous image copies are a FlashCopy image copy and sequential format image copies that were generated in the same execution of the COPY utility.
- The previous image copies are sequential format image copies that were generated from a FlashCopy image copy by the COPYTOCOPY utility.

System action: If the COPY statement included the REPORTONLY option, the statistical information report is not generated. If the REPORTONLY option was omitted, a full image copy is taken.

User response: If REPORTONLY was specified, submit another COPY job to take a full image copy.

Severity: 4 (warning)

Related concepts:

➥ Specifying conditional image copies (DB2 Utilities)

Related reference:

➥ Syntax and options of the COPY control statement (DB2 Utilities)

➥ COPY-pending status (DB2 Utilities)

DSNU447I   *csect-name* **- COPY SHRLEVEL CHANGE OF** *obj-type obj-qual.obj-name* **IS NOT ALLOWED BECAUSE IT HAS A LOGGING ATTRIBUTE OF NOT LOGGED**

Explanation: COPY with SHRLEVEL CHANGE was specified for a COPY utility job that refers to an object with a NOT LOGGED logging attribute. This situation is not supported.

System action: Utility processing halts for this object. The utility job completes with a return code of 8.

If the utility job step include an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Remove the SHRLEVEL CHANGE option from the utility control statement.

Severity: 8 (error)

DSNU448I   *csect-name* **- SCOPE PENDING WAS SPECIFIED, BUT NO OBJECT WITH PENDING STATUS WAS FOUND**

Explanation: SCOPE PENDING was specified for the utility job, however no objects existed in the list with a pending status.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU449I   *csect-name* **- THE NOT LOGGED INDEX** *obj-qual.obj-name* **IN ICOPY PENDING STATE MUST BE COPIED WITH ITS TABLE SPACE IN THE SAME COPY INVOCATION**

Explanation: A NOT LOGGED index that is in ICOPY pending state must be copied together with its table space in the same COPY invocation. An image copy of such an index made without its table space cannot be used for recovery of the index to a point consistent with its table space because the recovery would require applying log records, which were not written. Therefore, such an image copy is not allowed.

System action: Utility processing halts for this object. The utility job completes with a return code of 8.

If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Include the index's table space in the COPY list.

Severity: 8 (error)

DSNU454I   *csect-name* **COPY MERGE COMPLETE. NUMBER OF COPIES =** *iiiii*, **NUMBER OF COPIES MERGED =** *cccccc*, **TOTAL NUMBER OF PAGES MERGED =** *ttttt*, **ELAPSED TIME =** *hh/mm/ss*

Explanation: This message marks the normal completion of the MERGECOPY utility. The variables are:

iiiii   The number of copies found.

cccccc The number of image copies that were merged. The number merged can be less than the number found if a work data set was not specified using the MERGECOPY WORKDDN option.

ttttt   The total number of pages that were merged.

The merged incremental copies were consolidated either into a single new incremental image copy or into a new full image copy, depending upon the MERGECOPY NEWCOPY option. The DB2 catalog table SYSIBM.SYSCOPY was updated to reflect the results.

This message may indicate that prior data set entries were deleted and, consequently, that backup data set resources were freed.

System action: Processing continues to the next utility statement in the utility job step, if any. If there are no more utility statements, the utility job step is normally terminated, and the resources are released to their prior availability state. The UTILID of the utility job step becomes available for reuse.

User response: You can review the reported statistics to assess the effectiveness of performing incremental image copies. You can examine the SYSIBM.SYSCOPY catalog table to review the incremental and full image copies that are subject to DB2 recovery.

The full or incremental image copy data set identified by the DD statement specified in the COPYDDN option was recorded in SYSIBM.SYSCOPY. DB2 assumes that the image copy data set is now available for subsequent recovery. Ensure that the data set is available for recovery.

Severity: 0/4 (informational/warning)

DSNU458I   *csect-name* **NO IMAGE COPIES TO MERGE**

Explanation: This message is issued in response to a MERGECOPY request. The MERGECOPY utility was invoked to merge existing incremental image copies. However, in checking the DB2 catalog table SYSIBM.SYSCOPY, no incremental image copies were found.

System action: This message indicates the completion of the MERGECOPY utility. Processing continues to the next

utility statement in the utility job step, if any. If there are no more utility statements, the utility job step is normally terminated, and the resources that were held become available for REFERENCE processing. The UTILID of the utility job step becomes available for reuse.

The requester may want to determine why the assumption that incremental copies existed was not true.

Operator response: The target copy data set identified by the DD statement nominated by the MERGECOPY COPYDDN option was not required. It was not tracked in the DB2 catalog table SYSIBM.SYSCOPY. It need not be kept for subsequent re-call.

Severity: 4 (warning)

A severity 4 condition is given to alert the invoker that processing did not proceed as defined. However, processing is proceeding; nothing was encountered that would invalidate further processing.

DSNU459I   **SYSCOPY P RECORD ENCOUNTERED FOR** *object-type object-qual.object-name*, **PIT RBA =** *xxxxxx*

Explanation: The utility encountered a SYSCOPY record that contains a 'P' value in the ICTYPE column. This value requires that the SYSCOPY record at the given RBA or LRSN be used.

System action: The utility properly traverses the SYSCOPY records following this SYSCOPY P record.

User response: No action is required.

Severity: 0 (informational)

Related concepts:

➤ Where to find recovery information (DB2 Administration Guide)

➤ How RBA and LRSN values are displayed (DB2 Administration Guide)

Related tasks:

➤ Discarding SYSCOPY and SYSLGRNX records (DB2 Administration Guide)

Related reference:

➤ SYSIBM.SYSCOPY table (DB2 SQL)

DSNU460I   *csect-name* **IMAGE COPIES INCONSISTENT. MERGECOPY REQUEST REJECTED**

Explanation: MERGECOPY was requested for an entire table space. However, the most recent image copy for the associated table space was on the data set level. In addition, if the associated table space was partitioned, this might indicate that any or all of its partitions were being reorganized or loaded without logging after the MERGECOPY starting point. The MERGECOPY starting point is the most recent full image copy for the entire table space.

System action: This message indicates the completion of the MERGECOPY utility. Processing continues to the next utility statement in the utility job step, if any. If there are no more utility statements, the utility job step is normally terminated, and the resources that were held become available for REFERENCE processing. The UTILID of the utility job step becomes available for reuse.

No output was produced and the SYSIBM.SYSCOPY catalog table remains unchanged.

User response: A full image copy of the associated table space should be taken. However, if it is preferred that a recovery be done on the data set level, a MERGECOPY job should be submitted by specifying 'DSNUM(data set number)' for each data set for which MERGECOPY is required.

Severity: 4 (warning)

A severity 4 condition is given to alert the user that processing did not proceed as defined. However, processing is proceeding; nothing was encountered that would invalidate further processing.

DSNU461I   *csect-name* **PRIMARY FULL COPY DATA SET NOT FOUND, BACKUP WILL BE USED INSTEAD**

Explanation: The primary full image copy cannot be found. The backup full image copy will be located and used for the merge process.

User response: If the MERGECOPY utility can use the backup full image copy, the utility continues to process. If no backup full image copy can be used, the utility terminates.

DSNU462I • DSNU464I

Severity: 0 (informational)

DSNU462I   *csect-name* **MERGECOPY DISALLOWED FOR REQUESTED TABLE SPACE**

Explanation: The MERGECOPY utility was requested for table space DSNDB01.DB01, DSNDB01.SYSUTILX, DSNDB06.SYSCOPY, or DSNDB01.SYSDBDXA.

MERGECOPY is not permitted, because only full image copies are allowed on these table spaces that are maintained by DB2. Also, no record of the full image copies exists in the SYSIBM.SYSCOPY catalog table; the copy information is in the DB2 log.

csect-name
> The name of the control section that issued the message.

System action: Further utility processing is terminated.

System programmer response: No action is required.

You do not need to merge copies for these table spaces, because no incremental image copies exist.

Severity: 8 (error)

Related reference:

➡ MERGECOPY (DB2 Utilities)

DSNU463I   *csect* **- THE** *xxxxxxx* **IMAGE COPY DATA SET** *data-set-name* **WITH DATE=**`dddddd` **AND TIME=**`tttttt`,
          IS PARTICIPATING IN MERGECOPY.

Explanation: This informational message indicates the image copy that is used in the merge process. The variables have the following meanings:

xxxxxxx
> PRIMARY or BACKUP, indicating a primary or backup image copy

data-set-name
> The image copy data set name

dddddd
> The image copy date

tttttt   The image copy time

User response: The utility continues to process normally.

Severity: 0 (informational)

DSNU464I   **THIS IS A** *site-type*; *output-type* **WITH NEWCOPY NO IS NOT ALLOWED.**

Explanation: The 'NEWCOPY NO' option to create incremental image copies is not supported for the current site. This error might be caused by the following situations:

- Running the MERGECOPY utility at *site-type* LOCALSITE, but selecting the RECOVERYDDN with the 'NEWCOPY NO' option.
- Running the MERGECOPY utility at *site-type* RECOVERYSITE, but selecting the COPYDDN with the 'NEWCOPY NO' option.

System action: The command cannot be executed.

User response: Correct the error and resubmit the job.

Severity: 8 (error)

Related concepts:

➥ Full or incremental image copy (DB2 Utilities)

Related reference:

➥ MERGECOPY (DB2 Utilities)

DSNU470I **TABLESPACE DSNDB01.SYSUTILX, ITS TABLES AND INDEXES CANNOT BE INCLUDED IN A LIST OF OBJECTS.**

Explanation: This message is issued by the running utility when a control statement contains a list of objects, or references a LISTDEF, which includes any of the following SYSUTILX related objects: TABLESPACE DSNDB01.SYSUTILX, INDEXSPACE DSNDB01.DSNLUX01, INDEXSPACE DSNDB01.DSNLUX02, TABLE SYSADM.SYSUTIL, TABLE SYSADM.SYSUTILX, INDEX SYSADM.DSNLUX01 or INDEX SYSADM.DSNLUX02.

System action: The utility job terminates.

User response: Remove the invalid object from the control statement and run the utility against DSNDB01.SYSUTILX, its tables and indexes separately.

Severity: 8(error)

DSNU471I *csect-name* **- COPY PENDING ON TABLESPACE** *databasename.tablespacename* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility against a table space. Either the table space was in copy pending state or DSNDB06.SYSCOPY was in copy pending state.

System action: Utility processing is terminated.

User response: Remove the table space from the copy pending state and resubmit the job.

Severity: 8 (error)

Related tasks:

➥ Backing up and recovering your data (DB2 Administration Guide)

DSNU472I *csect-name* **- TABLESPACE** *tablespace-name* **HAS WRITE ERROR RANGE OR IS IN DEFERRED RESTART.**

Explanation: The attempt to externalize table space 'tablespace-name' pages to the media has failed.

System action: QUIESCE still continues but you cannot make an image copy of the table space. Processing continues for all indexes within the table space and continues to the next table space in the tablespace-list, if any.

User response: Run DISPLAY DATABASE to determine what exception state, if any, would prevent image copy on the table space.

Severity: 4 (warning)

Related concepts:

➥ Deferral of restart (DB2 Administration Guide)

Related tasks:

➥ Backing up and recovering your data (DB2 Administration Guide)

DSNU473I *csect-name* **- UNABLE TO EXTERNALIZE TABLESPACE** *tablespace-name* **PAGES**

Explanation: The attempt to externalize pages from the specified table space to the media failed.

csect-name
      The name of the control section that issued the message.

tablespace-name
      The name of the table space.

System action: The QUIESCE utility continues processing, but the table space cannot be volume-dumped.

Problem determination: The problem might be an I/O error.

Severity: 4 (warning)

Related concepts:

�ढ़ Reasons why QUIESCE fails to write to disk (DB2 Utilities)

DSNU474I   *csect-name* **- QUIESCE AT RBA** *rrrrrrrrrrr* **AND AT LRSN** *xxxxxxxxx*

Explanation: This message displays the log RBA and LRSN at the time of the quiesce. This value is unique for all quiesced table spaces in the list and is recorded in the START_RBA column of the SYSCOPY catalog table entry for each quiesced table space in the list.

System action: Processing continues.

Severity: 0 (informational)

Related concepts:

▱ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU475I   *csect-name* **- QUIESCE UTILITY COMPLETE, ELAPSED TIME=***tttttt*

Explanation: The message marks the normal completion of the QUIESCE utility. The elapsed clock time that the QUIESCE utility uses is shown by 'tttttt'.

System action: Processing continues to the next utility statement in the utility job step, if any. If there are no more utility statements, the utility job step is normally terminated, and the resources are released to their prior available state. The UTILID of the utility job step becomes available for reuse.

Severity: 0 (informational)

DSNU476I   *csect-name* **- INDEX** *index-name* **HAS WRITE ERROR RANGE OR IS IN DEFERRED RESTART**

Explanation: The attempt to externalize index *index-name* pages to the media failed.

System action: QUIESCE processing continues.

User response: Issue the DISPLAY DATABASE command to determine the exception state,

Severity: 4 (warning)

Related concepts:

▱ Deferral of restart (DB2 Administration Guide)

Related tasks:

▱ Backing up and recovering your data (DB2 Administration Guide)

Related reference:

▱ -DISPLAY DATABASE (DB2) (DB2 Commands)

DSNU477I   *csect-name* **- QUIESCE SUCCESSFUL FOR** *obj-type obj-qual.obj-name*

Explanation: The QUIESCE for this object was successful. If the object is a table space, then the quiesce for all of its indexes was successful. This message will also be issued for each index with the COPY YES attribute over the quiesced table space. This message will be repeated for every quiesced table space in list.

System action: All data held in the DB2 buffer pool(s) for the table space and its indexes have been written out to the media if no message DSNU473 or DSNU479 were issued earlier. A SYSCOPY record of ICTYPE 'Q' has been recorded in the SYSCOPY catalog table for the table space and for each of its indexes with the COPY YES attribute. The START-RBA value in the SYSCOPY record(s) is the same for all objects quiesced during this invocation of the utility. This RBA/LRSN value will be printed in the message DSNU474 following this message. Processing continues to the next table space in the table space list, if any.

Severity: 0 (informational)

DSNU478I   *csect-name* **- INDEX** *index-name* **HAS PENDING STATE**

Explanation: The specified index is in a pending state. This message is repeated for all indexes in a pending state.

csect-name
　　　The name of the control section that issued the message.

index-name
　　　The index name.

System action: The QUIESCE utility stops processing but continues to check all table spaces and indexes for any additional table space or index that has a pending state. The SYSCOPY table is not updated for any table space in the list.

User response: Issue the DISPLAY DATABASE command to look at the utility status byte in the DBA table for a specified pending state.

Severity: 8 (error)

Related tasks:

 Backing up and recovering your data (DB2 Administration Guide)

Related reference:

 Advisory or restrictive states (DB2 Utilities)

 -DISPLAY DATABASE (DB2) (DB2 Commands)

DSNU479I   *csect-name* **- UNABLE TO EXTERNALIZE INDEX** *index-name* **PAGES**

Explanation: The attempt to externalize index 'index-name' pages to the media has failed.

System action: QUIESCE still continues but the table space cannot be volume-dumped.

Problem determination: The problem may be an I/O error.

Severity: 4 (warning)

DSNU480I   *csect-name* **- UNABLE TO EXTERNALIZE TABLESPACE** *tablespace-name* **PARTITION** *partition-number*
　　　　　　PAGES

Explanation: The attempt to externalize TABLESPACE 'tablespace-name' PARTITION 'partition-number' pages to the media failed.

System action: QUIESCE still continues, but the table space cannot be volume-dumped.

Problem determination: The problem might be an I/O error.

Severity: 4 (warning)

DSNU481I   *csect-name* **- QUIESCE SUCCESSFUL FOR** *obj-type obj-qual.obj-name* **PARTITION** *partition*

Explanation: The QUIESCE for PARTITION 'partition-number' of this object was successful. If the object is a table space partition, then the quiesce for all of its indexes was successful. This message will also be issued for each index with the COPY YES attribute over the quiesced table space. This message is repeated for every quiesced table space in the list.

System action: All data held in the DB2 buffer pool(s) for the table space and its indexes have been written out to the media if message DSNU480 or DSNU482 was not issued earlier. A SYSCOPY record of ICTYPE 'Q' was recorded in the SYSCOPY catalog table for the table space and for each of its indexes with the COPY YES attribute. The START-RBA value in this SYSCOPY record is the same for all objects quiesced during this invocation of the utility. This RBA value is printed in message DSNU474 following this message. Processing continues to the next table space in the table space list, if any.

Severity: 0 (informational)

DSNU482I   *csect-name* **- UNABLE TO EXTERNALIZE INDEX** *index-name* **PARTITION** *partition-number* **PAGES**

Explanation: The attempt to externalize index 'index-name' 'partition-number' pages to the media failed.

System action: QUIESCE still continues, but the table space cannot be volume-dumped.

Problem determination: The problem might be an I/O error.

Severity: 4 (warning)

DSNU483I   *csect-name* **- TABLESPACE** *tablespace-name* **PARTITION** *partition-number* **HAS WRITE ERROR RANGE OR IS IN DEFERRED RESTART.**

Explanation: The attempt to externalize table space *tablespace-name* partition *partition-number* pages to the media failed.

System action: QUIESCE still continues, but you cannot make an image copy of the table space or partition. Processing continues for the index partition and all nonpartitioned indexes within the table space, and then proceeds to the next table space in the table space list, if any.

User response: Issue DISPLAY DATABASE to determine what exception state, if any, would prevent image copy on the table space or partition.

Severity: 4 (warning)

Related concepts:

➡ Deferral of restart (DB2 Administration Guide)

Related tasks:

➡ Backing up and recovering your data (DB2 Administration Guide)

DSNU484I   *csect-name* **- INDEX** *index-name* **PARTITION** *partition-number* **HAS WRITE ERROR RANGE OR IS IN DEFERRED RESTART**

Explanation: The attempt to externalize index *index-name* pages partition *partition-number* pages to the media failed.

System action: QUIESCE processing continues.

User response: Issue DISPLAY DATABASE to determine the exception state.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of the database recovery function.

Severity: 4 (warning)

DSNU485I   *csect-name* **QUIESCE WRITE(NO) REQUESTED ON** *obj-type obj-qual.obj-name* **WHICH HAS THE NOT LOGGED ATTRIBUTE**

Explanation: QUIESCE with the WRITE(NO) option was requested on an object with the NOT LOGGED attribute. Invoking QUIESCE WRITE(NO) on an object with the NOT LOGGED attribute does not create a recoverable point and does not write the pages from the buffer pool to DASD.

System action: The object will be skipped by QUIESCE, and the other objects in the list are processed.

User response: If you want a quiesced point for the object, complete the following steps:

• Alter the table space to have the LOGGED attribute.
• Take a full image copy.
• Rerun QUIESCE.

Severity: 0 (informational)

DSNU500I   *csect-name* **RECOVERY COMPLETE, ELAPSED TIME =** *hh/mm/ss*

Explanation: This message is issued at the normal completion of the RECOVER utility. The time that elapsed since the beginning of the RECOVER utility to its completion was 'hh' hours, 'mm' minutes, and 'ss' seconds.

System action: This message indicates the normal completion of the RECOVER utility. Processing continues to the next utility statement in the utility job step, if any. If not, the utility job step is normally terminated, and the resources held become available for further processing. The UTILID of the utility job step becomes available for reuse.

Operator response: The recovered resources are available for further processing.

User response: The recovered data is now available for further processing.

Severity: 0/4 (informational/warning)

DSNU501I   *csect-name* **A PAGE HAS BEEN STOPPED DURING** *utility-name*. **SEE PREVIOUS DSNI012 MESSAGE**

Explanation: The utility completed normally, but the data manager subcomponent cannot apply all of the log records because the DB2 is inconsistent.

csect-name
>   The name of the control section that issued the message.

utility-name
>   The name of the utility: RECOVER or FLASHCOPY WITH CONSISTENCY.

System action: The data manager subcomponent marked the pages that are identified in message the preceding DSNI012 message as broken. All other pages were recovered successfully. Log apply was stopped at the point of the inconsistency in the DB2 log for that page.

User response: Use the REPAIR utility to apply changes that should have been made by the log record that is in error.

If you were recovering to the current point in time, use the RECOVER utility, specifying `'PAGE(`*page-number*`)` `CONTINUE'`, where *page-number* is the page number of the page where the inconsistency was found. DB2 will retrieve the page from disk and apply log records starting after the inconsistent one.

Severity: 8 (error)

Related concepts:

FlashCopy image copies (DB2 Utilities)

Related reference:

RECOVER (DB2 Utilities)

REPAIR (DB2 Utilities)

Related information:
"DSNH012I" on page 157

DSNU502I   *csect-name* **PAGE** *page number* **WAS NOT FOUND IN ANY IMAGE COPY DATA SET**

Explanation: The page specified for 'RECOVER page' was not found in any image copy data sets.

This is an informational message only. It does not indicate termination of processing.

System action: The RECOVER utility continues processing. The assumption is that the page was inserted after the last image copy was made. The page is reset and recovered from the log, starting at the log RBA recorded in the SYSCOPY record for the last image copy data set.

Severity: 0 (informational)

DSNU503I   *csect-name* **PAGE** *page-number* **IS WITHIN PHYSICAL ERROR RANGE**

Explanation: The page specified for 'RECOVER page' is within the physical error range of the table space partition, index space partition, or data set. 'RECOVER page' cannot be effective in this case, because the page would still be within the physical error range and therefore inaccessible to the user.

System action: The RECOVER utility terminates with return code 8. No changes are made to the table space.

User response: Resubmit the RECOVER utility specifying 'error range'.

Severity: 8 (error)

DSNU504I   *csect-name* **MERGE STATISTICS FOR** *obj-type obj-qual.obj-name* **- NUMBER OF COPIES =** *cc*,
　　　　　NUMBER OF PAGES MERGED = *nnnnnn*, **ELAPSED TIME =** *hh/mm/ss*

Explanation: This message is issued at the completion of the MERGE process for the MERGECOPY or RECOVER utility. The message indicates the number of image copies currently processed ('cc') and the total number of pages that were merged ('nnnnnn'). The same page can be obtained from more than one incremental image copy. MERGE is repeated for MERGECOPY if all of the input image copy data sets cannot be allocated and a temporary work data set has been specified.

For the RECOVER utility, MERGE is executed whether or not there are incremental image copies participating. In this case, the number of copies ('cc') is one.

The elapsed time of the MERGE is 'hh/mm/ss', hours, minutes, and seconds, respectively.

System action: The MERGECOPY or RECOVER utility continues processing; this message is for information only.

User response: The statistics are indicative of the processing required. For MERGECOPY, it could indicate that next time more input image copy data sets should be allocated. For example, if there was more than one MERGE pass then this message will be issued more than once. For both MERGECOPY and RECOVER, it could indicate that more frequent invocation of MERGECOPY is appropriate.

DSNU505I   *csect-name* **PAGE** *page-number* **IS BEYOND END OF** *obj-type*

Explanation: A page number was found to be beyond the end of the table space, index space, or the current data set.

System action: The RECOVER utility terminates with ABEND 04E reason code 00E40100. If RECOVER PAGE was specified, then the utility terminates with return code 8 and no changes are made to the object.

User response: Check that the image copy data set(s) in use during the RECOVER contain valid pages. If running RECOVER with the PAGE option, then resubmit the utility with the correct page specified.

Severity: 8 (error)

DSNU506I   *csect-name* **TABLESPACE** *obj-qualifier.obj-name* **DSNUM** *dsnum* **IS IN REORG PENDING STATE**

Explanation: This message is issued at completion of a utility when an object has the REORP state set.

The utility can set the REORP state if the table space contains a table that includes an identity column. It can also set the REORP state when a partitioned table space, which has had its limiting keys altered, is recovered to a point in time.

System action: The utility continues processing.

Severity: 0 (informational)

DSNU507I   *csect-name* **NO USABLE BACKUP FOR RECOVER**

Explanation: The RECOVER utility determined that a backup was required but not found. The utility encountered one of the following situations:

- An error occurred while processing the oldest available full image copy data set. This might be the only full image copy data set available.
- While scanning backward through the SYSCOPY table, a LOAD LOG(NO) or REORG log(NO) record type was found.

Any of the following is acceptable as a backup:

- A full image copy.
- A LOAD LOG(YES).
- A REORG LOG(YES).
- For RECOVER PAGE, an incremental image copy that contains the specified page.
- A 'no record found' return from the first request for a SYSCOPY record indicates that the object was entirely populated by 'inserts'. The object will be recovered from the DB2 log if its entire history lies within the scope of the log, that is, archive log data sets containing updates for the object have not been deleted.

System action: The RECOVER utility terminates.

User response: Re-create the object that was to be recovered.

Severity: 8 (error)

DSNU508I   *csect-name* **IN FALL BACK PROCESSING TO PRIOR FULL IMAGE COPY**

Explanation: One of the following occurred:

- The RECOVER utility could not allocate the most recent full image copy data set. The prior full image copy is used.

  Accompanying messages indicate why the image copy data set could not be allocated.

- PAGE or ERROR RANGE was specified and the recoverable point identified an image copy that was created by the DFSMSdss DUMP command with the CONCURRENT option. Such an image copy cannot be used to process PAGE or ERROR RANGE. A prior recoverable point is located.

System action: The RECOVER utility continues processing, searching for the prior image copy.

DSNU509I   *csect-name* **- NO ERROR RANGE TO RECOVER**

Explanation: The RECOVER utility was invoked to recover the existing ERROR RANGE. However, no ERROR RANGE was found.

System action: This message indicates the completion of the RECOVER ERROR RANGE utility. Processing continues to the next utility statement in the utility job step. If there are no more utility statements, the utility job step is normally terminated.

Severity: 4 (warning)

DSNU510I   **NO RECOVERY BASE AVAILABLE FOR RECOVERY OF** *obj-type obj-qual*.*obj-name*

Explanation: It is not possible to locate a good recovery base. A recovery base could be either a full image copy or a system-level backup. If you only take image copies, the following conditions are possible:

- No back level image copy exists, or
- An I/O error occurred while reading the back level image copies, or
- DB2 was unable to locate a suitable image copy

The return code that accompanies this message is 0 (informational) or 8 (error) under the following conditions:

0 (informational)
    A full image copy is not found in the SYSCOPY table during RECOVER initialization or during fall back processing or if YES was specified for the value of the SYSTEM_LEVEL_BACKUPS subsystem parameter and a system-level backup was not chosen by RECOVER as a recovery base. However, a LOG(YES) event was detected and recovery continues from the log. Note that if you have performed a REORG after your last image copy or system-level backup, then the image copy of system-level backup will not be chosen as the recovery base.

8 (error)
    The RECOVER utility is in fallback mode and no base recoverable point (a full image copy or LOG(YES) event) was found in the SYSCOPY table. Further utility processing is terminated.

System action: If the return code is 0, processing continues. If the return code is 8, further utility processing is terminated.

System programmer response: If the return code is 8 and the object being recovered is a table space, the data must be repaired so that a backup can be taken.

User response: Notify the system programmer if the return code is 8. If the return code is 0, no action is required.

Severity: 0 (informational) or 8 (error)

Related concepts:

➡ Point-in-time recovery using the RECOVER utility (DB2 Administration Guide)

Related tasks:

➡ Backing up and recovering your data (DB2 Administration Guide)

Related reference:

➡ RECOVER (DB2 Utilities)

➡ SYSTEM-LEVEL BACKUPS field (SYSTEM_LEVEL_BACKUPS subsystem parameter) (DB2 Installation and Migration)

DSNU511I   *csect-name* **RECOVERY INFORMATION INDICATES THAT** *obj-type obj-qual.obj-name* **IS NOT IN A RECOVERABLE STATE**

Explanation: The requested recovery of an object would result in recovery entirely from the log. Because the object is also in copy pending status, recovery might fail in log apply in these situations:

- A previous MODIFY utility job deleted the last recoverable point indicated in the SYSCOPY catalog table.
- A previous MODIFY utility job deleted all SYSCOPY entries.

In either case, the MODIFY utility leaves the object in copy pending status and the RECOVER utility is unable to determine the reliability of recovery from the log.

System action: The RECOVER utility terminates with return code 8.

User response: Determine the recoverability status of the object. SYSCOPY information is probably not useful as no recoverable point (FULL IMAGE COPY, REORG LOG(YES) or LOAD REPLACE LOG(YES)) was found by the RECOVER utility. User information might be the only indication about recoverability.

To allow RECOVER to execute, copy pending status must be turned off. It is safest to make a full image copy before attempting the recovery. Then run the RECOVER utility with the TORBA or TOLOGPOINT option.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 8 (error)

DSNU512I   *csect-name* **DATA SET LEVEL RECOVERY IS REQUIRED FOR** *obj-type obj-qual.obj-name*

Explanation: Recovery was requested for an entire object. However, an image copy exists for a specific data set that is associated with the object. Hence, the object must be recovered by data set.

csect-name
        The name of the control section that issued the message.

obj-type The type of object.

obj-qual.obj-name
        The qualified name of the object.

System action: RECOVER completes unsuccessfully with a return code of 8.

User response: Terminate the current RECOVER job. Then, submit a RECOVER job with the DSNUM(*data set*

number) option for each data set for which recovery is required.

Severity: 8 (error)

Related tasks:

Recovering a data set or partition (DB2 Utilities)

Related reference:

Syntax and options of the RECOVER control statement (DB2 Utilities)

DSNU513I *csect-name utility-name* **UTILITY LOG APPLY RANGE IS RBA** *rba-start* **LRSN** *lrsn-start* **TO RBA** *rba-end* **LRSN** *lrsn-end*

Explanation: This message is issued one or more times to display the entire log range before applying updates.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility: RECOVER or FLASHCOPY WITH CONSISTENCY.

rba-start **and** *rba-end*
> The beginning and end of the RBA range.
>
> These values are zero in a data sharing environment for the catalog and directory objects that do not have DSNDB01.SYSLGRNX entries. (For a list of catalog and directory objects that do not have DSNDB01.SYSLGRNX entries, see the information about point-in-time recovery of the catalog and directory.)

lrsn-start **and** *lrsn-end*
> The beginning and end of the LRSN range.

The RBA and LRSN range values are equal in a non-data sharing environment.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related concepts:

FlashCopy image copies (DB2 Utilities)

Point-in-time recovery of the catalog, directory, and all user objects (DB2 Utilities)

How RBA and LRSN values are displayed (DB2 Administration Guide)

Related reference:

RECOVER (DB2 Utilities)

DSNU514I *csect-name* **RECOVERY DATA DOES NOT PERMIT DSNUM(ALL) RECOVERY OF** *obj-type obj-qual.obj-name*

Explanation: Recovery was requested for an entire object. If the object is a table space, then the latest activity on the entire table space was a REORG LOG (NO) or a LOAD LOG (NO). If the object is an index space, then the latest activity on the entire index space was a REORG INDEX or REBUILD INDEX. If full image copies have been taken at the data set level since then, recover the object on the data set level.

System action: RECOVER completes unsuccessfully with a return code of 8.

User response: Submit a recover job by specifying DSNUM (data set number) for each data set requiring recovery. If the object is an index space, then the REBUILD INDEX utility may be used to rebuild the index.

Severity: 8 (error)

DSNU515I • DSNU516I

DSNU515I   *csect-name* **THE IMAGE COPY DATA SET** *dsn* **WITH DATE=***date* **AND TIME=***time* **IS PARTICIPATING IN RECOVERY OF** *obj-type obj-qual.obj-name*

Explanation: This message is issued by the RECOVER utility for each participating image copy data set before it opens the data set and begins the restore process. If an error occurs while processing, the utility searches for a new recoverable base.

This message is routed to SYSOUT.

System action: RECOVER processing continues.

User response: No action is required.

Severity: 0 (informational)

DSNU516I   **NO BACKUP WAS TAKEN AFTER LOAD, REORG OR CHECK DATA LOG(NO) OF** *obj-type obj-qual.obj-name*

Explanation: Recovery was requested for a data set of an object. If the object is a table space, then a system-level backup or full image copy was not taken after or during a LOAD, REORG, or CHECK DATA without logging. The table space is in an unrecoverable state. If the object is an index space, then a full image copy or system-level backup was not taken after a REORG INDEX or REBUILD INDEX.

System action: The RECOVER utility completes unsuccessfully with a return code of 8.

System programmer response: The data should be repaired before a full image copy or system-level backup is taken. The recovery information for the object (use the REPORT RECOVERY utility) should be scrutinized.

User response: If the object is a table space, the data set is in an unrecoverable state. If the object is an index space, then the REBUILD INDEX utility may be used to rebuild the index. Notify the system programmer for corrective action.

Severity: 8 (error)

DSNU517I   *csect-name* **- SYSCOPY RECORD DELETED BY MODIFY UTILITY. DSN=***dsn***, VOL=SER=***nnn***,**
FILESEQNO=*fff*

Explanation: This message is repeated for each volume of a multivolume data set that is deleted.

The variables have the following meanings:

dsn   The name of the image copy data set whose record was removed from the SYSCOPY table by the MODIFY
            utility.

nnn   The serial number of the volume containing the data set.

fff   The tape file sequence number. It will be zero for DASD data sets.

System action: The MODIFY utility continues its termination processing.

System programmer response: Because DB2 no longer has any knowledge of the image copy data set, you may
delete it and reuse the space. If the last full image copy was deleted, ensure that another one is taken as soon as
possible.

Severity: 0 (informational)

DSNU518I   *csect-name object-type object-qualifier.object-name* **CONTAINS BROKEN PAGE** *page-number*, **ERROR**
CODE *error*

Explanation: A broken page, *page number*, was detected during execution of an image copy utility job. The copy step
is terminated and the broken page is not copied to the output data set.

System action: COPY utility ends with a return code of 8.

System programmer response: Use either the RECOVER, RECOVER PAGE, or REPAIR utility to correct the broken
page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and
reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: The associated data set has not been copied. If the object is in the COPY-pending status, it is in an
unrecoverable state. Notify the system programmer for corrective action.

Severity: 8 (error)

Related concepts:

➡ Recovery of page sets and data sets (DB2 Administration Guide)

➡ Recovering a page (DB2 Utilities)

Related tasks:

➡ Repairing a damaged page (DB2 Utilities)

Related reference:

➡ RECOVER (DB2 Utilities)

➡ REPAIR (DB2 Utilities)

DSNU519I   *csect-name* **TOCOPY DATA SET NOT FOUND**

Explanation: The data set specified with the keywords TOCOPY, TOVOLUME, or TOSEQNO cannot be found in the
SYSCOPY table for the object to be recovered.

This may be caused by a mismatch in granularities. If the RECOVER specifies a particular DSNUM, the data set
specified with the TOCOPY keyword may be for an image copy of this DSNUM or of the entire table space or index
space. However, if the RECOVER is for the entire table space or index space, the specified data set must be for an
image copy of the entire table space or index space.

This message may also be issued if TOVOLUME is specified for a cataloged data set.

RECOVER TOCOPY can issue DSNU519I even if the data set information was correct but the data set was not for
this site. For example, if the DB2 subsystem is specified as LOCALSITE, but the TOCOPY data set information is for
a RECOVERYSITE data set, this message will be issued.

System action: RECOVER processing is terminated.

User response: Correct the input and resubmit the RECOVER job.

Severity: 8 (error)

DSNU520I   *csect-name* **RECOVER TO A PRIOR POINT IN TIME MIGHT LEAVE TABLE SPACE** *table-space-name* **INCONSISTENT**

Explanation: RECOVER DSNUM to a previous point in time (for example, using the TOLOGPOINT option) was specified for a nonpartitioned table space. You might have recovered a data set that has different compression characteristics than the replaced data set. For example, the data set to which you recovered might have been compressed with a different dictionary.

System action: Processing continues.

User response: The table space might be in an inconsistent state. If the dictionary was not built with the data set that was just recovered, the results are unpredictable and might jeopardize all the data in the entire table space.

Recover all of the data sets in the table space to the same point in time. Do not ignore this message unless all data sets in this table space were recovered to the same point in time. Otherwise, the state of this nonpartitioned table space is inconsistent and unpredictable.

Severity: 4 (warning)

DSNU521I   *csect-name* **- RECOVER UTILITY FAILED. UNABLE TO READ REQUIRED LOG RECORDS.**

Explanation: Recovery was requested for a table space, index space, or data set. An attempt was made to apply changes from the DB2 log, but the necessary log records could not be read from the log data set. This message is accompanied by message DSNI007I.

System action: The RECOVER utility completes unsuccessfully with return code 8.

Operator response: Provide the console listing of the DSNI007I message for the system programmer.

System programmer response: Run the print log map (DSNJU004) utility. Find the data set that contains the requested RBA or LRSN from the output of the utility.

User response: Consult the system programmer for corrective action.

Problem determination: If no listed data set contains the required RBA or LRSN, one of the following might have occurred:

- The data set was deleted from the bootstrap data set (BSDS) by the change log inventory (DSNJU003) utility. Rerun the utility to insert the missing data set name. DB2 must be stopped to run this utility.
- The BSDS was damaged. The BSDS can be restored using the dump and the most recent archive log data set. Run the change log inventory (DSNJU003) tility to supply changes that occurred during and after the last offload operation. Changes can be determined from the operator's console listing.

If a listed log data set contains the required RBA or LRSN, this message is preceded by one or more DSNJxxx error messages. See the explanation of those messages for the appropriate action.

Severity: 8 (error)

DSNU522I   *csect-name* **RECOVER CANNOT PROCEED FOR** *obj-type obj-qual.obj-name* **BECAUSE A NON-RECOVERABLE EVENT HAS BEEN ENCOUNTERED FROM SYSIBM.SYSCOPY WHICH HAS DBNAME=**dbname **TSNAME=**tsname **DSNUM=**dsnum **ICTYPE=**ictype **START_RBA=X'**rba**'**

Explanation: Recovery was requested for an index, but activity on its table space (identified by the SYSIBM.SYSCOPY columns) prohibits recovery. The use of an index's full image copy by recovery is prohibited by a REORG or LOAD on its underlying table space.

System action: Utility processing is terminated. All objects in the RECOVER statement are left in the same state as before the RECOVER utility was invoked.

User response: Use the REBUILD INDEX utility to rebuild the index from the table space. For more information about the recoverability of this index, run the REPORT RECOVERY utility on both the index and its table space.

Severity: 8 (error)

Related concepts:

➦ How RBA and LRSN values are displayed (DB2 Administration Guide)

**DSNU523I   RECOVER UTILITY FAILED. NO RECOVERY BASE WAS AVAILABLE AND THE FIRST LOG
RECORD WAS NOT THE REQUIRED INITIALIZATION RECORD**

Explanation: RECOVER was requested for a table space or a data set of a table space and no image copy or
system-level backup was available. Recovery was attempted from the log and failed due to missing data set
initialization records. Specifically, the log records which format space map and header pages were missing.

System action: RECOVER completes unsuccessfully with a return code of 8.

System programmer response: Use the problem determination information given below to determine if a user error
has occurred. If possible, correct the error and resubmit the RECOVER utility job for this table space or data set. If
you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting
the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of
possible recovery procedures.

User response: The associated data set has been reset to an empty state. Notify system programmer for corrective
action.

Problem determination: This message was preceded by message DSNU513I which identifies the log RBA range
being applied. Run the print log map (DSNJU004) utility for the table space in question over the indicated log RBA
range.

This will show what log records are present and may assist in determining if a log data set has been damaged or
deleted, or if the BSDS has been damaged. If damage to table space DSNDB01.SYSLGRNX is suspected, recover it
and attempt the earlier failing recovery again.

Severity: 8 (error)

**DSNU524I   _csect-name_ INVALID DATA SET OR PARTITION NUMBER _partition-number_ SPECIFIED FOR
object-type object-qualifier._object-name_**

Explanation: An invalid number was specified for the data set or partition of the object named in the utility
statement.

System action: Further utility processing terminates.

User response: Correct the data set/partition number, or verify that the object name is specified correctly. Then
resubmit the utility statement in a new utility job step.

Severity: 8 (error)

Related concepts:

➦ DB2 online utilities (DB2 Utilities)

➦ DB2 stand-alone utilities (DB2 Utilities)

**DSNU525I   _csect-name_ RECOVERY DATA DOES NOT PERMIT DSNUM(ALL) RECOVERY OF _obj-type_
obj-qual._obj-name_**

Explanation: Recovery was requested for an entire object. However, the object is not recoverable at that level for one
of the following reasons:

- The object was recovered to a point in time (for example, TOLOGPOINT was specified), which required fall back
  processing. However, no subsequent image copy was taken to establish a base for recovery.
- A partition of the object was recovered to a point in time after the last full image copy of the object was taken.

System action: The RECOVER utility completes unsuccessfully with return code 8.

User response: Submit the RECOVER job with DSNUM specified for each data set for which recovery is required.

Severity: 8 (error)

Related reference:

➦ Syntax and options of the RECOVER control statement (DB2 Utilities)

DSNU526I   NO BACKUP WAS TAKEN AFTER PARTIAL RECOVERY OF *obj-type objqual. obj-name*

Explanation: Recovery was requested for a data set of an object. However, an earlier point-in-time recovery (for example, using the TOLOGPOINT option) was specified and no subsequent image copy or system-level backup was taken to establish a base for recovery.

System action: The RECOVER utility completes unsuccessfully with return code 8.

System programmer response: You can use the RECOVER utility with the TOCOPY, TORBA, or TOLOGPOINT option to force recovery to an earlier point in time. You can use the DSN1COPY utility to restore the data set to an earlier image copy. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of database recovery procedures.

User response: The associated data set is in a nonrecoverable state. Consult the system programmer for corrective action.

Severity: 8 (error)

DSNU527I   *csect-name* DB2 OBJECT RECOVERED TO AN INDETERMINATE POINT

Explanation: The image copy data set specified with the TOCOPY keyword of the RECOVER utility was one of the following:

- An image copy data set made with the SHRLEVEL CHANGE option. In this mode, changes might have been made to the object while image copy was in progress. If a change was made to a page before image copy processed the page, the update will be in the copy data set. If a change was made to a page after image copy processed it, the change will not appear in the copy data set. The data set may contain partial updates from some transactions.
- An image copy data set made during the LOAD REPLACE utility. An image copy produced during LOAD REPLACE will not contain updates made to the table space to correct unique index violations during the INDEXVAL phase or referential constraint violations during the ENFORCE phase. Therefore, it is not recommended as a recovery point.

If the object recovered is a table space, then indexes on the table space are placed in rebuild pending status and dependent table spaces, if any, are placed in check pending status. The indexes placed in rebuild pending are identified by message DSNU831I, and the dependent table spaces placed in check pending are identified by message DSNU563I.

System action: The utility terminates normally.

User response: Based on the explanation above, determine if the recovery is satisfactory. Rebuild the indexes identified by message DSNU557I in the job output. Run CHECK DATA on the table spaces identified by message DSNU563I to correct any referential constraint violations.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of database recovery procedures.

Severity: 4 (warning)

DSNU528I   NO FULL IMAGE COPY WAS AVAILABLE AND THERE ARE NO UPDATES TO APPLY FROM
            THE DB2 LOG FOR TABLESPACE *tablespace-name* DSNUM *dsnum*

Explanation: Recovery was requested for a table space or a data set of a table space and no image copy or system-level backup was available. Recovery was attempted from the log, but no log records were applied because no SYSLGRNX records that correspond to the recovery scope were found.

tablespace-name
        The name of the table space for which recovery was requested.

dsnum   Identifies the partition or data set that is to be recovered. If *dsnum* is 0, recovery was requested for the entire table space.

System action: The associated data set was reset to an empty state, and the data set was placed in RECOVER-pending status.

If RECOVER was processing a list of table spaces, only the associated data set was reset to an empty state. However, all data sets in the list were placed in RECOVER-pending status.

System programmer response: Use the information in the problem determination section to determine whether a user error occurred. Correct the error and resubmit the RECOVER utility for the table space.

User response: If the TORBA or TOLOGPOINT option was specified, use the information in the problem determination section to verify that the value is a valid RBA or LRSN. Correct the error and resubmit the RECOVER utility for the table space.

Problem determination: Run the REPORT RECOVERY utility for the table space at the appropriate table space or data set level (DSNUM). The output shows the recovery points (SYSCOPY records) and the update log ranges (SYSLGRNX records) that exist for the table space.

If you take system-level backups with the BACKUP SYSTEM utility, run the Print Log Map (DSNJU004) utility to print the system-level backup information. If you use data sharing, ensure that the system-level backup information from all members is gathered. Locate the system-level backup information in the DSNJU004 output and compare it with the output from the REPORT RECOVERY utility output.

Severity: 8 (error)

Related reference:

➥ RECOVER (DB2 Utilities)

➥ RECOVER-pending status (DB2 Utilities)

➥ SYSIBM.SYSLGRNX table (DB2 SQL)

DSNU529I   *csect-name* **INVALID SPECIFICATION OF TORBA =** *xxx*

Explanation: 'xxx' is the log RBA specified for the RECOVER utility TORBA keyword. The RBA is not within the scope of the log.

System action: The RECOVER utility terminates with a return code of 8 without making any changes to the object to be recovered.

User response: Correct the input and resubmit the job. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference. for a discussion of database recovery procedures.

Problem determination: The listing produced by the print log map (DSNJU004) utility will show the current extent of the DB2 log.

Severity: 8 (error)

Related concepts:

➥ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU530I   *csect-name* **- MULTIPLE DATA SETS FOUND FOR THE COPY DATA SET SPECIFICATION**

Explanation: DB2 issues this message during the RECOVER or COPYTOCOPY utility when the specification of the input image copy data set is ambiguous and does not uniquely identify an image copy. Multiple image copies were found in SYSIBM.SYSCOPY for the object with the same data set name.

System action: Processing halts for this object.

RECOVER completes unsuccessfully with a return code of 8.

For the COPYTOCOPY utility, if the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Two or more DSNU531I messages will follow this message providing more information about each

duplicate data set. Select the correct data set from the DSNU531I messages.

Rerun the RECOVER job, specifying a unique combination of TOVOLUME and TOSEQNO.

Rerun the COPYTOCOPY job, specifying an unique combination of FROMVOLUME and FROMSEQ.

Severity: 8 (error)

DSNU531I *csect-name - dsn* **EXISTS WITH - VOLUME=***vol*, **FILE SEQUENCE NUMBER=***fseq*, **RBA=***rba*, **DATE=***date*, **TIME=***time*

Explanation: DB2 issues this message two or more times to document the duplicate data sets detected by the RECOVER utility during TOCOPY processing, or by the COPYTOCOPY utility during FROMCOPY processing. See the explanation for message DSNU530I.

System action: RECOVER completes unsuccessfully with a return code of 8.

For COPYTOCOPY, if the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the next item in the list will be processed.

User response: Use the information provided in all of the DSNU531I messages to determine the correct data set for this recovery. 'rba', 'date' and 'time' are provided for each data set to assist in the selection. The DSNU531I message for the data set you select will provide the correct 'vol' and 'fseq' values to be specified on the TOVOLUME and TOSEQNO keywords respectively.

Rerun the RECOVER job specifying TOCOPY and TOVOLUME(vol-ser). Include the TOSEQNO(fseq) keyword if it is necessary to distinguish an unique data set.

Rerun the COPYTOCOPY job specifying FROMCOPY and FROMVOLUME(vol-ser). Include the FROMSEQNO(fseq) keyword if it is necessary to distinguish a unique data set.

Severity: 8 (error)

Related concepts:

How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU532I *csect-name* **- RECOVER** *obj-type obj-qual.obj-name* **START**

Explanation: This message is issued when there is more than one object in the list. It indicates that the recover of a particular object is starting. If the PARALLEL keyword has been specified, then the restore phase messages for different objects in the list may be intermingled.

This message is routed to SYSPRINT.

System action: RECOVER processing continues.

Severity: 0 (informational)

DSNU533I *csect-name* **-** *obj-type obj-qual.obj-name* **IS A DUPLICATE OF** *obj-type obj-qual.obj-name*

Explanation: The utility found a duplicate object in the list. The object name may indicate a partition or piece number.

System action: The utility job terminates processing.

User response: Correct the object list. Resubmit the job if needed.

Severity: 8 (error)

DSNU534I *csect-name* **TARGET RECOVERY POINT IS NOT A QUIESCE POINT**

Explanation: The RECOVER utility attempted to recover to a point in time, but one of the following conditions was detected:

- The table space set is not being recovered to a quiesce point or to a set of image copies that were taken with SHRLEVEL REFERENCE.
- Not all table spaces in the table space set have the same quiesce point.

| • A table space in the table space set contains a dependent table with a referential constraint that was defined after the specified quiesce point.

| A *table space set* is a group of related table spaces. It includes a table space and the following related table spaces:
| • Any referentially related table spaces
| • Any associated auxiliary LOB or XML table spaces
| • If the table space contains an archive-enabled table, the table space that contains the associated archive table

| *csect-name*
| The name of the control section that issued the message.

| **User response:** If possible, recover the table space set to a quiesce point or to a set of image copies that were taken with SHRLEVEL REFERENCE. Otherwise, use the CHECK DATA utility to check and correct referential violations. Use the CHECK DATA utility on the base table space and the CHECK LOB utility on any LOB table spaces to validate the LOB columns.

| **Severity:** 4 (warning)
| **Related concepts**:

| ➡ Common quiesce points (DB2 Utilities)

| ➡ Point-in-time recovery (DB2 Utilities)

| ➡ Creation of relationships with referential constraints (Introduction to DB2 for z/OS)

| ➡ Large object table spaces (Introduction to DB2 for z/OS)

| ➡ XML table spaces (Introduction to DB2 for z/OS)

| ➡ Archive-enabled tables and archive tables (Introduction to DB2 for z/OS)

| **Related reference**:

| ➡ CHECK DATA (DB2 Utilities)

| ➡ CHECK LOB (DB2 Utilities)

DSNU535I   *csect-name* **- FOLLOWING TABLESPACES RECOVERED TO A CONSISTENT POINT**

Explanation: The RECOVER utility recovered a table space set to a quiesce point or to a set of image copies that were taken with SHRLEVEL REFERENCE.

A *table space set* is a group of related table spaces. It includes a table space and the following related table spaces:
• Any referentially related table spaces
• Any associated auxiliary LOB or XML table spaces
• If the table space contains an archive-enabled table, the table space that contains the associated archive table

All table spaces that were recovered as part of the table space set are listed after the message heading.

csect-name
        The name of the control section that issued the message.

System action: The listed table spaces are recovered.

User response: No action is required.

Severity: 0

Related concepts:

➔ Common quiesce points (DB2 Utilities)

➔ Creation of relationships with referential constraints (Introduction to DB2 for z/OS)

➔ Large object table spaces (Introduction to DB2 for z/OS)

➔ XML table spaces (Introduction to DB2 for z/OS)

➔ Archive-enabled tables and archive tables (Introduction to DB2 for z/OS)

Related reference:

➔ RECOVER (DB2 Utilities)

DSNU536I   *csect-name* **- RELATIONSHIP** *relationship-identifier* **ON TABLE** *table-name* **CREATED AFTER QUIESCE POINT**

Explanation: The relationship named *relationship-identifier* was created on the dependent table, *table-name*, after the quiesce point used for recovering to a point in time. The table space in which *table-name* resides is set to check pending beginning with the first record of the *table-name* table.

System action: No action is taken.

User response: Analyze and resolve the check pending state.

Severity: 4 (warning)

Related concepts:

➔ Implications of adding parent or foreign keys (DB2 Administration Guide)

➔ Resetting the CHECK-pending status (DB2 Utilities)

Related tasks:

➔ Resetting CHECK-pending status (DB2 Utilities)

➔ Collecting data for failures after a point-in-time recovery (Collecting data)

➔ Adding referential constraints to existing tables (DB2 Administration Guide)

Related reference:

➔ CHECK-pending status (DB2 Utilities)

Related information:

➔ Recovering from referential constraint violation (DB2 Administration Guide)

DSNU537I   *csect-name object-type object-qualifier.object-name* **IS NOT ALLOWED IN THE LIST**

Explanation: The RECOVER utility or COPY utility found a directory or catalog table space in a list with other table spaces or with user-defined indexes.

csect-name
        The name of the control section that issued the message.

object-type
        The type of object.

object-qualifier.object-name
        The qualified object name.

For the COPY utility, you cannot specify the following table spaces in a list with other table spaces:
- DSNDB01.SYSUTILX
- DSNDB06.SYSCOPY
- DSNDB01.SYSLGRNX
- DSNDB01.SYSDBDXA

However, you can specify these table spaces in a list with their indexes that are defined by IBM.

For the RECOVER utility, you cannot recover more than one directory or catalog table space in a list. Directory and catalog table spaces must be recovered individually in a particular order, because the recovery of some objects depends on information in other objects. Indexes that are defined by IBM can be specified with their associated directory or catalog table space in a list.

System action: The utility job terminates.

User response: Remove the directory and catalog object names from the list. Then, submit the COPY or RECOVER job again.

Severity: 8 (error)

Related concepts:

➡ Copies of lists of objects (DB2 Utilities)

Related tasks:

➡ Recovering catalog and directory objects (DB2 Utilities)

DSNU538I **RECOVER ERROR RANGE OF DSN=***dataset-name* **ON VOLUME=***volume-serial* **FROM CCHH=X'***low-cchh***' TO CCHH=X'***high-cchh***' CONTAINS PHYSICAL ERROR**

Explanation: This message indicates that the error range involved in the error range recover *dataset-name* is the data set that has the physical error. The data set resides on DASD volume *volume-serial*, and the error range is from track address X'*low-cchh*' to track address X'*high-cchh*'.

If the error range of a data set covers more than one extent or more than one volume, one message is issued per extent per volume.

dataset-name
> The data set name.

volume-serial
> The volume serial number.

low-cchh
> The track address of the beginning of the error range, in CCCCcccH format.

high-cchh
> The track address of the end of the error range, in CCCCcccH format.

System action: Physical I/O error detected during error range recover. Processing terminated with return code 8.

System programmer response: A DSNU086 message is issued to the console by DB2 while running error range recovery. This message indicates that the physical error is not recovered. Run ICKDSF or equivalent service utility to recover the physical error then run the error range recover again.

Problem determination: Use the information that is provided by this message to run ICKDSF and then run error range recover again.

Severity: 8 (error)

Related reference:

➡ DB2 catalog tables (DB2 SQL)

Related information:
"DSNU086I" on page 617

DSNU539I *csect-name* **- RECOVER PROCEEDS AT THE** *site-type* **SYSTEM**

Explanation: The RECOVER process continues at the RECOVERY SITE system, and/or the TRACKER SITE system. DB2 issues this message for informational purposes only.

System action: The utility continues.

Severity: 0 (informational)

DSNU540I *csect-name* **DYNAMIC ALLOCATION FOR DATA SET** *data-set-name* **WILL WAIT FOR UNIT TO BECOME AVAILABLE**

Explanation: A request to dynamically allocate a data set for a BATCH UTILITY job was refused because a unit was not available. The request was repeated after setting an indicator in the dynamic allocation parameter list to cause the job to wait until a unit becomes available.

System action: Processing continues after a unit becomes available and is allocated.

Severity: 0 (informational)

DSNU541I **UNABLE TO ALLOCATE THE PRIMARY COPY DATASET** *data-set-name*, **BACKUP COPY IS USED**

Explanation: If RECOVER or MERGECOPY has an allocation error on the primary copy data set, it will locate the backup copy data set for the same system (local or recovery). The utility continues.

Severity: 0 (informational)

DSNU542I *csect-name* **- UNABLE TO OPEN THE PRIMARY COPY DATASET** *dataset-name*, **BACKUP COPY IS USED**

Explanation: If RECOVER or MERGECOPY has an open error in the primary copy data set, it will deallocate that copy data set and try to allocate the backup copy data set for that same system (local or recovery). The utility continues.

Severity: 0 (informational)

DSNU543I *csect-name* **- I/O ERROR READING PAGE** *page_number* **IN THE PRIMARY COPY DATASET** *dataset-name*, **BACKUP COPY IS USED**

Explanation: If RECOVER or MERGECOPY has an I/O error when reading a page, with the specified page number, from the primary copied data set, it will locate the backup copied data set in that same system (local or recovery) and continue reading pages starting from the page number in error.

Severity: 0 (informational)

DSNU544I *csect-name* **- UNABLE TO OPEN THE COPY DATASET** *dataset-name* **FOR THE RECOVERY OF** *obj-type obj-qualifier.obj-name*, **RC=** *reason-code*

Explanation: RECOVER was unable to open the data set 'dataset-name'. because of 'reason-code' from the open abend. The 'reason-code' is taken from field SDWAABCC in the SDWA.

System action: The RECOVER utility continues processing, searching for the prior image copy or applying log records.

Severity: 0 (informational)

DSNU545I *csect-name* **- I/O ERROR READING PAGE** *page-number* **IN THE COPY DATASET** *dataset-name* **DURING RECOVERY OF** *obj-type obj-qual.obj-name*

Explanation: An I/O error was detected while reading page *page-number* from copy data set *dataset-name*. The message is issued for the following reasons:

- The data set is the primary copy data set, and there is no backup data set.
- The open error is from the backup copy data set.

System action: The RECOVER utility continues processing, searching for the prior image copy, or applying log records.

Severity: 0 (informational)

DSNU546I   LOGONLY RECOVER CANNOT PROCEED BETWEEN BEGINNING RBA '*xxxxxxxxxxxx*' AND
ENDING RBA '*yyyyyyyyyyyy*' ON DSNUM *dsnum* BECAUSE A NON-RECOVERABLE EVENT HAS
BEEN ENCOUNTERED FROM SYSIBM.SYSCOPY WHICH HAS DBNAME=*database-name*
TSNAME=*space-name* DSNUM=*dsnum* ICTYPE=*x* START_RBA=X'*zzzzzzzzzzzz*'.

Explanation: A nonrecoverable event (for example, LOAD LOG(NO), REORG LOG(NO)) occurred within the log
range between the beginning log RBA or LRSN and ending log RBA or LRSN for the RECOVER LOGONLY utility.

System action: The RECOVER utility completes unsuccessfully with return code 8.

System programmer response: Find a proper prior point in time as a target recovery point and rerun the RECOVER
utility job with the TORBA, TOCOPY, or TOLOGPOINT option.

Severity: 8 (error)

Related concepts:

▶ Point-in-time recovery using the RECOVER utility (DB2 Administration Guide)

▶ How RBA and LRSN values are displayed (DB2 Administration Guide)

Related tasks:

▶ Backing up and recovering your data (DB2 Administration Guide)

Related reference:

▶ RECOVER (DB2 Utilities)

DSNU547I   LOGONLY *action* CANNOT PROCEED BECAUSE THE STARTING RBA=X'*xxxxxxxxxxxx*' IN THE
HEADER PAGE PGNO=X'*xxxxxx*' FOR *obj-type obj-qual.obj-name* IS INVALID.

Explanation: One of the following results occurred:
- The data set was restored from an earlier release backup that does not have a starting RBA or LRSN in the header
  page.
- The TORBA or TOLOGPOINT option is specified to an RBA or LRSN that is lower than the starting RBA or LRSN
  in the header page.

System action: The RECOVER or RESTORE utility job terminates with return code 8.

System programmer response: Use the RECOVER or RESTORE utility without the LOGONLY option, if possible, or
change the RBA or LRSN of the TORBA or TOLOGPOINT option to a valid RBA or LRSN value.

Severity: 8 (error)

Related concepts:

▶ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU548I   DBID=X'*xxxx*' and PSID=X'*xxxx*' IN THE HEADER PAGE OF *obj-type obj-qual.obj-name* DO NOT
MATCH THE DB2 CATALOG.

Explanation: The DBID and PSID in the header page do not match the DB2 catalog. It is likely that the wrong data
sets were restored.

System action: The RECOVER utility job terminates with return code 8.

System programmer response: Find the correct data sets and rerun the job.

Severity: 8 (error)

DSNU549I   RECOVER *obj-type obj-qual.obj-name dsnum* USES ONLY DB2 LOGS STARTING FROM
LOGPOINT=X'*xxxxxxxxxxxx*'.

Explanation: The RECOVER utility issues one message for each DB2 object that uses only DB2 logs to recover. In a
non data-sharing environment, the LOGPOINT value is specified by an RBA; in a data sharing environment it is
specified by an LRSN.

System action: RECOVER utility processing continues.

DSNU550I • DSNU554I

Severity: 0 (informational)

DSNU550I *csect-name* **INDEX NAME=** *index-name* **IS EMPTY**

Explanation: While attempting to REBUILD the indicated index, it was determined that the table upon which the index is based is empty.

System action: Utility processing proceeds to the next utility request, if any.

Severity: 4 (warning)

DSNU551I *csect-name* **INDEX NAME=** *index-name* **PART=***part-num* **IS EMPTY**

Explanation: While attempting to REBUILD the indicated index partition, it was determined that the partition upon which the specified index partition is based is empty.

System action: Utility processing proceeds to the next utility request, if any.

Severity: 4 (warning)

DSNU552I *csect-name* **RECOVER CANNOT PROCEED BECAUSE THE** *starting/ending* **LOGPOINT** *xxx* **IS NOT WITHIN THE RECOVERY RANGE OF** *xxx* **AND** *xxx* **FOR** *obj-type obj-qual.obj-name*

Explanation: The starting or ending LOGPOINT found for this recovery job is outside the available recovery information. A LOGPOINT value is an RBA or LRSN. The RECOVER utility cannot process this request.

System action: The RECOVER utility terminates.

User response: Use the recovery information that is available from the REPORT utility. Execute the RECOVER utility within the available recovery range.

Severity: 8

DSNU553I *csect-name* **INVALID PARTITION NUMBER=***partition-number* **SPECIFIED FOR INDEX NAME=***index-name*

Explanation: An invalid partition number *partition-number* was specified for index *index-name* in the utility statement.

System action: Further utility processing is terminated.

User response: Correct the partition number or verify that the index name is specified correctly, and resubmit the utility statement in a new utility job step.

Severity: 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

➡ DB2 stand-alone utilities (DB2 Utilities)

DSNU554I *csect-name* **- DUPLICATE INDEX NAMES EXIST IN THE SPECIFIED INDEX NAME LIST**

Explanation: When attempting to CHECK or REBUILD the indicated indexes, it was determined that duplicate index names were specified in the index name list.

System action: The utility job terminated.

User response: Correct the index-name(s) in the index name list and resubmit the job.

Severity: 8 (error)

DSNU555I *csect-name* **UNLOAD PHASE STATISTICS, NUMBER OF RECORDS=***xxxx*

Explanation: This message is issued at the normal completion of the UNLOAD phase of the REBUILD INDEX utility. It gives the number of records that were unloaded (xxxx) for the specified table space.

Severity: 0 (informational)

DSNU556I *csect-name* **RECOVER CANNOT PROCEED FOR** *object-type object-qualifier.object-name* **BECAUSE A SYSCOPY RECORD HAS BEEN ENCOUNTERED WHICH HAS DBNAME=** *dbname* **TSNAME=** *tsname* **DSNUM=** *dsnum* **ICTYPE=** *ictype* **STYPE=** *stype* **STARTRBA=X'***rba***' LOWDSNUM=** *low-dsnum* **HIGHDSNUM=***high-dsnum***.**

Explanation: DB2 encountered an event that is recorded in a SYSCOPY record that prohibits recovery processing.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

database-name
> Name of the database.

tablespace-name
> Name of the table space.

ictype **and** *stype*
> The type of event is recorded in the *ictype* and *stype* columns.

> **ICTYPE=R,S,W, or X (REORG, LOAD REPLACE, CATENFM)**

>> **STYPE=A**
>>> A SYSCOPY record for a REORG or LOAD REPLACE operation that redistributed data across partitions was found. The partitions that were affected by the ALTER INDEX that changed the limit keys must be recovered in the same list.

>> **STYPE=S**
>>> The CATENFM utility was run. A point-in-time recovery before the CATENFM process is not allowed unless the entire system (including the DB2 catalog and directory) are recovered to the same point.

>> **STYPE= "blank"**
>>> A SYSCOPY record was found for a REORG operation with the LOG(YES) option in the recovery period. Recovery to a prior point in time is not possible.

> **ICTYPE=A (ALTER)**

>> **STYPE=C**
>>> An ALTER took place that performed one of the following actions:
>>> - Added a column to a table and the column was not added to the index in the same commit scope. Rebuild the index instead of recovering it.
>>> - Dropped a column from a table.

>>> A TTYPE value of blank indicates that the column was added and a TTYPE value of 'D' indicates that the column was dropped.

>> **STYPE=D**
>>> One of the following situations occurred:
>>> - If the TTYPE value is blank, ALTER TABLE ALTER COLUMN SET DEFAULT was executed, and point-in-time recovery before the ALTER is not allowed.
>>> - If the TTYPE value is non-blank, the TTYPE value is the data set size. The DSSIZE attribute of the table space was updated by an ALTER statement. Recovery to a point in

time before the ALTER operation is not allowed unless REORG was run to materialize the ALTER operation in Version 11 new-function mode or later.

STYPE=E

An ALTER TABLE EXCHANGE occurred and a point-in-time recovery before the exchange is not allowed. If the object that is being recovered is an index, the index cannot be recovered before the rotation of its underlying table.

STYPE=F

The buffer pool attribute of the object was updated by an ALTER statement and a point-in-time recovery before the update is not allowed.

STYPE=G

An ALTER INDEX REGENERATE was done. Rebuild the index instead of recovering it.

STYPE=I

A SYSCOPY record for a REORG with TTYPE=I or TTYPE=D indicates that a REORG to materialize an ALTER COLUMN INLINE LENGTH was found. However, not all related objects (base table space and LOB table space) were recovered as a set to the same point in time. The point-in-time recovery is not allowed unless both the related base and LOB objects are recovered as a set.

STYPE=M

The MAXPARTITIONS attribute of the table space was updated by an ALTER statement and a point-in-time recovery before the update is not allowed.

STYPE=N

The index was updated by an ALTER statement to have the NOT PADDED attribute. Rebuild the index instead of recovering it.

STYPE=P

The index was updated by an ALTER statement to have the PADDED attribute. Rebuild the index instead of recovering it.

STYPE=R

A table space partition was rotated and a point-in-time recovery before the rotation is not allowed. If the object that is being recovered is one of the following objects, it cannot be recovered to a point in time before the rotation:

- An index partition
- A LOB table space that belongs to the rotated partition

STYPE=S

The SEGSIZE attribute of the table space was updated by an ALTER statement and a point-in-time recovery before the update is not allowed.

STYPE=V

An ALTER TABLE COLUMN occurred on an indexed column for a numeric data type change. Rebuild the index instead of recovering it

STYPE=X

A REORG dropped one or more empty partitions from the related table space. A partition cannot be recovered to a point before it was dropped by the REORG utility.

STYPE=Z

A column that is in the key of an index that was versioned before DB2 Version 8 was updated by an ALTER statement. Rebuild the index instead of recovering it

ICTYPE=F (full image copy)

One of the following situations occurred:

- An attempt was made to recover an index to a prior point-in-time and more than 16 versions of the index exist. (ALTER TABLE COLUMN was done more than 16 times to alter the index key). The point-in-time recovery is not allowed.
- An attempt was made to recover a compressed, versioned index, but the format of the image copy of the index is not correct. This situation occurs when the image copy was taken before DB2 code was applied to allow recovery of compressed, versioned indexes.

| In both cases, rebuild the index.

    ICTYPE=L

        STYPE=M

            Indicates that an SQL mass DELETE, TRUNCATE TABLE, DROP TABLE, or ALTER TABLE ROTATE PARTITION occurred on a segmented table space or universal table space. The LOWDSNUM column contains the OBID of the affected table. The BACKOUT YES option for RECOVER is not supported.

    ICTYPE=M (modify recovery)

        STYPE=R

            Indicates that the MODIFY RECOVERY utility deleted recovery information that is required for this recovery.

    ICTYPE=P (point in time recovery)

        An index is being recovered to a point in time that is different than the point in time of its underlying table space. The recover of the index is not allowed. Use the REBUILD INDEX utility instead.

    ICTYPE=P or E

        If the DSNU1528I message was previously issued for this object, then see that message for details.

System action: The RECOVER utility terminates.

User response: Run the REPORT RECOVERY utility, and analyze the report of the recovery information for the object.

If a SYSCOPY record for a REORG or LOAD REPLACE operation that redistributed partition data was found, add the missing partitions to the recover list. Use the LOWDSNUM and HIGHDSNUM values in the SYSCOPY record as a guide.

If a SYSCOPY record for a REORG operation that materialized an ALTER COLUMN INLINE LENGTH was found, add the missing objects to the point-in-time recover list.

If the object is an index, use the REBUILD INDEX utility to rebuild the index.

Severity: 8 (error)

Related concepts:

⮕ How RBA and LRSN values are displayed (DB2 Administration Guide)

⮕ Point-in-time recovery (DB2 Utilities)

DSNU557I   *csect-name* **-** *obj-type obj-qual.obj-name* **IS IN RECOVER PENDING**

Explanation: If the REBUILD utility issued this message, the corresponding table space to this index was rebuilt to a point in time with keyword TOCOPY, TOLOGPOINT, or TORBA. Thus, the index was placed in REBUILD pending status.

If the REBUILD INDEX utility issued this message, the index is in REBUILD pending status. Either duplicate index entries existed during the BUILD phase of the REBUILD INDEX, or invalid KEY/RIDs, which do not belong to the partition of the cluster index, were detected. The duplicate index entries or invalid KEY/RID entries must be resolved before starting the REBUILD INDEX utility again to REBUILD the index and remove the index from REBUILD pending status.

If the REORG utility with SHRLEVEL REFERENCE or CHANGE (or -TERM UTILITY) issued this message, the index is in REBUILD pending status. During the renaming of a data set during REORG, either a data set with the source name did not yet exist or a data set with the target name already existed.

User response: If the REBUILD or REORG utility (or -TERM UTILITY) issued this message, use the REBUILD INDEX utility to REBUILD this index.

If the REBUILD INDEX utility issued this message, take one of the following actions:

* Resolve the duplicate index entries or invalid keys and rerun the REBUILD INDEX utility.
* Use the REPAIR SET NORCVRPEND utility or the START DATABASE FORCE command to reset the REBUILD pending status. However, these two options will not correct the index.

DSNU558I • DSNU561I

DSNU558I   *csect-name* **ALL INDEXES OF TABLESPACE=** *table space-name* **ARE IN REBUILD PENDING**

Explanation: The REBUILD TORBA, TOLOGPOINT, or TOCOPY utility issues this message to inform the user of the state of all indexes belonging to the REBUILDed table space. All indexes were put in REBUILD pending status to prevent data and index inconsistency. All indexes should be REBUILDed before they are used.

System action: The REBUILD utility completed successfully.

User response: REBUILD or rebuild all indexes of the REBUILD table space.

Severity: 4 (warning)

DSNU559I   *csect-name* **-** *obj-type obj-qual.obj-name* **PARTITION** *partition-number* **IS IN RECOVER PENDING STATE**

Explanation: If the REBUILD utility issued this message, one partition of a partitioned table space is being rebuilt to a point in time with keyword TOCOPY, TOLOGPOINT, or TORBA. The corresponding index for that partition was placed in rebuild pending status.

If the REBUILD INDEX utility issued this message, the index or the partition of a cluster index is in rebuild pending status. Either duplicate index entries existed during the BUILD phase of the REBUILD INDEX or INDEX PART, or invalid KEY/RIDs, which do not belong to the partition of the cluster index, were detected. The duplicate index entries or invalid KEY/RID entries must be resolved before starting the REBUILD INDEX utility again to rebuild the index and remove the index from rebuild pending status.

If the REORG utility with SHRLEVEL REFERENCE or CHANGE (or -TERM UTILITY) issued this message, one partition of a partitioned index was placed in rebuild pending status. During the renaming of a data set during REORG, either a data set with the source name did not yet exist or a data set with the target name already existed.

User response: If the REBUILD or REORG utility (or -TERM UTILITY) issued the message, rebuild this index with the REBUILD INDEX utility.

If the REBUILD INDEX utility issued this message, take one of the following actions:
- Resolve duplicate index entries or invalid keys and rerun the REBUILD INDEX utility.
- Use the REPAIR SET NORCVRPEND utility or START DATABASE FORCE command to reset the rebuild pending status. However, these two options will not correct the index.

Severity: 4 (warning) or 8 (error)

DSNU560I   *csect-name* **- TABLESPACE=** *database.tablespace* **PART=** *partition number* **IS IN RECOVERY PENDING STATE**

Explanation: The table space partition of a table space on which the utility was running is in recovery pending status. A RECOVER TABLESPACE at the partition level is required.

System action: No action is taken.

User response: Take one of the following actions:
- Run the RECOVER TABLESPACE utility on the table space partition.
- Run the LOAD REPLACE utility on the table space partition.
- Use the REPAIR SET NORCVRPEND utility or the START DATABASE FORCE command to reset the recovery pending status. However, these two options will not correct the table space.

Severity: 4 (warning)

DSNU561I   *csect-name* **-** *obj-type obj-qual.obj-name* **PARTITION** *part-number* **IS IN CHECK PENDING**

Explanation: This message is issued to report the check pending state of the specified partition.

User response: If the object is a table space partition with referential relationships, then run the CHECK DATA utility to check for referential integrity violations.

If the object is a base table space partition, then run the CHECK DATA utility to validate the auxiliary columns.

If the object is an index space partition, then run the CHECK INDEX utility to verify that the index keys are consistent with the data in the table space partition.

Correct all errors before attempting to use the object again.

Severity: 4 (warning)

DSNU562I *csect name* **- TABLESPACE** *database.tablespace* **IS IN RECOVERY PENDING**

Explanation: The table space against which the utility was running is in recovery pending status. A RECOVER TABLESPACE is required.

System action: No action is taken.

User response: Take one of the following actions:
- Run the RECOVER TABLESPACE utility on the table space.
- Run the LOAD REPLACE utility on the table space.
- Use the REPAIR SET NORCVRPEND utility or the START DATABASE FORCE command to reset the recovery pending status. However, these two options will not correct the table space.

Severity: 4 (warning)

DSNU563I *csect name- obj-type obj-qual.obj-name* **IS IN CHECK PENDING**

Explanation: This message is issued to report the check pending state of the specified object.

User response: If the object is a table space with referential relationships, then run the CHECK DATA utility to check for referential integrity violations.

If the object is a base table space with LOB columns, then run the CHECK DATA utility to validate the auxiliary columns.

If the object is a LOB table space, then run the CHECK LOB utility to validate the structure of the LOB table space.

If the object is an index space, then run the CHECK INDEX utility to verify that the index keys are consistent with the data in the table space or table space partition.

Correct all errors before attempting to use the object again.

Severity: 4 (warning)

DSNU564I *csect* **- RESTART NOT ALLOWED FOR** *obj-type obj-qual.obj-name*

Explanation: Restart of the RECOVER utility is not allowed for one of the following reasons:
1. During the RESTORE phase, access to image copies failed. Therefore, RECOVER determined an earlier point of recovery to search for available image copies. RECOVER might not be able to re-create the original failures for restart.
2. The utility uses the LOGONLY option and was previously restarted with RESTART(CURRENT). It cannot now be started with RESTART(PHASE).

System action: The utility job terminates. If RECOVER acts on a list of objects, the restarted job terminates with the first object for which restart is not allowed.

User response: For case 1, execute REPORT RECOVERY on the object and also collect the output from the previous execution of RECOVER TABLESPACE. Using those two outputs and the console log, determine the reason for the failure during the RESTORE phase. Correct the reason for the failure, if necessary. Terminate the outstanding utility then rerun the RECOVER TABLESPACE without RESTART.

For case 2, restart the utility with RESTART(CURRENT) or restore the data sets and rerun the RECOVER job from the beginning.

Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

Severity: 8 (error)

DSNU565I   *csect-name* **CHECK CONSTRAINT** *constraint-name* **ON TABLE** *table-name* **WAS CREATED AFTER THE TARGET RECOVERY POINT**

Explanation: The check constraint *constraint-name* was created on table *table-name* after the target recovery point. The target recovery point is the specified time or RBA in the point-in-time recovery. The associated table space was placed in check pending status because the RECOVER utility cannot determine whether check integrity was maintained. Data in the table might have constraint violations.

System action: RECOVER processing continues.

User response: Resolve the check pending status.

Severity: 4 (warning)

Related concepts:

▶ Implications of adding parent or foreign keys (DB2 Administration Guide)

▶ Resetting the CHECK-pending status (DB2 Utilities)

Related tasks:

▶ Resetting CHECK-pending status (DB2 Utilities)

▶ Collecting data for failures after a point-in-time recovery (Collecting data)

▶ Adding referential constraints to existing tables (DB2 Administration Guide)

Related information:

▶ Recovering from referential constraint violation (DB2 Administration Guide)

DSNU566I   *csect-name* **RESTORE OF** *obj-type obj-qual.obj-name* **FROM DATA SET** *data-set-name* **COMPLETED, ELAPSED TIME =** *hh*:*mm*:*ss*

Explanation: This message is issued at the successful completion of the RESTORE phase of the RECOVER utility when the DFSMSdss RESTORE command is issued. The *data-set-name* identifies the DFSMS concurrent copy that was being restored. The elapsed time of the RESTORE is shown in hours, minutes, and seconds.

System action: RECOVER utility processing continues.

User response: No action is required.

DSNU568I   *csect-name obj-type obj-qual.obj-name* **IS IN INFORMATIONAL COPY-PENDING STATE**

Explanation: This message reports that the state of a table space or index has informational COPY- pending status. This state is an advisory state, so the object is available for read and write access. The Informational Copy Pending state is set for table spaces with the NOT LOGGED attribute when the table space is opened for update. The Informational Copy Pending state is set for indexes when a new recovery base needs to be established due to a non-recoverable event (such as a REORG or LOAD utility).

System action: None.

User response: Take a full image copy to remove the informational COPY-pending status.

Severity: 0 (informational)

DSNU569I   *csect-name* **RESTORE OF** *obj-type obj-qual.obj-name* **FROM DATA SET** *data-set-name* **FAILED WITH RETURN CODE** *return-code*

Explanation: This message is issued in the RESTORE phase of the RECOVER utility. The DFSMSdss RESTORE command was issued to restore the DFSMS concurrent copy named *data-set-name*. The return code is from DFSMSdss RESTORE.

System action: The RECOVER utility searches for the next recoverable point. DFSMSdss issues ADR messages to the job output to indicate the reason for the return code.

User response: You might want to investigate why this DFDSS RESTORE command failed. Collect any console messages. Look up the return code and any ADR messages in the appropriate DFSMSdss manual.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Job output.

Severity: 4 (warning)

DSNU570I *csect-name obj-type obj-qual.obj-name* **PARTITION** *part-num* **IS IN INFORMATIONAL COPY PENDING**

Explanation: An image copy of the index partition is required to establish a new base from which the index partition can be recovered. This is an advisory state, so the index is available for read or write access.

User response: Take an image copy of the index partition or of the entire index, depending on your backup and recovery procedures.

Severity: 0 (informational)

DSNU571I *csect-name* **ALL** *site* **SYSCOPY RECORDS FOR** *obj-type=* *obj-qual.obj-name* **PARTITION=***part-num* **HAVE BEEN DELETED**

Explanation: The MODIFY RECOVERY utility deleted all of the SYSCOPY table records for the specified table space or index space partition. Any related SYSLGRNX rows are deleted if DSNUM ALL is specified. The 'part-num' of specified table space or index space is no longer recoverable at the 'site'. The 'site' can be 'LOCALSITE' or 'RECOVERYSITE'.

System action: Utility processing continues.

System programmer response: The 'part-num' of specified table space or index space is no longer recoverable at the 'site' after this message is received. Use the COPY utility to create one or more copies for the 'site' to establish a base for future recovery.

Severity: 4 (warning)

DSNU572I *csect-name* **ALL** *site* **SYSCOPY RECORDS FOR** *obj-type=* *obj-qual.obj-name* **HAVE BEEN DELETED**

Explanation: The MODIFY RECOVERY utility deleted all of the SYSCOPY table records for the specified table space or index space. Any related SYSLGRNX rows are deleted if DSNUM ALL is specified. The table space or index space is no longer recoverable at the 'site'. The 'site' can be 'LOCALSITE' or 'RECOVERYSITE'.

System action: Utility processing continues.

System programmer response: The table space or index space is no longer recoverable at the 'site' after this message is received. Use the COPY utility to create one or more copies for the 'site' to establish a base for future recovery.

Severity: 4 (warning)

DSNU573I *csect-name* **NO BACKUP COPIES FOUND FOR THIS TABLESPACE OR DATASET**

Explanation: The MODIFY RECOVERY utility was unable to locate any SYSCOPY records older than the determined cleanup date. No records have been deleted.

System action: Processing completes normally.

Severity: 4 (warning)

DSNU575I *csect-name* **- MODIFY COMPLETED SUCCESSFULLY**

Explanation: This message reports the successful completion of the MODIFY utility. The requested cleanup of old SYSIBM.SYSCOPY entries was completed.

System action: The MODIFY utility completed normally.

DSNU577I • DSNU582I

Severity: 0 (informational)

DSNU577I   *csect-name* **MODIFY HAS NOT DELETED ALL SYSCOPY RECORDS AS REQUESTED.**

Explanation: If MODIFY RECOVERY is invoked with a partition-number (DSNUM), and the table space level recoverable point is older than the determined cleanup date, all SYSCOPY records after the table space recoverable point cannot be deleted.

System action: Processing continues normally.

Severity: 4 (warning)

DSNU578I   *csect-name* **SYSLGRNX INFORMATION FOR MEMBER** *member-name*

Explanation: In the data sharing environment, the SYSLGRNX records reported with DSNU513 are grouped by the respective member name. Each group member has header message DSNU578 to identify the member. This message precedes DSNU513.

System action: The RECOVER utility continues with the LOGAPPLY phase.

DSNU579I   *csect-name* **RECOVER UTILITY LOG APPLY AT LOGPOINT** *xxx*

Explanation: This message shows the progress of the LOGAPPLY phase of the RECOVER utility. The highest LOGPOINT that was applied is *xxx*. The value of logpoint is RBA in a non data-sharing environment and LRSN in a data sharing environment.

System action: The LOGAPPLY phase continues normally.

DSNU580I   *csect-name* **REPORT UTILITY COMPLETE, ELAPSED TIME=***hh/mm/ss*

Explanation: The message is issued at the normal completion of the REPORT utility. The time that elapsed since the beginning of the REPORT utility to its completion was 'hh' hours, 'mm' minutes, and 'ss' seconds.

System action: This message indicates the normal completion of the REPORT utility. Processing continues to the next utility statement in the utility job step, if any. If not, the utility job step is normally terminated, and the resources held become available for further processing. The UTILID of the utility job step becomes available for reuse.

Severity: 0 (informational)

DSNU581I   *csect-name* **REPORT RECOVERY** *obj-type obj-qual.obj-name*

Explanation: This message indicates the beginning of the recovery information report for a table space or index space.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU582I   *csect-name* **REPORT RECOVERY** *obj-type object-qualifier.object-name* **SYSCOPY ROWS AND SYSTEM-LEVEL BACKUPS**

Explanation: This message provides the header information for the SYSCOPY section of the REPORT RECOVER output. This section displays the SYSCOPY rows for the table space or index space. Whether the information is displayed at the partition level is determined by the DSNUM specification in the REPORT RECOVERY input statement.

This section also displays the SYSTEM LEVEL BACKUPs that can be used for recovering this table space or index space or specific partitions of it.

csect-name
> The name of the control section that issued the message.

The fields in the row are:

• TIMESTAMP

- ICTYPE, if a SYSCOPY row is present.

  The following delimiters might be present around the ICTYPE value to provide additional information when a SYSCOPY row is present:

  # #　　SYSCOPY entry that was created before any alteration was materialized.

  * *　　Non image copy SYSCOPY entry.

  < >　　Image copy prior to rebalancing of table space partitions.

  ( )　　Image copy prior to LOG(NO) event. For image copies of indexes, the LOG(NO) event might have occurred on its underlying table space.

- TYPE = SLB, if a row that is related to system-level backup is present.
- DSNUM
- FILESEQNO
- START_RBA
- SHRLEVEL
- DEVTYPE
- ICBACKUP
- DSNAME
- STYPE
- PIT_RBA
- DSVOLSER
- HIGHDSNUM
- LOWDSNUM

When a row related to system-level backups is present, the following fields are included:
- RBLP START
- STCK DATA
- DATA COMPLETE
- LRSN TOKEN
- MEMBER NAME

User response: No action is required.

Severity: 0

Related reference:

➡ SYSIBM.SYSCOPY table (DB2 SQL)

DSNU583I　　*csect-name* **SYSLGRNX ROWS FROM REPORT RECOVERY FOR** *obj-type obj-qual.obj-name*

Explanation: This is the header message for the SYSLGRNX section of the REPORT RECOVERY output. This section displays the SYSLGRNX rows for the table space, index space, or partition as specified in the REPORT RECOVERY input statement. The fields in the SYSLGRNX row are:
- UCDATE - Modification date
- UCTIME - Modification time
- START RBA - Starting RBA
- STOP RBA - Stopping RBA
- START LRSN - Starting LRSN
- STOP LRSN - Stopping LRSN
- DSNUM - Partition number
- MEMBER - Data sharing member

An invalid SYSLGRNX record is marked with '**' at the end. The invalid SYSLGRNX record cannot be used for recovery because the range is outside recovery. If records are marked with '**', then a DSNU594 message should have been issued prior to this.

A SYSLGRNX record created after the last FULL IMAGE COPY, LOAD REPLACE LOG YES, or REORG LOG YES is marked with '*' at the end.

System action: Processing continues normally.

Severity: 0 (informational)

DSNU584I   *csect-name* **REPORT RECOVERY** *obj-type obj-qual.obj-name* **BSDS VOLUMES**

Explanation: Message for the BSDS VOLUMES section of the REPORT RECOVERY output. This section contains BSDS volumes for the archive log (1 or 2) records associated with the table space or index space specified in the REPORT RECOVERY input statement.

An archive log record created after the last FULL IMAGE COPY, LOAD REPLACE LOG YES, or REORG LOG YES is marked with and asterisk (*) at the end.

System action: Processing continues normally

DSNU585I   *csect-name* **REPORT RECOVERY** *obj-type obj-qual.obj-name* **CURRENT**

Explanation: The message is issued in response to the CURRENT option of REPORT RECOVERY utility. It indicates that only the information since the last recoverable point for the specified table space or index space is reported.

System action: Normal utility processing continues.

Severity: 0 (informational)

DSNU586I   *csect-name* **REPORT RECOVERY** *obj-type obj-qual.obj-name* **SUMMARY:**

Explanation: This is the header message for the SUMMARY section of the REPORT RECOVERY utility. All the volumes to be used by the RECOVER utility are summarized.

The volume name is marked with '*' at the end if any archive log record in the volume was created after the last FULL IMAGE COPY, LOAD REPLACE LOG YES, or REORG LOG YES.

System action: Processing continues normally

DSNU587I   *csect-name* **REPORT TABLESPACE SET WITH TABLESPACE** *dbname.tsname*

Explanation: Header message for the REPORT TABLESPACESET output. The output contains all table spaces, tables, and dependent tables associated with the table space specified in the REPORT TABLESPACESET input statement.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU588I   *csect-name* **NO DATA TO BE REPORTED**

Explanation: The message is issued when no data can be found for a section of the REPORT output.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU589I   *csect-name* **REPORT RECOVERY** *obj-type obj-qual.obj-name* **COMPLETE**

Explanation: The message is issued at the normal completion of reporting the RECOVERY information for the table space or index space by the REPORT RECOVERY utility.

System action: This message indicates the normal completion of the REPORT RECOVERY utility. Processing continues to the next utility statement in the utility job step, if any. If not, the utility job step is normally terminated,

and the resources held become available for further processing. The UTILID of the utility job step becomes available for reuse. The output from REPORT RECOVERY is now available for RECOVER utility use.

Severity: 0 (informational)

DSNU590I   *csect-name object* **RESOURCE NOT AVAILABLE -** *object-name* **REASON = X'***reason-code***'**

Explanation: A required object *object-name* could not be accessed for the reason shown in *reason-code*. The required object can be a database, table space, or index space. If the object is in a stopped state and has partitions, then the stopped state applies to one or more of the partitions.

System action:
- If the return code is 8, utility processing is terminated.
- If the return code is 4, utility processing continues to the next request.

User response: Determine the cause for unavailability. The -DISPLAY DATABASE command can be used to determine the status. After ensuring the object is available, resubmit the job.

Severity: 8 (error) or 4 (warning)

DSNU591I   *csect-name* **AN UNSUCCESSFUL COMPLETION OF A LOG READ, REASON=X'***reason-code***'**

Explanation: The log manager subcomponent of DB2 detected an error during a log read of a SYSIBM.SYSCOPY record for DSNDB06.SYSCOPY, DSNDB01.DBD01, DSNDB01.SYSUTILX, or DSNDB01.SYSDBDXA

csect-name
        The name of the control section that issued the message.

reason-code
        A numeric value that indicates the reason for the failure.

System action: Processing continues.

Problem determination: For more information about the problem, see the information for the reason codes that is returned in this message.

Severity: 4 (warning)

Related concepts:

➡ DB2 reason codes (DB2 Codes)

DSNU592I   *csect-name* **REPORT RECOVERY INFORMATION FOR DATA SHARING MEMBER** *member-name*

Explanation: The REPORT RECOVERY utility displays this message as the heading for SYSLGRNX and BSDS records or BSDS summary information for each data sharing member. Records for this member follow, if there is data to report.

System action: Utility processing continues.

Severity: 0 (Informational)

DSNU593I   *csect-name* **REPORT RECOVERY ENVIRONMENT INFORMATION: MINIMUM RBA:** *rba1*
              MAXIMUM RBA: *rba2* **MIGRATING RBA:** *rba3*

Explanation: The REPORT utility issues this message to display recovery environment information used by the RECOVER utility. The maximum and minimum RBAs give the range of possible recovery points on the log before the start of data sharing. The migrating RBA is the RBA at the time of migration from DB2 Version 3 to Version 4. If there is data to report, other messages from the REPORT utility follow to display SYSCOPY, SYSLGRNX and BSDS records.

System action: Utility processing continues.

Severity: 0 (Informational)

DSNU594I • DSNU598I

Related concepts:

➡ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU594I   *csect-name* **DB2 ENCOUNTERED SYSLGRNX RECORDS THAT CANNOT BE USED FOR RECOVERY.**

Explanation: The utility found one or more SYSLGRNX records that cannot be used for recovery because the range is outside recovery. Invalid SYSLGRNX records are marked with '**' at the end of the record under message DSNU583I.

System action: REPORT utility processing continues. RECOVER utility terminates.

System programmer response: Run the RECOVER utility with the TOCOPY option.

Severity: 4 (REPORT - warning) or 8 (RECOVER - error)

DSNU595I   *csect-name* **ERROR IN GETTING THE HIGHEST MEMBER ID IN DATA SHARING.**

Explanation: The utility could not find a valid highest member ID. The highest member is the last member to join a data sharing group. This message is issued only in a data sharing environment.

System action: The utility stops abnormally.

System programmer response: This error might be an internal DB2 problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU596I   *csect-name* **ERROR IN GETTING THE MEMBER NAME FOR MEMBER ID**

Explanation: The REPORT utility could not find a valid member name for the member ID, or the member ID supplied is invalid. This message is issued only in a data sharing environment.

System action: The REPORT utility abends.

System programmer response: This might be a DB2 internal problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU597I   *csect-name* **INVALID SYSLGRNX INFORMATION FROM REPORT RECOVERY**

Explanation: This is the header message for the invalid SYSLGRNX records section of the REPORT RECOVERY output. This section lists SYSLGRNX records containing information that is invalid for recovery purposes.

System action: Utility processing continues.

User response: Examine the invalid SYSLGRNX records with a zero member ID which were generated before enabling data sharing. The MODIFY utility can be used to delete the invalid SYSLGRNX records.

Severity: 0 (Informational)

DSNU598I   *csect-name* **REPORT RECOVERY SYSTEM-LEVEL BACKUPS**

Explanation: This message provides the header information for the system-level backup section of the REPORT RECOVERY output. This section displays the system-level backups that are available. In a data sharing environment the system-level backups are provided in ascending order for the whole data sharing group.

csect-name
    The name of the control section that issued the message.

The system-level backup rows contain the following fields:
- START STCK DATA

- START STCK LOG RBLP
- DATA COMPLETE LRSN
- DATA/LOG COMPLETE DATE
- DATA/LOG COMPLETE LTIME
- LOCATION NAME
- TOKEN
- INCREMENTAL
- SUBSYSTEM ID MEMBER NAME z/OS
- CAPTURE CATALOG INFO
- LOG COPY POOL

System action: Processing continues.

User response: No action is required.

Severity: 0

Related concepts:

▶ Recovery information that REPORT provides (DB2 Utilities)

Related tasks:

▶ Recovering with a system-level backup (DB2 Utilities)

Related reference:

▶ REPORT (DB2 Utilities)

▶ RECOVER (DB2 Utilities)

DSNU599I *csect-name obj-type obj-qual.obj-name* **HAS BEEN RECOVERED TO A CONSISTENT POINT IN TIME WITH** *obj-type obj-qual.obj-name*

Explanation: This message is issued when an index and its underlying table space have been recovered to a consistent point in time. A consistent point in time is a common QUIESCE point or a common SHRLEVEL REFERENCE copy set (objects were copied in the same list with SHRLEVEL REFERENCE specified). The index is available for read and write access when the recover completes.

System action: Recover processing continues.

Severity: 0 (informational)

DSNU600I *csect-name* **INDEX** *creator.index-name* **POINTS TO A DIFFERENT DATABASE.**

Explanation: A specified index does not point to a table in the database selected. For a RUNSTATS Format 1 invocation, the database is the one specified (or defaulted) in the TABLESPACE parameter. For a RUNSTATS Format 2 invocation, the database is the one which contains the first index in the index-name list.

System action: All remaining RUNSTATS invocation parameters are validated and the executing utility job terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system; therefore, it cannot be restarted.

User response: Ensure that all indexes in the list point to the same database and resubmit the RUNSTATS job.

Problem determination: Refer to catalog table SYSIBM.SYSINDEXES to determine the table upon which the index is based. This table must be in the database and table space targeted by the RUNSTATS utility.

Severity: 8 (error)

DSNU601I *csect-name* **INDEX** *creator.index-name* **POINTS TO A DIFFERENT TABLESPACE.**

Explanation: A specified index does not point to a table in the table space selected. The table space is either the one specified in the TABLESPACE parameter or the one which contains the table pointed to by the first specified index.

System action: All remaining RUNSTATS invocation parameters are validated and the executing utility job

terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system; therefore, it cannot be restarted.

User response: Ensure that all indexes in the list point to tables in the targeted table space and resubmit the RUNSTATS job.

Problem determination: Refer to catalog table SYSIBM.SYSINDEXES to determine the table upon which the index is based. Using the table-name found in SYSIBM.SYSINDEXES, refer to catalog table SYSIBM.SYSTABLES to determine the containing table space.

Severity: 8 (error)

DSNU602I   *csect-name* **STATISTICS ARE NOT COLLECTED FOR NONPARTITIONED INDEX** *creator*.*index-name*

Explanation: A LOAD utility invocation specified the STATISTICS and PART keywords, but a nonpartitioned index exists. Statistics cannot be collected for nonpartitioned indexes for partition-level inline statistics, because a complete index scan is used to capture these statistics.

System action: No statistics are collected for the specified indexes, and processing continues. When INDEX ALL is specified, a DSU602I message is issued for each nonpartitioned index.

User response: Invoke the RUNSTATS utility for the object to collect statistics for the nonpartitioned indexes.

Severity: 4 (warning)

Related concepts:

➦ Nonpartitioned secondary indexes (Introduction to DB2 for z/OS)

Related reference:

➦ LOAD (DB2 Utilities)

➦ RUNSTATS (DB2 Utilities)

DSNU603I   *csect-name* **- OBJECT=***object-name* **HAS AN INCOMPLETE DEFINITION**

Explanation: The identified object (TABLE or TABLE SPACE) does not have a complete definition. This can be caused by one of these errors:

- Not defining any tables in the table space (RC=4).
- Not creating the accompanying partitioned index if the table space is partitioned (RC=8).

System action: If the return code is 8, utility processing terminates. If the return code is 4, utility processing continues to the next request.

User response: Complete the table space definition by creating at least one table in the table space, or create the accompanying partitioned index if the table space is partitioned.

Problem determination: Issue an SQL SELECT statement on the catalog to determine if the tables or a partitioned index was not created for the table space.

Severity: 8 (error) or 4 (warning)

DSNU604I   *csect-name object* **IS EMPTY**

Explanation: The object currently to be scanned by the RUNSTATS utility contains no records. The 'object' can be TABLESPACE or INDEXSPACE.

System action: The object currently to be scanned is recorded in the catalog as empty. The statistics for the object are updated and RUNSTATS continues to the next object, if there is one.

Severity: 0 (informational)

DSNU605I   *csect-name* **ERROR OCCURRED IN ACCESSING TABLESPACE** *database-name*.*tablespace-name*

Explanation: A previous error makes it impossible for RUNSTATS to scan TABLESPACE 'creator.tbspname'.

System action: The executing utility job step terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system; therefore, it cannot be restarted.

User response: Examine prior error messages for error identification. If the error can be corrected, resubmit the RUNSTATS utility in a new utility job step. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Prior messages identify this error. See the documentation for these messages in this manual for additional problem determination.

Severity: 8 (error)

DSNU606I   *csect-name* **USER NOT AUTHORIZED TO ACCESS TABLESPACE** *database-name*.*tablespace-name*

Explanation: This user is not authorized to execute the RUNSTATS utility against TABLESPACE 'creator.tbspname'.

System action: The executing utility job step terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system; therefore, it cannot be restarted.

User response: Contact your system administrator to obtain the proper authority, or have another user with the required authority execute the utility for you.

Severity: 8 (error)

DSNU607I   *csect-name* **ERROR OCCURRED ACCESSING** *obj-type obj-qual.obj-name*

Explanation: A previous error makes it impossible for RUNSTATS to scan *obj-type obj-qual.obj-name*.

obj-type   INDEX or INDEXSPACE that was specified

obj-qual   creator of the index or database-name of the index-space that was specified

obj-name
        index-name or index-space-name that was specified.

System action: The executing utility job step terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system; therefore, it cannot be restarted.

User response: Examine prior error messages for error identification. If the error can be corrected, resubmit the RUNSTATS utility in a new utility job step.

Problem determination: Prior messages identify this error. See the documentation for these messages in this manual for additional problem determination. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU608I   *csect-name* **USER NOT AUTHORIZED TO ACCESS** *obj-type obj-qual.obj-name*.

Explanation: This user is not authorized to execute the RUNSTATS utility against *obj-type obj-qual.obj-name*.

obj-type   INDEX or INDEXSPACE that was specified

obj-qual   creator of the index or database-name of the index-space that was specified

obj-name
        index-name or index-space-name that was specified.

System action: The executing utility job step terminates. Work performed by previous utilities in the job step

remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained in the system. Therefore it cannot be restarted.

User response: Contact your system administrator to obtain the proper authority, or have another user with the required authority execute the utility for you.

Severity: 8 (error)

DSNU609I   *csect-name* **TABLE** *table-name* **IS IN A DIFFERENT TABLESPACE**

Explanation: A table listed in the table name list of the RUNSTATS TABLE keyword was not found in the specified table space.

System action: The executing utility job step terminates. Work performed by previous utilities in the job step remains in effect. Subsequent utility control statements are flushed. No record of the utility job step is retained by the system; therefore it cannot be restarted.

User response: Ensure that all tables in the list are in the specified table space and resubmit the utility.

Problem determination: Refer to the catalog table SYSIBM.SYSTABLES to determine the table space in which the table is located.

Severity: 8 (error)

DSNU610I   *csect-name catalog-table-name* **CATALOG UPDATE FOR** *object-name* **SUCCESSFUL**

Explanation: This message indicates that the updating of the statistics columns in the CATALOG TABLE 'catalog-table-name' was successful.

System action: Processing continues with the next catalog table.

Severity: 0 (informational)

DSNU611I   *csect-name* **COLUMN** *column-name* **NOT FOUND FOR TABLE** *creator*.*table-name*

Explanation: A column name, given as an operand of the COLUMN keyword, was not found in the specified table.

System action: Validation of the utility invocation statement is completed and the executing utility job step is terminated.

User response: Ensure that the column name given in the message is correct for the table specified.

Problem determination: Check that the column name specified in the utility invocation statement is identical to that found in the SYSIBM.SYSCOLUMNS catalog table. Correct any errors and reissue the utility job.

Severity: 8 (error)

DSNU612I   *csect-name* **SYSTABLESPACE CATALOG STATISTICS FOR** *tablespace-name* **NACTIVE =** *nactive-value*
               NACTIVEF = *nactivef-value*

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the specified table space.

csect-name
> The name of the control section that issued the message.

tablespace-name
> The name of the table space.

nactive-value
> The number of active pages in the table space, as an integer value.

nactivef-value
> The number of active pages in the table space, as a floating point value.

This message is routed to SYSPRINT.

System action: RUNSTATS processing continues.

User response: No action is required.

Related reference:

▶ Statistics used for access path selection (DB2 Performance)

▶ RUNSTATS (DB2 Utilities)

▶ SYSIBM.SYSTABLESPACE table (DB2 SQL)

DSNU613I   *csect-name* **SYSTABLEPART CATALOG STATISTICS FOR** *tablespace-name* **PARTITION** *part-number*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics values that were collected:

```
CARD = card-value
CARDF = cardf-value
NEARINDREF = nearindref-value
FARINDREF = farindref-value
PERCACTIVE = peractive-value
PERCDROP = percdrop-value
PAGESAVE = pagesave-value
SPACE = space-value
SPACEF = spacef-value
PQTY = pqty-value
SQTY = sqty-value
DSNUM = dsnum-value
EXTENTS = extents-value
```

csect-name
> The name of the control section that issued the message.

part-number
> The value that identifies the partition or data set. A 0 value means that the statistics apply to the entire table space. A non-zero value indicates that the statistics apply to that partition that is identified by the value.

card-value
> The number of rows in the reported partition number, as an integer value.

cardf-value
> The number of rows in the reported partition number, as a floating point value.

nearindref-value
> The number of rows that were relocated near their original page.

farindref-value
> The number of rows that were relocated far from their original page.

peractive-value
> The percentage of space occupied by actual data from active tables.

percdrop-value
> The percentage of space occupied by rows of data from dropped tables.

pagesave-value
> The percentage of pages saved in the table space or partition by using data compression.

space-value
> The space currently allocated for all extents, measured in kilobytes, as an integer value.

spacef-value
> The space currently allocated for all extents, measured in kilobytes, as a floating point value.

pqty-value
> The primary space allocation for the data set in 4K blocks.

sqty-value
> The secondary space allocation for the data set in 4K blocks.

dsnum-value
> The number of data sets.

DSNU614I

extents-value
> The number of extents for the last data set of the object.

This message is routed to SYSPRINT.

System action: RUNSTATS processing continues.

User response: No action is required.

Severity: 0

Related reference:

➡ Statistics used for access path selection (DB2 Performance)

➡ RUNSTATS (DB2 Utilities)

➡ SYSIBM.SYSTABLESPACE table (DB2 SQL)

DSNU614I *csect-name* **SYSTABLES CATALOG STATISTICS FOR** *table-nametable-name*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for the specified table.

```
CARD = card-value
CARDF = cardf-value
NPAGES = npages-value
NPAGESF = npagesf-value
PCTPAGES = pctpages-value
PCTROWCOMP = pctrowcomp-value
AVGROWLEN = avgrowlen-value
SPACEF = spacef-value
```

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

card-value
> The number of rows in the table.
>
> For large table spaces, the value of CARD is set to -1.

cardf-value
> The number of rows in the table, as a floating point value.
>
> For large table spaces, the value of CARDF is set to -1.

npages-value
> The number of pages on which rows of the table appear.

pctpages-value
> The percentage of total pages of the table space that contain rows from this table.

pctrowcomp-value
> The percentage of rows compressed within the total number of active rows in the table.

avgrowlen-value
> The total number of bytes divided by the number of rows.

spacef-value
> The space currently allocated for all extents, measured in kilobytes.

This message is routed to SYSPRINT.

System action: RUNSTATS processing continues.

User response:

User response: No action is required.

Severity: 0

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSTABLES table (DB2 SQL)

DSNU615I *csect-name* **SYSCOLUMNS CATALOG STATISTICS FOR** *column-name*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for column *column-name*. Message DSNU619I is issued to identify the table.

```
COLCARD   = colcard-value
COLCARDF  = colcardf-value
HIGH2KEY  = high2key
LOW2KEY   = low2key
```

csect-name
> The name of the control section that issued the message.

column-name
> The name of the column.

colcard-value
> The number of distinct values in the column. For a large table spaces, the value of COLCARD is set to -1.

colcardf-value
> The number of distinct values in the column, as a floating point value.

high2key-value
> The second highest value of the column.

low2key-value
> The second lowest value of the column.

For empty partitions, RUNSTATS sets the values for HIGH2KEY and LOW2KEY to null ("X).

System action: Processing continues.

User response: No action is required.

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSINDEXES table (DB2 SQL)

DSNU616I *csect-name* **SYSCOLDIST CATALOG STATISTICS FOR** *column-name*, *column-name*, *column-name* **...**

```
CARDINALITY = cardinality
FREQUENCY   COLVALUE
frequency   X'hex-string'
frequency   X'hex-string'
..          ..
frequency   X'hex-string'
```

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the indexed column *column-name*. If more than one column is listed, then the statistics collected and the column values reported represent the concatenation of those columns.

The value for CARDINALITY is the number of distinct values for the listed columns. This statistic appears only if the KEYCARD option was specified.

The values under FREQUENCY are the same as those that are inserted into the DB2 catalog. Consider each 'frequency' as a four-digit number with an implied decimal point two places to the left of the final digit. The

DSNU617I

resulting number is the percentage of total rows in the table that contain a particular value for the named index columns.

Frequency is associated with the occurrence of a single value which is listed under COLVALUE.

Frequencies for the most frequent values are collected by default, or if the FREQVAL option is specified and given a COUNT value greater than zero. If the default is taken, then the ten most frequently occurring values are reported in descending order of frequency.

- If fewer than ten values appear, then there are fewer than ten distinct values in the index.
- If no values appear, then the values are uniformly distributed.
- If frequencies appear but column values do not, then the invoker of RUNSTATS is not authorized to read the table. Message DSNU619I is issued to identify the table.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action.

Severity: 0 (informational)

Related reference:

➡ Statistics used for access path selection (DB2 Performance)

➡ RUNSTATS (DB2 Utilities)

➡ SYSIBM.SYSCOLDIST table (DB2 SQL)

Related information:
"DSNU619I" on page 756

DSNU617I   *csect-name* **SYSINDEXES CATALOG STATISTICS FOR** *index-name*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for the specified index.

```
CLUSTERED = clustered-value
CLUSTERRATIO = clusterratio-value
CLUSTERRATIOF = clusterratiof-value
FIRSTKEYCARD = firstkeycard-value
FIRSTKEYCARDF = firstkeycardf-value
FULLKEYCARD = fullkeycard-value
FULLKEYCARDF = fullkeycardf-value
NLEAF = nleaf-value
NLEVELS = nlevels-value
SPACEF = spacef-value
DATAREPEATFACTORF = datarepeatfactorf-value
```

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index.

clustered
> Whether the table is actually clustered by this index (YES or NO)

clusterratio-value
> The percentage of data rows that are in clustering order for this index, as an integer value

clusterratiof-value
> A number between 0 and 1 that when multiplied by 100 gives the percentage of data rows in the partition that are in clustering order for this index, as a floating point value

firstkeycard-value
> The number of distinct values of the first key column, as an integer value

firstkeycardf-value
> The number of distinct values of the first key column, as a floating point value

fullkeycard-value
> The number of distinct values of the key, as an integer value

fullkeycardf-value
> The number of distinct values of the key, as a floating point value

nleaf-value
> The number of active leaf pages in the index

nlevels-value
> The number of levels in the index tree

spacef-value
> The space currently allocated for all extents, measured in kilobytes.

datarepeatfactorf-value
> The anticipated number of data pages repeatedly scanned, following index key order

For a LARGE table space, the values of FIRSTKEYCARD and FULLKEYCARD are set to -1. RUNSTATS uses floating point fields to report these statistics.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSINDEXES table (DB2 SQL)

DSNU618I   *csect-name* **SYSINDEXPART CATALOG STATISTICS FOR** *index-name* **PARTITION** *part-number*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for the specified index.

```
CARD = card-value
CARDF = cardf-value
NEAROFFPOS = nearoffpos-value
NEAROFFPOSF = nearoffposf-value
FAROFFPOS = faroffpos-value
FAROFFPOSF = faroffposf-value
LEAFDIST = leafdist-value
SPACEF = spacef-value
DSNUM = dsnum-value
EXTENTS = extents-value
LEAFNEAR = leafnear-value
LEAFFAR = leaffar-value
PSEUDO_DEL_ENTRIES= psuedo_del_entries-value
```

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index.

part-number
> The value that identifies the partition or data set. If this value equals 0, the statistics apply to the entire index. If it is a non-zero value, the statistics apply to the specified partition of the index.

DSNU619I

card-value
> The number of rows in the partition or index, as an integer value. For large table spaces, the value of CARD is set to -1.

cardf-value
> The number of rows in the partition or index, as a floating point value.

nearoffpos-value
> The number of rows pointed to that are near but not at their optimal positions, as an integer value. For large table spaces, the value of NEAROFFPOS is set to -1.

nearoffposf-value
> The number of rows pointed to that are near but not at their optimal positions, as a floating point value.

faroffpos-value
> The number of rows pointed to that are far from their optimal positions, as an integer value. For large table spaces, the value of FAROFFPOS is set to -1.

faroffposf-value
> The number of rows pointed to that are far from their optimal positions, as a floating point value.

leafdist-value
> 100 times the average distance in page IDs between successive leaf pages during a sequential access of the index.

spacef-value
> The space currently allocated for all extents, measured in kilobytes.

dsnum-value
> The number of data sets.

extents-value
> The number of extents for the last data set of the object.

leafnear-value
> The number of leaf pages physically near the previous leaf page for successive active leaf pages.

leaffar-value
> The number of leaf pages located physically far away from the previous leaf pages for successive active leaf pages accessed in an index scan.

psuedo_del_entries-value
> The number of entries that are logically deleted.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Related reference:

⬤ Statistics used for access path selection (DB2 Performance)

⬤ RUNSTATS (DB2 Utilities)

⬤ SYSIBM.SYSINDEXPART table (DB2 SQL)

DSNU619I   *csect-name* **- USER** *auth_id* **DOES NOT HAVE PRIVILEGE ON TABLE** *table_name*

Explanation: The RUNSTATS utility issues this message to report that authorization ID 'auth_id' does not have privilege on table 'table_name'. Some column values reported by messages DSNU615I, DSNU616I, DSNU625I, and DSNU626I contain the null value.

System action: RUNSTATS processing continues.

User response: To obtain the key values, the user must be granted authority on the table. The RUNSTATS utility can be rerun with the report option.

Severity: 0 (informational)

| **DSNU620I** *csect-name* **RUNSTATS CATALOG STATISTICS STATSTIME =** *timestamp*

Explanation: The RUNSTATS utility issues the DSNU620I message, when utility processing completes, to report the timestamp value when statistics were collected or reset.

csect-name
  The name of the control section that issued the message.

timestamp
  The timestamp value of the statistics. The same value is common to all of the statistics that were updated by a single invocation of RUNSTATS.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

➥ Statistics used for access path selection (DB2 Performance)

➥ RUNSTATS (DB2 Utilities)

DSNU621I *csect-name* **RUNSTATS COMMIT/ROLLBACK FAILED**

Explanation: The RUNSTATS utility updated the catalog with gathered statistics and has tried either to commit those changes or to perform rollback processing to remove them following an error. The commit or rollback failed.

System action: RUNSTATS processing is terminated. Subsequent utility control statements in the utility job step are executed.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: This is an internal error.

Severity: 8 (error)

DSNU622I *csect-name* **CATALOG ACCESS FOR** *object-name* **NOT SUCCESSFUL, REASON=**reason-code

Explanation: Due to catalog access error for object *object-name* and reason code *reason-code*, the catalog update, insert, or select was not successful for the object when inline statistics was employed.

User response: Run the RUNSTATS utility against the object.

Severity: 4 (warning)

DSNU623I *csect-name* **-** *catalog-table-name* **CATALOG NOT UPDATED WITH AGGREGATE STATISTICS FOR** *object-name* **BECAUSE SOME PARTITIONS HAVE NO VALID STATISTICS**

Explanation: Aggregate statistics were requested, but partition statistics do not exist for every partition. Without partition statistics, the aggregate statistics cannot be calculated. The statistics for *object-name* are not updated in the specified DB2 catalog table. If you specified UPDATE NONE, then this message will not show up.

One possible cause for this message is that someone altered the length of a varying-length column and then tried to run RUNSTATS with the PART option.

System action: Processing continues.

User response: Run the RUNSTATS utility to collect statistics for:

- The entire table space, table, column or index, or
- Every partition.

Severity: 4 (warning)

DSNU624I   *csect-namecsect-name* **SYSTABSTATS CATALOG STATISTICS FOR** *table-name* **PARTITION** *part-number*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for the specified partition of the table space that contains the table.

```
CARD = card-value
CARDF = cardf-value
NPAGES  = npages-value
PCTPAGES = pctpages-value
NACTIVE = nactive-value
PCTROWCOMP = pctrowcomp-value
```

card-value
>   The number of rows in the table space partition, as an integer value

cardf-value
>   The number of rows in the table space partition, as a floating point value

npages-value
>   The number of pages on which rows of the table space partition appear

pctpages-value
>   The percentage of total pages of the table space partition that contain rows from this table

nactive-value
>   The number of active pages in the table space partition

pctrowcomp-value
>   The percentage of rows compressed within the total number of active rows in the partition

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Related reference:

➡ Statistics used for access path selection (DB2 Performance)

➡ RUNSTATS (DB2 Utilities)

➡ SYSIBM.SYSTABSTATS table (DB2 SQL)

DSNU625I   *csect-name* **SYSCOLSTATS CATALOG STATISTICS FOR** *column-name* **PARTITION** *part-num*

```
COLCARD       = cardinality
HIGHKEY       = highkey
HIGH2KEY    = high2key
LOW2KEY     = low2key
LOWKEY       = lowkey
```

Explanation: The RUNSTATS utility issues this message to report the statistics collected for column *column-name* partition *part-num* of a table. Message DSNU619I is issued to identify the table.

cardinality
>   The number of distinct values in the column for the partition

highkey
>   The highest value of the column in the partition

high2key
>   The second highest value of the column in the partition

low2key
>   The second lowest value of the column in the partition

lowkey   The lowest value of the column in the partition

For empty partitions, RUNSTATS set the values for *highkey*, *high2key*, *low2key*, and *lowkey* to null ("X).

This message is routed to SYSPRINT.

User response: No action.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSCOLSTATS table (DB2 SQL)

DSNU626I   *csect-name* **SYSCOLDISTSTATS CATALOG STATISTICS FOR** *column-name*, *column-name*, *column-name* **... PARTITION** *part-num*

```
CARDINALITY = cardinality
FREQUENCY   COLVALUE
frequency   X'hex-string'
frequency   X'hex-string'
..          ..
frequency   X'hex-string'
```

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the indexed column *column-name* partition *part-num*. If more than one column is listed, then the statistics collected and the column values reported represent the concatenation of those columns.

The value for CARDINALITY is the number of distinct values for the listed columns. This statistic appears only if the KEYCARD option was specified.

The values under FREQUENCY are the same as those that are inserted into the DB2 catalog. Consider each 'frequency' as a four-digit number with an implied decimal point two places to the left of the final digit. The resulting number is the percentage of total rows in the table that contain a particular value for the named index columns.

Frequency is associated with the occurrence of a single value which is listed under COLVALUE.

Frequencies for the most frequent values are collected by default, or if the FREQVAL option is specified and given a COUNT value greater than zero. If the default is taken, then the ten most frequently occurring values, plus the lowest and highest values in the partition (if not already included in the ten greatest frequencies), are reported in descending order of frequency.

- If fewer than ten values appear, then there are fewer than ten distinct values in the index.
- If no values appear, then the values are uniformly distributed.
- If frequencies appear but column values do not, then the invoker of RUNSTATS is not authorized to read the table. Message DSNU619I is issued to identify the table.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSCOLDISTSTATS table (DB2 SQL)

Related information:

DSNU627I   *csect-name* **SYSINDEXSTATS CATALOG STATISTICS FOR** *index-name* **PARTITION** *part-number*

Explanation: The RUNSTATS utility issues this message on multiple lines to report the statistics collected for the specified partition of the specified index.

```
CLUSTERRATIO =clusterratio-value
CLUSTERRATIOF = clusteratiof-value
FIRSTKEYCARD = firstkeycard-value
FIRSTKEYCARDF = firstkeycardf-value
FULLKEYCARD = fullkeycard-value
FULLKEYCARDF = fullkeycardf-value
NLEAF = nleaf-value
NLEVELS = nlevels-value
KEYCOUNT = keycount-value
KEYCOUNTF = keycountf-value
DATAREPEATFACTORF= datarepeatfactorf-value
```

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index.

part-number
> The value that identifies the partition or data set.

clusterratio-value
> The percentage of data rows in the partition that are in clustering order for this index, as an integer value.

clusterratiof-value
> The percentage of data rows in the partition that are in clustering order for this index, as a floating point value.

firstkeycard-value
> The number of distinct values of the first key column in the partition, as an integer value.

firstkeycardf-value
> The number of distinct values of the first key column in the partition, as a floating point value.

fullkeycard-value
> The number of distinct values of the key in the partition, as an integer value.

fullkeycardf-value
> The number of distinct values of the key in the partition, as a floating point value.

nleaf-value
> The number of active leaf pages in the index partition.

nlevels-value
> The number of levels in the index tree in the partition.

keycount-value
> The number of index keys (entries) in the partition, as an integer value.

keycountd-value
> The number of index keys (entries) in the partition, as a floating point value.

datarepeatfactorf-value
> The anticipated number of data pages repeatedly scanned, following index key order.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSINDEXSTATS table (DB2 SQL)

DSNU628I   *csect-name* **STATISTICS WILL NO LONGER BE COLLECTED ON RESTART**

Explanation: Since inline statistics is not restartable, Statistics may not be collected on the object that was being processed during the restart of the LOAD/REORG utility. If a list of objects is being processed, then objects that are not processed yet will still have statistics collected. The error message will be issued on restart of a LOAD/REORG utility with the STATISTICS keyword specified.

User response: Run the runstats utility on the object after the utility execution unless the DSNU620I message is present, indicating that the statistics were collected.

Severity: 4 (warning)

DSNU629I   *csect-name* **INLINE STATISTICS IS NOT SUPPORTED FOR** *object-name*

Explanation: The REORG TABLESPACE utility or REBUILD INDEX utility with the STATISTICS keyword was run for a catalog or directory table space. The collection of inline statistics during REORG TABLESPACE or REBUILD INDEX is not supported for that table space.

tablespace-name
> The name of the table space.

System action: Processing continues.

Inline statistics are not collected.

User response: Remove the keyword STATISTICS from the utility job statement, and rerun the utility. Then run the RUNSTATS utility to collect statistics.

Severity: 4 (warning)

Related reference:

REBUILD INDEX (DB2 Utilities)

REORG TABLESPACE (DB2 Utilities)

DSNU630I   *csect-name* **- SYSLOBSTATS CATALOG STATISTICS FOR** *LOB table space* **AVGSIZE =** *avgsize*
> **FREESPACE =** *freespace* **ORGRATIO =** *orgratio*

Explanation: The RUNSTATS utility issues this message to report statistics collected for the LOB table space.

AVGSIZE
> Average size of a LOB in the LOB table space.

FREESPACE
> Amount of free space in the LOB table space.

ORGRATIO
> The percentage of organization in the LOB table space. The following values indicate the state of the LOB table space:

100.00   The table space is completely organized or it is empty.

1   The table space is disorganized.

0.00   The table space is completely disorganized.

System action: RUNSTATS processing continues.

DSNU632I   *csect-name*- SYSCOLDIST CATALOG HISTOGRAM STATISTICS FOR *column-name*, *column-name*, ### QUANTILE *quan-num* LOWVALUE = X'*hex-value*' HIGHVALUE = X'*hex-value*' CARDF = *cardf-value* FREQUENCYF = *frequencyf-value*

Explanation: The RUNSTATS utility issues this message to report the histogram statistics that are collected for the specified column or column group for each quantile, as indicated by *quan-num*.

For single-column, the LOWVALUE and HIGHVALUE represent the column value, low bound and high bound, for each quantile range. For column-group, the LOWVALUE and HIGHVALUE represent the concatenation of those columns.

CARDF represents the number of distinctive values in each quantile. For column group, CARDF represents the number of distinctive combinations of that column group in each quantile.

FREQUENCYF represents the percentage of rows in each quantile.

If the total number of distinctive values for that column or column group is less than the specified number of quantiles, then the quantiles collected is up to the number of distinctive values.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

➡️ Statistics used for access path selection (DB2 Performance)

➡️ RUNSTATS (DB2 Utilities)

➡️ SYSIBM.SYSCOLDIST table (DB2 SQL)

DSNU633I   *csect-name*- HISTOGRAM STATISTICS NOT COLLECTED FOR INDEX WITH MIXED ORDERING ON COLUMNS

Explanation: The RUNSTATS utility issues this message to warn users that no histogram statistics are collected for the index key columns of mixed order.

System action: Processing continues with the histogram option ignored.

Severity: 4 (warning)

DSNU634I   *csect-name*- NUMBER SPECIFIED FOR NUMQUANTILES EXCEEDS MAXIMUM

Explanation: The RUNSTATS utility issues this message to warn that the number of quantiles that the user specified exceeds the maximum limit of 100.

System action: Processing continues with the maximum limit as the number of quantiles.

Severity: 4 (warning)

DSNU635I   COLGROUP STARTING WITH first-column-name IS TOO LONG

Explanation: The length of the COLGROUP value has exceeded the length of COLVALUE column defined in SYSIBM.SYSCOLDIST catalog table.

System action: The executing utility job step terminates.

User response: Alter the COLGROUP specification such that the maximum length of the COLGROUP will not exceed the length of COLVALUE column defined in SYSIBM.SYSCOLDIST catalog table and resubmit the RUNSTATS utility. If the length of a COLGROUP with a single column specified exceeds the maximum length of COLVALUE, then the distribution statistics cannot be collected on that COLGROUP.

Severity: 8 (error)

DSNU636I *csect-name* **SYSCOLDISTSTATS CATALOG HISTOGRAM STATISTICS FOR** *column-name*,
column-name, **### PARTITION** *part-num* **QUANTILE** *quan-num* **LOWVALUE = X'***hex-value***'**
HIGHVALUE = X'*hex-value***' CARDF =** *cardf-value* **FREQUENCYF =** *frequencyf-value*

Explanation: The RUNSTATS utility issues this message to report the histogram statistics that are collected for the specified column or column group for each quantile, as indicated by *quan-num*, for a partition, as indicated by *part-num*.

For single-column, the LOWVALUE and HIGHVALUE represent the column value, low bound and high bound, for each quantile range. For column-group, the LOWVALUE and HIGHVALUE represent the concatenation of those columns.

CARDF represents the number of distinctive values in each quantile. For column group, CARDF represents the number of distinctive combinations of that column group in each quantile.

FREQUENCYF represents the percentage of rows in each quantile.

If the total number of distinctive values for that column or column group is less than the specified number of quantiles, then the quantiles collected is up to the number of distinctive values.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

➦ Statistics used for access path selection (DB2 Performance)

➦ RUNSTATS (DB2 Utilities)

➦ SYSIBM.SYSCOLDISTSTATS table (DB2 SQL)

DSNU637I *csect-name* **NUMBER OF** *option-name* **SPECIFIED EXCEEDS MAXIMUM LIMIT ALLOWED OF**
maximum-allowed

Explanation: The utility control statement contains too many instances of an option.

csect-name
    The name of the control section that issued the message.

option-name
    The name of the option, such as COLGROUP

maximum-allowed
    The maximum number of instances of the indicate option that is permitted in a utility control statement.

System action: Utility processing does not start.

User response: Reduce the number of *option-name* option instances in the utility control statement.

Severity: 8 (error)

DSNU638I *csect-name* **- UTILITY NOT ALLOWED TO RUN ON TABLE SPACE =** *tablespace-name*
PART*part-number* **BECAUSE OF SYSTEM LEVEL READ-ONLY SPECIFICATIONS**

Explanation: An attempt to execute a utility failed, because the target table space or partition resides in data sets that are defined at the system level as read-only.

csect-name
    The name of the control section that issued the message.

tablespace-name
    The name of the table space.

part-number

> The value that identifies the partition or data set. PART *part-number* is issued only if the utility was requested at the partition level.

System action: Processing continues to the next table space or index space, if any.

User response: Contact the system programmer.

Severity: 4 (warning)

DSNU639I   *csect-name* **- UTILITY NOT ALLOWED TO RUN ON INDEX SPACE =** *index-space-name* **PART** *part-number* **BECAUSE OF SYSTEM LEVEL READ-ONLY SPECIFICATIONS**

Explanation: An attempt to execute a utility failed, because the target index space or partition resides in data sets that are defined at the system level as read-only.

csect-name

> The name of the control section that issued the message.

index-space-name

> The name of the index space.

part-number

> The value that identifies the partition or data set. PART *part-number* is issued only if the utility was requested at the partition level.

System action: Processing continues to the next table space or index space, if any.

User response: Contact the system programmer.

Severity: 4 (warning)

DSNU640I   *csect-name* **DATA SET=** *nnn* **OF TABLESPACE=** *tttttttt* **IN DATABASE=** *dddddddd* **HAS BEEN MIGRATED BY HSM**

Explanation: The STOSPACE utility has found that a DB2 data set has been migrated by the Data Facility Hierarchical Storage Manager. The amount of storage space used cannot be determined for DATA SET 'nnn' of TABLESPACE 'tttttttt' in DATABASE 'dddddddd'. The space allocation information in the catalog will not include the storage used by the migrated data set.

System action: Processing continues to the next data set in the storage group, if any.

System programmer response: If required, the data set may be recalled using HSM, and the STOSPACE utility may be run again.

Severity: 0 (informational)

DSNU641I   *csect-name* **DATA SET=** *nnn* **OF INDEXSPACE=** *iiiiiiii* **IN DATABASE=** *dddddddd* **HAS BEEN MIGRATED BY HSM**

Explanation: The STOSPACE utility has found that a DB2 data set has been migrated by the Data Facility Hierarchical Storage Manager. The amount of storage space used cannot be determined for DATA SET 'nnn' of INDEXSPACE 'iiiiiiii' in DATABASE 'dddddddd'. The space allocation information in the catalog will not include the storage used by the migrated data set.

System action: Processing continues to the next data set in the storage group, if any.

System programmer response: If required, the data set may be recalled using HSM, and the STOSPACE utility may be run again.

Severity: 0 (informational)

DSNU642I   *csect-name* **UNABLE TO ACCESS DBD FOR DATABASE =** *dddddddd***, TABLESPACE =** *tttttttt*

Explanation: A database descriptor control block could not be accessed by the STOSPACE utility. Information in the DBD is required before space allocation data can be obtained from the appropriate ICF catalog. Storage space used by table space 'tttttttt' in database 'dddddddd' cannot be determined.

System action: Processing continues with the next storage group.

System programmer response: Determine if the reason for the DBD unavailability may require system corrective action.

User response: Determine the cause for unavailability. The database may be stopped or the DBD may be marked as invalid. The -DISPLAY DATABASE command can be used to determine the status.

Problem determination: A DSNT501I message may have been sent to the operator console. If so, see the problem determination section for message DSNT501I.

Severity: 4 (warning)

DSNU643I   *csect-name* **UNABLE TO ACCESS DBD FOR DATABASE =** *dddddddd*, **INDEXSPACE =** *tttttttt*

Explanation: A database descriptor control block could not be accessed by the STOSPACE utility. Information in the DBD is required before space allocation data can be obtained from the appropriate ICF catalog. Storage space used by index space 'tttttttt' in database 'dddddddd' cannot be determined.

System action: Processing continues with the next storage group.

System programmer response: Determine if the reason for the DBD unavailability may require system corrective action.

User response: Determine the cause for unavailability. The database may be stopped or the DBD may be marked as invalid. The -DISPLAY DATABASE command can be used to determine the status.

Problem determination: A DSNT501I message may have been sent to the operator console. If so, see the problem determination section for message DSNT501.

Severity: 4 (warning)

DSNU644I   *csect-name* **USER** *user-name* **NOT AUTHORIZED FOR STOSPACE UTILITY**

Explanation: Utility processing for the user indicated on the job statement failed, because the user is not authorized to execute the STOSPACE utility.

System action: Utility processing is terminated without invoking the requested utility or utilities.

System programmer response: If the USER specified on the job statement is incorrect, correct it, and resubmit the job. To run the STOSPACE utility, the user must either have SYSADM authority or have been granted the STOSPACE privilege. Use the SQL GRANT command to give the user the required authority.

User response: Contact the systems programmer for assistance with your security classification.

Severity: 8 (error)

DSNU645I   *csect-name* **NO STORAGE GROUPS NAME FOUND**

Explanation: You requested the STOSPACE utility to report on all storage groups, but no storage group was found.

System action: This message indicates completion of the STOSPACE utility. Processing continues to the next utility statement in the utility job step, if any. If not, the utility job step is normally terminated. Resources are released to their prior availability state. The UTILID of the utility job step becomes available for reuse.

System programmer response: This message indicates either that no storage groups were defined by the installation process (or by subsequent SQL procedures) or they have all been dropped. If this is an error, the required storage groups must be re-created with the CREATE STOGROUP SQL statement.

User response: This message indicates that there are no storage groups defined. Consult the system programmer to determine the reason for this condition.

Severity: 4 (warning)

This condition does not prevent subsequent processing.

DSNU646I   *csect-name* **STORAGE GROUP NAME** *nnnnnnnn* **NOT FOUND**

Explanation: This message is issued by STOSPACE when one of the requested storage groups is not found.

The following circumstances may exist:

1. When an inconsistency occurs between SYSIBM.SYSSTOGROUP at the beginning of STOSPACE execution and SYSIBM.SYSSTOGROUP during or at the end of processing. The inconsistency may be due to a storage group being dropped during the processing of STOSPACE utility.
2. The storage group name is misspelled.

System action: Processing proceeds normally to the next storage group if any.

User response: Verify the existence or spelling of the storage group names by referring to SYSIBM.SYSSTOGROUP. Submit another request with the correct storage group designated.

Severity: 4 (warning)

DSNU647I   *csect-name* **UNAVAILABLE REASON X** *reason-code* **UNAVAILABLE TYPE X** *type-code*, **TABLESPACE NAME=** *table-space-name* **NOT FOUND**

Explanation: The indicated table space was not available for processing by the STOSPACE utility. The reason for its unavailability is given in 'reason-code'. The type of unavailability is given in 'type-code'.

This message may be issued when an inconsistency between SYSIBM.SYSTABLESPACEand SYSIBM.SYSTABLEPART has been detected. The inconsistency may be due to a table space being dropped during the processing of the STOSPACE utility.

System action: Processing continues to the next table space in the storage group, if any.

System programmer response: Determine if the 'reason-code' and 'type-code' require system corrective action.

User response: Use the 'reason-code' and 'type-code' to determine why the resource was unavailable. When the problem has been resolved, resubmit the STOSPACE request.

Problem determination: See the Problem Determination section for message DSNT500I.

Severity: 4 (warning)

DSNU648I   *csect-name* **UNAVAILABLE REASON X** *reason code*, **UNAVAILABLE TYPE X** *type-code*, **INDEX NAME=***index-name* **NOT FOUND**

Explanation: The indicated index was not available for processing by the STOSPACE utility. The reason for its unavailability is given in 'reason-code'. The type of unavailability is given in 'type-code'.

This message may be issued when an inconsistency between SYSIBM.SYSINDEXES and SYSIBM.SYSINDEXPART has been detected. The inconsistency may be due to an index being dropped during the processing of the STOSPACE utility.

System action: Processing continues to the next index in the storage group, if any.

System programmer response: Determine if the 'reason-code' and 'type-code' require system corrective action.

User response: Use the 'reason-code' and 'type-code' to determine why the resource was unavailable. When the problem has been resolved, resubmit the STOSPACE request.

Problem determination: See the Problem Determination section for message DSNT500I.

Severity: 4 (warning)

DSNU650I   *csect-name* *control-statement-text*

Explanation: Each control statement is returned to the invoker as a part of the normal output. If an error occurred while processing the statement, a separate error message follows this message.

System action: The processing of the utility request continues normally.

Severity: 0 (informational)

DSNU651I   *csect-name* **SET NOCOPYPEND OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET TABLESPACE NOCOPYPEND utility operation. The COPY PENDING restriction has been removed from the specified table space.

System action: The REPAIR SET statement has completed normally.

Severity: 0 (informational)

DSNU652I   *csect-name* **VERIFY OPERATION SUCCESSFUL**

Explanation: This message is issued during REPAIR utility processing in response to a VERIFY control statement. It indicates the verification requested was successful, permitting a subsequent REPLACE request to take effect.

System action: The requested verification was successful. If this is the only verification required for the subsequent REPLACE control statement, the REPLACE control statement takes effect. If there are other verification control statements, they must also complete successfully before the subsequent REPLACE control statement takes effect.

Severity: 0 (informational)

DSNU653I   *csect-name* **REPLACE OPERATION AT OFFSET** *hex-data* **WITH DATA** *hex-data* **FAILED. ATTEMPTED TO REPLACE CRITICAL NON-DATA INFORMATION ON THE PAGE** *page-num* **WITH INCORRECT INFORMATION.**

Explanation: The REPAIR utility job with the LOG YES option was unable to replace certain critical, nondata information in the page for one of these reasons:

- Some critical information on any page at the indicated offset cannot be replaced by this method.
- The page is a header page and some of the information at the indicated offset cannot be replaced by this method.
- The information can be replaced only by using the LOG NO option.

System action: REPAIR resumes executing at the next LOCATE statement. Later control statements in the same LOCATE block are not executed.

User response: Correct the REPAIR control statements and resubmit the request in a new utility job step.

User response: Check the OFFSET option on the REPLACE statement to be sure it does not cause critical information to be replaced. If the replacement is intentional, use the LOG NO option.

Severity: 8 (error)

DSNU654I   *csect-name* **VERIFY OPERATION FAILED, DATA IS** *hex-data*

Explanation: The verify operation requested on the control statement failed. *hex-data* contains the data value that failed the verify operation.

System action: All REPAIR REPLACE actions subject to the current LOCATE scope are ignored. Processing continues at the next LOCATE request, if any.

User response: Determine why the data (as reported in the message) does not match the expectations of VERIFY option. Make the appropriate correction, and resubmit the REPAIR request in a new utility job step.

Problem determination: The REPAIR DUMP option can help you determine the correct verification data.

Severity: 4 (warning)

Related concepts:

▷ Using VERIFY with REPLACE and DELETE operations (DB2 Utilities)

Related reference:

▷ REPAIR (DB2 Utilities)

▷ Syntax and options of the REPAIR control statement (DB2 Utilities)

DSNU655I   *csect-name* **DELETE OPERATION SUCCESSFUL**

Explanation: This message is issued during the REPAIR utility in response to a DELETE control statement.

System action: The data record specified on the previous LOCATE statement and all of its related INDEX entries have been deleted from the system. Processing continues.

Severity: 0 (informational)

DSNU656I   *csect-name* **REPLACE OPERATION SUCCESSFUL, DATA WAS** *hex-data*

Explanation: This message is issued by the REPAIR utility in response to a REPLACE control statement request. The message indicates that replacement of the data occurred and that the replaced data had the indicated hexadecimal value ('hex-data').

System action: The REPAIR utility located the indicated record, performed all requested verifications, and replaced the data in the record at the indicated offset with the new data as defined by the REPLACE control statement.

System programmer response: If a changed record is recovered before an image copy is made, and the REPAIR utility was run with LOG NO, the repair must be reapplied. If the REPAIR utility was run with LOG YES, no image copy is required before recovering the data after the repair.

Severity: 0 (informational)

DSNU657I   *csect-name* **REPLACE RESET OPERATION SUCCESSFUL**

Explanation: This message is issued by the REPAIR utility in response to a REPLACE RESET control statement request. The message indicates that the broken page indicator has been reset.

System action: The REPAIR utility has reset the broken page indicator for the specified page. Although the reset indicator has been committed, it has not been logged. If the page is recovered before an image copy is made, the REPAIR RESET must be reapplied.

However, to make the repair persist across subsequent RECOVER operations, an image copy should be taken subsequent to the completion of the REPAIR utility.

Severity: 0 (informational)

DSNU658I   *csect-name* **MULTIPLE RECORDS FOUND WITH SPECIFIED KEY**

Explanation: The specified key is not unique. Several DSNU660I messages, giving possible RID values for this KEY, follow this message.

System action: Processing continues normally.

User response: Change the LOCATE statement to locate the correct RID, and resubmit the request in a new utility job step.

Severity: 8 (error)

DSNU659I   *csect-name* **REPAIR LEVELID OPERATION DISALLOWED.**

Explanation: The REPAIR utility with the LEVELID clause is not allowed. Possible reasons are:
- The data set was in use at the time the REPAIR utility was run.
- The page set has outstanding log apply activity (an indoubt UR has interest in the page set or the page set has pages in the logical page list (LPL)).

System action: Processing ends with this operation.

User response: The REPAIR LEVELID operation for the given index space or table space cannot continue at this stage. If the page set is in use, you should not attempt to run REPAIR LEVELID. If the page set has log apply activity outstanding (indoubt UR or pages in LPL), the log apply activity must complete before using REPAIR LEVELID. Submit other REPAIR control statements in a different job step.

Severity: 8 (error)

DSNU660I   *csect-name* **POSSIBLE RID** *rid*

Explanation: 'rid' is one of several records that map to the specified KEY.

System action: Other records also mapping to the specified KEY are identified by other DSNU660I messages. This message is preceded by message DSNU658I. Processing continues normally.

User response: Change the LOCATE statement to locate the correct RID, and resubmit the request in a new utility job step. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU661I   *csect-name* **RECORD NOT FOUND IN PAGE, MAXID=** *hex-data*

Explanation: A RID was either specified or obtained from the clustering index via the specified KEY. The PAGE indicated in the RID did not contain the RECORD indicated in the RID.

System action: Processing proceeds to the next LOCATE statement, if there is one.

User response: Either change the LOCATE statement to locate a valid RID or KEY, or use LOCATE PAGE and resubmit the REPAIR statements in a new utility job step.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU662I   *csect-name* **NO RECORD FOUND WITH SPECIFIED KEY**

Explanation: A KEY was specified on the LOCATE statement, but no entry exists in the clustering index for that KEY.

System action: Processing begins at the next LOCATE statement, if there is one.

User response: Correct the LOCATE statement, and resubmit the REPAIR statements in a new utility job step.

Severity: 8 (error)

DSNU663I   *csect-name* **PAGE** *hex-page-number* **NOT FOUND**

Explanation: The page identified a record identifier (RID) or KEY does not exist in the specified table space.

System action: Processing begins at the next LOCATE statement, if there is one.

User response: Correct the LOCATE statement, and resubmit the REPAIR statements in a new utility job step.

Severity: 0 (informational) when issued by DUMP PAGES(n). 8 (error) when issued by LOCATE PAGE(n).

DSNU664I   *csect-name* **INVALID OFFSET OR LENGTH FOR** *xxxxx*

Explanation: 'xxxxx' is 'RECORD' or 'PAGE'. The specified or default values for OFFSET and LENGTH combine to extend beyond the specified record or page.

System action: Processing continues at the next LOCATE statement, if there is one.

User response: Correct the OFFSET and/or LENGTH specification, and resubmit the REPAIR statements in a new utility job step. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 4 (warning)

DSNU665I   *csect-name* **NO RECORD ASSOCIATED WITH SPECIFIED RID**

Explanation: A record identifier (RID) was specified on the LOCATE statement, but that RID is not associated with any record in the specified table space. If DELETE processing returns this message, the RID may exist but be invalid for DELETE.

DSNU666I • DSNU670I

System action: Processing begins at the next LOCATE statement, if there is one. Subsequent utilities in the same job step will not be executed.

User response: Correct the LOCATE statement, and resubmit the REPAIR statements in a new utility job step.

Problem determination: REPAIR DELETE will issue this message for RIDs which are internal and do not directly represent user data. The DUMP keyword may be used to view the RID, if it exists. These internal records may only be repaired using the REPLACE keyword.

Severity: 8 (error)

DSNU666I   *csect-name* **NO INDEX EXISTS**

Explanation: The KEY for this table space is not a full clustering key. The statement is invalid.

System action: Processing continues at the next LOCATE statement, if there is one.

User response: Correct the LOCATE statement to use the clustering index, or change LOCATE by PAGE or RID.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU668I   *csect-name* **UNABLE TO CONVERT KEY TO** *format*

Explanation: The value of *format* specifies the data type to which the KEY specification must be converted. The conversion has failed.

System action: Processing begins at the next LOCATE statement, if there is one.

User response: Correct the LOCATE statement to specify a KEY either of the indicated data type or of a compatible data-type. Then resubmit the REPAIR statements in a new utility job step.

Severity: 8 (error)

Related reference:

⇨ REPAIR (DB2 Utilities)

⇨ Syntax and options of the REPAIR control statement (DB2 Utilities)

DSNU669I   *csect-name* **INDEX NOT FOUND FOR SPECIFIED TABLESPACE**

Explanation: An index was specified on the LOCATE KEY statement via the INDEX keyword, but no index by that name was found related to the specified TABLESPACE.

System action: Processing begins at the next LOCATE statement, if there is one.

User response: Correct the LOCATE statement and resubmit the REPAIR statements in a new utility job step.

Problem determination: Valid index names can be found in the SYSIBM.SYSINDEXES catalog table.

Severity: 8 (error)

DSNU670I   *csect-name* **PAGE X** *page-id* **IS A BROKEN PAGE**

Explanation: The indicated page located by the REPAIR utility has been marked broken by the data manager. The 'page id' is given in hexadecimal. This page is not accessible until the broken page indicator is reset by the REPAIR REPLACE RESET option.

System action: The message reports that the page in which the REPAIR utility is requested to REPLACE some data is broken.

System programmer response: Determine the cause of the broken page. It could be an internal error and require support center assistance. The REPAIR DUMP option can be used to determine the contents of the page for failure analysis. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Determine the cause of the broken page with the assistance of the system programmer. When it is repaired, use the REPAIR RESET option.

Severity: 4 (warning)

| **DSNU671I**   *csect-name* **- DBID=X'***dbid***' PSID=X'***psid***' OBID=X'***obid***'** *mismatch-information-type* **IN THE CATALOG DOES NOT MATCH THE PAGE SET**

| **Explanation:** The REPAIR utility with the CATALOG or VERSIONS option detected that information in the catalog table is different from the information in the page set. In this case, no value was available for the page set.

| This problem usually occurs after copying data by using DSN1COPY.

| *csect-name*
| The name of the control section that issued the message.

| *dbid*   The database identifier (DBID) for the database in which mismatched information was detected.

| *psid*   The page set identifier (PSID) for the table space for which mismatched information was detected.

| *obid*   The object identifier (OBID) for the table for which mismatched information was detected.

| *mismatch-information-type*
| The type of information that did not match. *mismatch-information-type* can have one of the following values:

| **DBID**   Database identifier

| **PSID**   Page set identifier

| **OBID**   Object identifier

| **TABLE SCHEMA**
| Schema definition of the table

| **System action:** The utility cannot continue processing and returns completion code 8.

| **User response:** Find the DSN1COPY job that copied the affected data, and check the following items:
| • Ensure that the SYSXLAT DD statement contains the correct IDs for each object to be translated.
| • Ensure that the correct object name is specified.
| • Ensure that the target and source table have matching columns.

| If you made any changes, rerun DSN1COPY.
| **Related reference**:
| ➥ DSN1COPY (DB2 Utilities)
| ➥ Syntax and options of the REPAIR control statement (DB2 Utilities)

DSNU672I   *csect-name* **INDEX USED HAS MULTIPLE-FIELD KEY**

Explanation: A REPAIR LOCATE KEY operation was attempted against a table space using an index that has a multiple-field key. LOCATE KEY does not support multiple-field keys.

System action: Processing begins at the next LOCATE scope, if any.

User response: Change the LOCATE statement to specify PAGE and RID, and resubmit the REPAIR statements in a new utility job step.

Severity: 8 (error)

DSNU673I   *csect-name* **REPAIR LOCATE RID LOCATED A NON-DATA PAGE**

Explanation: The LOCATE RID option of the REPAIR utility was used to process a nondata page.

System action: Processing continues at the next LOCATE statement. Otherwise, processing stops.

User response: To process nondata pages, use the LOCATE PAGE option in the REPAIR control statements and resubmit the new utility job step. If you suspect an error in DB2, you might need to report the problem. For

DSNU674I

information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

| **DSNU674I** | *csect-name* **-** *mismatch-information-type* **FOR DBID=X'***dbid***' PSID=X'***psid***' IS** *catalog-format-type* **IN THE DB2 CATALOG BUT IS** *pageset-format-type* **IN THE PAGE SET** |

Explanation: The REPAIR utility with the CATALOG or VERSIONS option detected that information in the catalog table is different from the data from the page set.

This problem usually occurs after copying data by using DSN1COPY.

csect-name
> The name of the control section that issued the message.

mismatch-information-type
> The type of information that did not match. *mismatch-information-type* can have one of the following values:
> - ROW FORMAT
> - RBA FORMAT
> - TABLESPACE TYPE

dbid   The database identifier (DBID) for the database in which mismatched information was detected.

psid   The page set identifier (PSID) for the page set for which mismatched information was detected.

catalog-format-type
> The value in the catalog for *mismatch-information-type*.
>
> *catalog-format-type* is type CHAR and has a maximum of eight characters. Possible values depend on the value of *mismatch-information-type* as follows:

Value of *mismatch-information-type*	**Possible values of** *catalog-format-type*

ROW FORMAT	**BRF**
	    Basic row format
	
	RRF
	    Reorder row format

RBA FORMAT	**BASIC**
	    6-byte RBA format
	
	EXPANDED
	    10-byte RBA format

Value of *mismatch-information-type*	Possible values of *catalog-format-type*
TABLESPACE TYPE	**PART** Table space that is partitioned (non-universal) **PBR** Range-partitioned universal table space **PBG** Partition-by-growth universal table spaces **SEG** Segmented table space (non-universal) **NON-SEGMENTED** Non-segmented or simple table space. **MC** Universal table space with the MEMBER CLUSTER attribute. **NOT MC** Universal table space without the MEMBER CLUSTER attribute **PBG Hash** Partition-by-growth table space that is organized for hash access **PBR Hash** Range-partitioned table space that is organized for hash access

pageset-format-type
>   The value from the page set for *mismatch-information-type*.

>   *pageset-format-type* is type CHAR and has a maximum of eight characters. Possible values are the same as *catalog-format-type*.

System action: If *mismatch-information-type* is TABLESPACE TYPE, the utility cannot continue processing. REPAIR returns completion code 8.

If *mismatch-information-type* is ROW FORMAT or RBA FORMAT, one of the following actions occurs:

* If the TEST keyword was not specified in the REPAIR statement, the utility changes the catalog value to match the data and returns completion code 0.
* If the TEST keyword was specified in the REPAIR statement, the utility does not change the catalog value and returns completion code 4.

User response: If the utility completion code is 8, find the DSN1COPY job that might have caused this error. Ensure that the correct object name is specified and that the target and the source have the same table space type. If you made any changes, rerun DSN1COPY.

If the utility completion code is 4, rerun the REPAIR utility without the TEST keyword to fix the catalog.

Related concepts:

Types of DB2 table spaces (Introduction to DB2 for z/OS)

Row format conversion for table spaces (DB2 Administration Guide)

Related reference:

DSN1COPY (DB2 Utilities)

Syntax and options of the REPAIR control statement (DB2 Utilities)

| **DSNU675I** *csect-name* - *mismatch-information-type* **FOR DBID=X'***dbid***' PSID=X'***psid***' IS** *catalog-value* **IN THE DB2 CATALOG BUT IS** *pageset-value* **IN THE PAGE SET.**

Explanation: The REPAIR utility with the CATALOG or VERSIONS option detected that information in the catalog table is different from the data from the page set.

This problem usually occurs after copying data by using DSN1COPY.

csect-name
> The name of the control section that issued the message.

mismatch-information-type
> The type of information that did not match. *mismatch-information-type* can have one of the following values:

> **LOW VERSION**
>> The oldest version number for the table space.

> **HIGH VERSION**
>> The current version number for the table space.

> **PBG HASHVALUE**
>> The hash value for a partition-by-growth universal table space.

> **PBR HASHVALUE**
>> The hash value for a range-partitioned universal table space.

> **SEGMENT SIZE**
>> The table space segment size.

> **PAGE SIZE**
>> The size of the data page.

dbid   The database identifier (DBID) for the database in which mismatched information was detected.

psid   The page set identifier (PSID) for the page set for which mismatched information was detected.

catalog-value
> The integer value in the catalog for *mismatch-information-type*.

pageset-value
> The integer value from the page set for *mismatch-information-type*.

System action: If the value of *mismatch-information-type* is SEGMENT SIZE or PAGE SIZE, the utility cannot continue processing. REPAIR returns completion code 8.

If the value of *mismatch-information-type* is LOW VERSION, HIGH VERSION, PBG HASHVALUE, or PBR HASHVALUE, one of the following actions occur:

• If the TEST keyword was not specified in the REPAIR statement, the utility changes the catalog value to match the data and returns completion code 0.

• If the TEST keyword was specified in the REPAIR statement, the utility does not change the catalog value and returns completion code 4.

User response: If the utility completion code is 8, find the DSN1COPY job that might have caused this error. Ensure that the correct object name is specified and that the target and the source have the same segment size and page size. If you made any changes, rerun DSN1COPY.

If the utility completion code is 4, rerun the REPAIR utility without the TEST keyword to fix the catalog.

Related reference:

↪ DSN1COPY (DB2 Utilities)

↪ Syntax and options of the REPAIR control statement (DB2 Utilities)

DSNU676I *csect-name* - **ROWID=***rowid* **VERSION=***version* **IS NOT FOUND IN AUXILIARY INDEX**

Explanation: The REPAIR utility was unable to locate the ROWID *rowid* and VERSION *version* in the index on the auxiliary table.

System action: Utility returns completion code=8.

Programmer response: Correct the ROWID or VERSION. Only ROWID and VERSION values previously generated by DB2 can be used. The most likely source values of a ROWID and VERSION are messages DSNU743I, DSNU805I, and DSNU806I from the CHECK DATA or CHECK LOB utilities.

DSNU677I   *csect-name*-DELETE OPERATION CANNOT DELETE RELATED INDEX ENTRIES OR OTHER ASSOCIATED OBJECTS

Explanation: A REPAIR utility control statement specified a DELETE operation for a row in a table that has indexes, LOB columns, XML columns, or referential integrity constraints. REPAIR could delete only the table row.

csect-name
>   The name of the control section that issued the message.

System action: The data record that is specified in the LOCATE block is deleted. However, any related index entries or associated objects are not deleted.

System programmer response: Run the appropriate CHECK utilities to ensure that the data is consistent with the other associated objects and follow the recommended actions.

Severity: 4 (warning)

Related reference:

▶ Syntax and options of the REPAIR control statement (DB2 Utilities)

▶ CHECK INDEX (DB2 Utilities)

▶ CHECK DATA (DB2 Utilities)

▶ CHECK LOB (DB2 Utilities)

DSNU678I   *csect-name* VERIFY, REPLACE, DUMP, OR DELETE BEFORE LOCATE

Explanation: The REPAIR utility was requested to VERIFY, REPLACE, or DUMP or DELETE a page or record and that page of record could not be located.

System action: Processing begins at the next LOCATE statement, if there is one. Subsequent utilities in the same job step will not be executed.

User response: Correct the REPAIR control statements, and resubmit the request in a new utility job step. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU679I   *csect-name* MULTIPLE 'DELETE'S PER 'LOCATE' NOT ALLOWED

Explanation: The REPAIR utility was requested to DELETE the same record more than once. Only one DELETE is allowed per LOCATE RID or LOCATE KEY.

System action: Processing begins at the next LOCATE scope, if any. Subsequent utilities in the same job step are not executed.

User response: Remove all but one DELETE control statement from each LOCATE scope which is intended to delete a record. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSNU680I   *csect-name* PAGE *hex-page-number*

Explanation: This message is issued as an identifier at the top of each page dumped by the REPAIR DUMP PAGES facility. 'hex-page-number' identifies the page which is about to be dumped.

System action: Processing continues.

Severity: 0 (informational)

DSNU681I   *csect-name* **SET NOCHECKPEND OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET TABLESPACE NOCHECKPEND utility operation. The check pending restriction has been removed from the specified table space.

System action: The REPAIR SET statement has completed normally.

Severity: 0(informational)

DSNU682I   *csect-name* **SET NORCVRPEND OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET NORCVRPEND utility operation. The RECOVERY PENDING restriction has been removed from the specified table space, from the specified index, or from all indexes of the specified table space.

System action: The REPAIR SET statement has completed normally.

Severity: 0 (informational)

DSNU683I   *csect-name* **REPAIR LEVELID OPERATION SUCCESSFUL.**

Explanation: The REPAIR utility with the LEVELID clause completed successfully.

System action: Processing completed successfully.

User response: Submit any other REPAIR control statements in a different job step.

Severity: 0 (informational)

DSNU684I   *csect-name* **- SET NOAUXCHKP OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET NOAUXCHKP utility operation. The auxiliary check pending (ACHKP) state has been removed for the specified base table space.

System action: Utility returns completion code=0.

DSNU685I   *csect-name* **SET NORBDPEND OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET NORBDPEND utility operation. The three restrictions REBUILD PENDING, REBUILD PENDING STAR and PAGESET REBUILD PENDING have been removed from the specified index or from all indexes of the specified table space.

System action: The REPAIR SET statement has completed normally.

Severity: 0 (informational)

DSNU686I   *csect-name* **- SET NOAUXWARN OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET NOAUXWARN utility operation. The AUX WARNING state has been removed for the specified table space.

System action: Utility returns completion code=0.

Programmer response: None.

Severity: 0 (informational)

DSNU687I   *csect-name* **- ROWID=***rowid* **VERSION=***version* **LOB DELETED**

Explanation: The LOB with the ROWID *rowid* and VERSION *version* was successfully deleted from the specified LOB table space.

System action: Utility returns completion code=0.

Programmer response: None

DSNU688I   *csect-name* **SET NOARBDPEND OPERATION SUCCESSFUL**

Explanation: The REPAIR utility successfully reset an object that was in an advisory rebuild pending (ARBD) state.

System action: No action required.

User response: No response required.

Severity: 0 (informational)

DSNU689I   *csect-name* **SET NOAREORPENDSTART OPERATION SUCCESSFUL**

Explanation: The REPAIR utility successfully reset an object that was in an advisory reorg pending (AREO*) + state.

System action: No action required.

User response: No response required.

Severity: 0 (informational)

DSNU690I   *csect-name* **- CAN NOT UPDATE INFORMATION IN THE CATALOG THAT DOES NOT MATCH THE PAGE SET**

Explanation: The REPAIR utility with the CATALOG or VERSIONS option detected an error in the catalog and page set information and cannot continue processing. The error is described in another DSNU message.

csect-name
    The name of the control section that issued the message.

System action: REPAIR cannot continue processing and returns completion code 8.

User response: Correct the problem that is reported in one of the following DSNU messages:
* DSNU671I
* DSNU674I
* DSNU675I

Related reference:

⮕ Syntax and options of the REPAIR control statement (DB2 Utilities)

Related information:
"DSNU671I" on page 771
"DSNU674I" on page 772
"DSNU675I" on page 774

DSNU691I   *csect-name* **- LOB WITH ROWID=**rowid **VERSION=**version **NOT FOUND**

Explanation: The REPAIR utility was unable to find a LOB with the ROWID of *rowid* and VERSION *version*.

System action: Utility returns completion code=8.

Programmer response: Either the ROWID is incorrect, or the index on the auxiliary table is broken. In the first case correct the ROWID as specified on the **LOCATE ROWID** statement. If the index on the auxiliary table is broken, execute the REBUILD INDEX utility to rebuild the index.

DSNU692I   *csect-name* **A RECOMMENDED REORG HAS NOT BEEN DONE**

Explanation: One of the following situations occurred:
* The REPAIR utility with the VERSIONS or CATALOG option detected that REORG was not run on the source object after the last alter operation that created a new version. Run REORG before running DSN1COPY on the object.
* REORG was not run on all partitions in the table space. For partitioned objects, run REORG on all partitions before running DSN1COPY on the object.

csect-name
    The name of the control section that issued the message.

System action: The utility continues processing. Objects are still usable.

User response: Consider running REORG on the object.

If you are moving objects from one subsystem to another, follow the steps in Updating version information when moving objects to another subsystem (DB2 Utilities).

Severity: 4 (warning)

Related reference:

⯈ DSN1COPY (DB2 Utilities)

⯈ Syntax and options of the REPAIR control statement (DB2 Utilities)

⯈ REORG TABLESPACE (DB2 Utilities)

⯈ REORG INDEX (DB2 Utilities)

DSNU694I   *csect-name utility-name* **CANNOT CONTINUE BECAUSE NOT ENOUGH INFORMATION IS AVAILABLE**

Explanation: The REPAIR utility does not have enough information to update the DB2 catalog for one of the following reasons:

- The utility is unable to find a V0 OBDREC in the system page.
- A system page does not exist.
- The utility is unable to find any information in the page set.

csect-name
      The name of the control section that issued the message.

utility-name
      The name of the utility. The name can be either REPAIR VERSIONS or REPAIR CATALOG.

System action: The utility terminates. The catalog is not updated.

User response: Perform at least one INSERT or UPDATE operation on one of the partitions, or reorganize the table space. Then use the DSN1COPY utility to copy the physical file from the source and replace the target. If you receive message DSNU694I again, call IBM Software Support.

Severity: 8 (error)

Related reference:

⯈ DSN1COPY (DB2 Utilities)

⯈ REORG TABLESPACE (DB2 Utilities)

⯈ Syntax and options of the REPAIR control statement (DB2 Utilities)

⯈ INSERT (DB2 SQL)

⯈ UPDATE (DB2 SQL)

DSNU695I   *csect-name* **- INFORMATION IN THE CATALOG WAS UPDATED TO MATCH THE PAGE SET**

Explanation: The REPAIR utility with the CATALOG or VERSIONS option fixed the mismatched information that was identified in one of the following messages:

- DSNU671I
- DSNU674I
- DSNU675I

The utility fixed the mismatched information by changing the value in the catalog table and directory to match the information in the page set.

csect-name
      The name of the control section that issued the message.

System action: The utility returns completion code 0.

User response: No action is required.

Related reference:

➡ Syntax and options of the REPAIR control statement (DB2 Utilities)

Related information:

"DSNU671I" on page 771

"DSNU674I" on page 772

"DSNU675I" on page 774

DSNU696I *csectname* **- GDGLIMIT COULD NOT BE DETERMINED FOR GDG** *gdg-name* **FOR TABLESPACE** *dbname.tsname* **RETURN CODE** *returncode* **REASON CODE** *reasoncode*

Explanation: An error was returned from the Catalog Search Interface (IGGCSI00) while trying to determine the GDGLIMIT for the data set shown. Cleanup for the table space shown was not performed.

System action: Utility processing continues.

System programmer response: Correct the problem with the GDG definition and resubmit the utility. If the problem cannot be corrected, this error can be bypassed by using RETAIN LAST(n) syntax and omitting the GDGLIMIT keyword.

Severity: 4 (warning)

DSNU697I *csectname* **- CLEANUP POINT** *date* **HAS BEEN DETERMINED**

Explanation: The MODIFY utility has generated the date shown as the cleanup point. The cleanup point is used to start the deletion of records in SYSCOPY and SYSLGRNX.

System action: Utility processing continues normally.

System programmer response: No response required.

Severity: 0 (informational)

DSNU698I *csect-name* **- BSDS HAS BEEN USED TO DETERMINE THE CLEANUP POINT**

Explanation: The MODIFY utility has obtained the oldest archive log information from the BSDS data set. Automatic cleanup of SYSCOPY and SYSLGRNX records has been performed based on this information. Records older than the oldest retained archive log record have been deleted.

System action: Utility processing continues normally.

User response: No response required.

Severity: 0 (informational)

DSNU699I *csect-name* **- GDG LIMIT** *limit* **HAS BEEN USED TO DETERMINE CLEANUP POINT**

Explanation: The MODIFY utility has obtained the Generation Data Group (GDG) limit from the GDG base. Automatic cleanup of SYSCOPY and SYSLGRNX records has been performed based on this information. Records that refer to GDG versions which have been reused more recently are deleted.

System action: Utility processing continues normally.

User response: No response required.

Severity: 0 (informational)

DSNU700I *csect-name nnnn* **INDEX ENTRIES UNLOADED FROM INDEX=***iiii* **PARTITION=***pppp*

Explanation: The CHECK utility UNLOAD phase unloaded nnnn records from the index named iiii and partition pppp.

System action: Processing continues.

DSNU701I • DSNU704I

Severity: 0 (informational)

DSNU701I   *csect-name nnnn* **INDEX ENTRIES UNLOADED FROM** *iiii*

Explanation: The REORG utility or the UNLOAD phase of the CHECK utility unloaded *nnnn* records from the index named *iiii*.

System action: Processing continues.

Severity: 0 (informational)

DSNU702I   *csect-name* **INVALID INDEX SPECIFICATION. INDEX=***iiii*

Explanation: All tables referenced by the INDEXES in the CHECK or REBUILD utility must be in the same table space. Index 'iiii' is not over a table in the table space specified with the TABLESPACE keyword.

System action: Utility processing is discontinued.

User response: Correct the index name and/or the TABLESPACE parameters such that a single table space is involved, and rerun the utility job.

Problem determination: To CHECK or REBUILD ALL INDEXES in a table space, use the TABLESPACE keyword, but do not use the list option. This is one way to correct this error. If you want to correct the list of indexes specified in the index name list, the following procedure will locate the table space name related to the index 'iiii'.

The index name 'iiii' reported in the message is of the form 'creator.name'. Obtain the TBCREATOR and TBNAME columns from the DB2 catalog table SYSIBM.SYSINDEXES where columns CREATOR and NAME have the values reported in the message. Obtain the DBNAME and TSNAME columns from the DB2 catalog table SYSIBM.SYSTABLES, where columns CREATOR and NAME have the values obtained from TBCREATOR and TBNAME of SYSIBM.SYSINDEXES.

The values for DBNAME and TSNAME must be the same for all indexes specified in the index name list and must be the same as the table space name specified on the TABLESPACE keyword (if provided).

Severity: 8 (error)

DSNU703I   *csect-name* **INVALID INDEX SPECIFICATION. INDEX=***iiii*

Explanation: All tables referenced by the indexes in the CHECK or REBUILD utility must be in the same table space. Index 'iiii' is not over a table in the same table space as the prior indexes listed in the index name list.

System action: Utility processing is discontinued.

User response: Correct the index name and/or the TABLESPACE parameters such that a single table space is involved, and rerun the utility job.

Problem determination: To CHECK or REBUILD ALL INDEXES in a table space, use the TABLESPACE keyword, but do not use the list option. This is one way to correct this error. If you want to correct the list of indexes specified in the index name list, the following procedure will locate the table space name related to the index 'iiii'.

The index name 'iiii' reported in the message is of the form 'creator.name'. Obtain the TBCREATOR and TBNAME columns from the DB2 catalog table SYSIBM.SYSINDEXES, where columns CREATOR and NAME have the values reported in the message. Obtain the DBNAME and TSNAME columns from the DB2 catalog table SYSIBM.SYSTABLES, where columns CREATOR and NAME have the values obtained from TBCREATOR and TBNAME of SYSIBM.SYSINDEXES.

The values for DBNAME and TSNAME must be the same for all indexes specified in the index name list and must be the same as the table space name specified on the TABLESPACE keyword if provided.

Severity: 8 (error)

DSNU704I   *csect-name* **INVALID INDEX NON-LEAF PAGE. INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found an invalid non-leaf page while traversing the specified index to find the specified key.

csect-name
> The name of the control section that issued the message.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues. The utility completes with return code 4.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➥ CHECK INDEX (DB2 Utilities)

➥ REBUILD INDEX (DB2 Utilities)

DSNU705I   *csect-name* **UNLOAD PHASE COMPLETE, ELAPSED TIME=***tttt*

Explanation: The unload phase of the CHECK or REBUILD INDEX utility is complete.

System action: Processing continues.

Severity: 0 (informational)

DSNU706I   *csect-name* **ROW IS IN WRONG PARTITION RID=***rrrr***, INDEX=***iiii***, KEY=***kkkk***, CORRECT PART=***pppp***, WRONG PART=***wwww*

Explanation: The row at RID 'rrrr' resides in the wrong partition. The partitioning KEY 'kkkk' for partitioning INDEX 'iiii' indicates that the row at RID 'rrrr' should reside in partition 'pppp', however the rid indicates that the row contains partition 'wwww'.

System action: Utility processing continues.

User response: Run RECOVER TABLESPACE utility to recover the data or run REPAIR DELETE utility to delete the row 'rrrr' followed by CHECK INDEX and CHECK DATA utility if referential constraints are involved.

Problem determination: The inconsistency between the table space partition number of a row and the index space partition number of the row's clustering key may be due to a DB2 internal error, but is most likely due to a user using the DSN1COPY service aid to populate a target table space with data from a source table space when the clustering key values on the source and target table spaces are not identical.

Severity: 4 (warning)

DSNU707I   *csect-name* **INDEX ENTRY POINTS TO WRONG TABLE. OBID=***oooo***, RID=***rrrr***, INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found that an index contains an entry for a key that points to a record at the specified RID. However, the record belongs to the table with the specified OBID.

csect-name
          The name of the control section that issued the message.

oooo   The object identifier (OBID).

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

CHECK INDEX (DB2 Utilities)

REBUILD INDEX (DB2 Utilities)

DSNU708I   *csect-name* **INDEX IS NOT UNIQUE. INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found that a unique index contains a non-unique key.

csect-name
> The name of the control section that issued the message.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Drop the unique index and delete the duplicate rows. Re-create the index.

Severity: 4 (warning)
Related reference:

CHECK INDEX (DB2 Utilities)

DSNU709I   *csect-name* **NO INDEX ENTRY FOR ROW. RID=***rrrr***, INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found that an index is missing an entry for the row at the specified RID with the specified key.

csect-name
> The name of the control section that issued the message.

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues. The utility completes with return code 4.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)
Related reference:

CHECK INDEX (DB2 Utilities)

REBUILD INDEX (DB2 Utilities)

DSNU710I   *csect-name* **INDEX ENTRY CONTAINS INVALID ZERO REPRESENTATION. RID=***rrrr***, INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found an index that has an entry that contains a negative zero. The index entry is for the row at the specified RID with the specified key.

csect-name
> The name of the control section that issued the message.

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➡ CHECK INDEX (DB2 Utilities)

➡ REBUILD INDEX (DB2 Utilities)

DSNU711I   *csect-name* **NO ROW FOR INDEX ENTRY. RID=**rrrr**, INDEX=**iiii**, KEY=**kkkk

Explanation: The CHECK INDEX utility found an index that contains an entry that has no row. The index entry contained the specified RID and key, but the RID is not a valid record.

csect-name
> The name of the control section that issued the message.

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➡ CHECK INDEX (DB2 Utilities)

➡ REBUILD INDEX (DB2 Utilities)

DSNU712I   *csect-name* **INDEX OUT OF SEQUENCE. INDEX=**iiii**, KEY=**kkkk

Explanation: The CHECK INDEX utility found that an index is not in the correct sequence.

csect-name
> The name of the control section that issued the message.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➡ CHECK INDEX (DB2 Utilities)

➡ REBUILD INDEX (DB2 Utilities)

DSNU713I   *csect-name* **KEYS MISMATCH. RID=**rrrr**, INDEX=**iiii**, KEY=**kkkk**, DATA KEY=**jjjj

Explanation: The CHECK INDEX utility found an index that contains an entry for the row at the specified rid that is inconsistent. The specified key does not match the specified data key.

csect-name
> The name of the control section that issued the message.

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

jjjj   The data key.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➡ CHECK INDEX (DB2 Utilities)

➡ REBUILD INDEX (DB2 Utilities)

DSNU714I   *csect-name* **CHECK DATA CANNOT GENERATE A REPAIR STATEMENT FOR A COMPRESSED ROW OF TABLE** *table-name*

Explanation: The CHECK DATA utility detected an inconsistency on a compressed row in the specified table. Although the SHRLEVEL CHANGE option and one of the following options was specified, a REPAIR statement cannot be generated for a compressed row:

• AUXERROR INVALIDATE
• LOBERROR INVALIDATE
• XMLERROR INVALIDATE

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table in which the inconsistency was detected.

System action: CHECK DATA continues processing.

System programmer response: See a previous message to determine the inconsistency.

Consider modifying the CHECK DATA utility control statement to use SHRLEVEL REFERENCE or the REPORT option of AUXERROR, LOBERROR, or XMLERROR.

Severity: 4 (warning)

Related reference:

➡ Syntax and options of the CHECK DATA control statement (DB2 Utilities)

DSNU715I   *csect-name* **MULTIPLE INDEX ENTRIES FOR ROW. RID=***rrrr***, INDEX=***iiii***, KEY=***kkkk*

Explanation: The CHECK INDEX utility found an index that has multiple index entries for the row at the specified RID with the specified key.

csect-name
> The name of the control section that issued the message.

rrrr   The record ID (RID) in hexadecimal format.

iiii   The index name.

kkkk   The hexadecimal representation of the key value.

System action: CHECK INDEX utility processing continues.

User response: Run the REBUILD INDEX utility for index *iiii*.

Severity: 4 (warning)

Related reference:

➡ CHECK INDEX (DB2 Utilities)

➡ REBUILD INDEX (DB2 Utilities)

DSNU717I   *csect-name nnnn* **ENTRIES CHECKED FOR INDEX=***iiii* **PARTITION=***pppp*

Explanation: The CHECK utility completed checking index iiii partition pppp and processed nnnn index entries. This message is issued once for each index partition checked.

System action: Processing continues.

Severity: 0 (informational)

DSNU718I   *csect-name* **NO INDEXES FOUND FOR TABLESPACE** *tttt*

Explanation: A utility has been invoked with the TABLESPACE keyword specifying a table space containing tables that have no indexes.

System action: This message indicates completion of the utility. Processing continues to the next utility statement in the job step, if any. If there are no utility statements following, the job step is terminated normally and the UTILID of the job step becomes available for reuse.

User response: If the table space specification is in error, correct it and resubmit the utility. If the table space specification is correct, and the table space contains no indexed tables, this is a normal indication and no response is necessary.

Severity: 4 (warning)

DSNU719I   *csect-name nnnn* **ENTRIES CHECKED FOR INDEX** *iiii*

Explanation: The CHECK utility has completed checking index 'iiii' and has processed 'nnnn' index entries. This message will be issued once for each index checked.

System action: Processing continues.

Severity: 0 (informational)

DSNU720I   *csect-name* **SORTCHK PHASE COMPLETE, ELAPSED TIME = ** *tttt*

Explanation: The CHECK utility SORTCHK phase is complete.

System action: Processing continues.

Severity: 0 (informational)

DSNU721I   *csect-name* **- USER** *auth-id* **DOES NOT HAVE DELETE AUTHORITY ON TABLE**

Explanation: The CHECK DATA utility was run with DELETE YES, but user *auth-id* does not have authority to delete from a table specified in the utility statement or from one of its descendents.

System action: Further utility processing is terminated.

User response: Obtain authorization to delete from the tables specified in the utility statement and any of its descendents that are affected.

Severity: 8 (error)

DSNU722I   *csect-name* **- USER** *userid* **DOES NOT HAVE INSERT AUTHORITY ON EXCEPTION TABLE**

Explanation: The CHECK DATA utility was run with FOR EXCEPTION specified but user 'userid' does not have authority to insert into an exception table specified in the utility statement.

System action: Further utility processing is terminated.

User response: Obtain authorization to insert into all of the exception tables specified in the utility statement.

Severity: 8 (error)

DSNU725I　*csect-name* **DUPLICATE TABLESPACE** *tsp* **NAMES EXIST IN THE TABLESPACE LIST**

Explanation:　This message is issued by the CHECK DATA utility when a duplicate table space 'tsp' is found in the table space list. An entry in the list for the same table space with a partition specified is also considered a duplicate. This is because all partitions are the default when no partition is specified for a partitioned table space. The CHECK DATA utility proceeds and the duplicate entry is skipped.

System action:　Processing continues.

Severity:　4 (warning)

DSNU726I　*csect-name* **SCOPE PENDING IS ACTIVE, BUT THE ENTIRE TABLE SPACE** *tsp* **MUST BE CHECKED**

Explanation:　This message is issued by the CHECK DATA utility when checking with the option SCOPE PENDING. Table space *tsp* is in check pending status, but the utility is checking all rows in dependent tables or tables defined with check constraints in that table space.

System action:　The utility continues processing.

Severity:　4 (warning)

DSNU727I　*csect-name* **TABLESPACE** *tsp* **IS NOT CHECK PENDING**

Explanation:　This message is issued when 'CHECK DATA TABLESPACE tsp SCOPE PENDING' is specified and the CHECK DATA utility finds that the table space is not in a check pending state. The CHECK DATA utility skips table space 'tsp' and will continue with any other table space specified. The catalog fields that reflect the extent of a check pending state are updated to ensure that they indicate that no check is required.

System action:　Processing continues.

Severity:　4 (warning)

DSNU728I　*csect-name* **INVALID EXCEPTION TABLE** *exception-table* **FOR THE** *dependent-table***.**

Explanation:　The CHECK DATA utility encountered one of the following errors:

- Row or column access control is enforced for the dependent table, but it is not enforced for the exception table.
- The length of the RID column of the exception table does not match the RID size of the table space for the table that is being checked.

System action:　The CHECK DATA utility will terminate processing after all the exception tables have been examined.

User response:　If row or column access control is enforced for the dependent table, ensure that it enforced for the exception table.

If the length of the RID column of the exception table does not match the RID size of the table space, alter the length of the RID column. CHAR(5) is required for LARGE or DSSIZE table spaces.

Severity:　8 (error)

Related reference:

➡ ALTER TABLE (DB2 SQL)

Related information:

➡ Managing access through row permissions and column masks (Managing Security)

DSNU729I　*csect-name* **TABLE** *table-name* **NOT SPECIFIED IN THE 'FOR EXCEPTION' CLAUSE**

Explanation:　The CHECK DATA utility did not find a required table listed in the FOR EXCEPTION clause. All tables with constraints in the table spaces being checked and their descendents must be listed in the FOR EXCEPTION clause of CHECK DATA utility.

System action:　The CHECK DATA utility terminates processing after all the required tables are examined.

User response:　Examine the utility input statement to determine the reason for the missing table named in the

message. Correct the input statement to include the missing table and resubmit the job.

Severity: 8 (error)

Related reference:

➡ CHECK DATA (DB2 Utilities)

➡ Syntax and options of the CHECK DATA control statement (DB2 Utilities)

DSNU730I   *csect-name* **CHECKING TABLE** *tab*

Explanation: This message is issued when the CHECK DATA utility begins to process a table. The name of table being checked is 'tab'.

System action: Processing continues.

Severity: 0 (informational)

DSNU731I   *csect-name* **EXCEPTION LIMIT REACHED**

Explanation: The EXCEPTION limit specified on the CHECK utility statement has been reached.

System action: The CHECK utility terminates processing.

User response: If required, increase the exception limit and resubmit the job.

Severity: 8 (error)

DSNU732I   **NO TABLE TO BE CHECKED FOR TABLESPACE** *tsp*

Explanation: The CHECK DATA utility determined that none of the tables in table space *tsp* has a constraint defined. Therefore, no table in that table space is checked.

System action: Utility processing continues.

Severity: 4 (warning)

DSNU733I   *csect-name* **ROW (RID=***row-identifier***) HAS NO PARENT FOR** *relationship-identifier*

Explanation: Th CHECK DATA utility found a row in a dependent table without a parent. The row with the *row-identifier* RID value has no parent for the relationship named by the *relationship-identifier* value. This message is always proceeded by a DSNU730I message that identifies the table in error.

System action: Processing continues.

User response: Resolve the check pending status.

Severity: 4 (warning)

DSNU735I • DSNU739I

Related concepts:

➭ Implications of adding parent or foreign keys (DB2 Administration Guide)

Related tasks:

➭ Adding referential constraints to existing tables (DB2 Administration Guide)

➭ Collecting data for failures after a point-in-time recovery (Collecting data)

Related reference:

➭ CHECK-pending status (DB2 Utilities)

➭ CHECK DATA (DB2 Utilities)

Related information:

"DSNU730I" on page 787

➭ Recovering from referential constraint violation (DB2 Administration Guide)

DSNU735I   *csect-name* **DUPLICATE TABLESPACE** *tsp* **PARTITION** *part* **NAMES EXIST IN THE TABLESPACE LIST**

Explanation: This message is issued by the CHECK DATA utility when a duplicate table space 'tsp' and partition 'part' is found in the table space list. An entry in the table space list with a partition specified is considered to be a duplicate when another entry for the same table space with no partition specified (defaults to all partitions for a partitioned table space) is found. In this case, the table space list entry with no partition specified has priority and the entry with a partition specified is skipped. The CHECK DATA utility proceeds and the duplicate entry is skipped.

System action: Processing continues.

Severity: 4 (warning)

DSNU736I   *csect-name* **SCOPE PENDING IS ACTIVE, BUT THE ENTIRE TABLE SPACE** *tsp* **PARTITION** *part* **MUST BE CHECKED**

Explanation: This message is issued by the CHECK DATA utility when checking with the option SCOPE PENDING. Table space *tsp* partition *part* is in check pending status, but the utility is checking all rows in that table space partition.

System action: The utility continues processing.

Severity: 4 (warning)

DSNU737I   *csect-name* **TABLESPACE** *tsp* **PARTITION** *part* **IS NOT CHECK PENDING**

Explanation: This message is issued when 'CHECK DATA TABLESPACE tsp PART part SCOPE PENDING' is specified and the CHECK DATA utility finds that the table space partition is not in a check pending state. The CHECK DATA utility skips table space 'tsp' partition 'part' and continues with any other table space specified. The catalog fields that reflect the extent of a check pending state are updated to ensure that they indicate that no check is required.

System action: Processing continues.

Severity: 4 (warning)

DSNU739I   *csect-name* **CHECK TABLE** *tab* **COMPLETE,** *elapsed time=t1*

Explanation: This message is issued when the CHECK DATA utility finishes processing a table. The elapsed clock time that the check of table 'tab' consumed is 't1'.

Severity: 0 (informational)

DSNU740I   *csect-name* **CHECKING TABLE** *tab* **PARTITION** *part*

Explanation: This message is issued when the CHECK DATA utility begins to process a table of a partitioned table space in which a single partition was specified. The name of the table and the number of the partition being checked are identified in the message.

System action: Processing continues.

Severity: 0 (informational)

DSNU741I   *csect-name* *nn* **ROWS DELETED FROM TABLE** *tab*

Explanation: This message is issued by the CHECK DATA utility when the DELETE YES option is specified. The number of rows deleted from TABLE *tab* is the number of rows that contain referential integrity violations or check constraint violations, plus their descendent rows. The number of rows deleted from TABLE *tab* is equal to the number of rows inserted into the corresponding exception table.

System action: Processing continues.

Severity: 4 (warning)

DSNU742I   *csect-name* **ROW (RID=**row-identifier**) VIOLATES CHECK CONSTRAINT** *constraint-name* **ON TABLE** *table-name*

Explanation: The CHECK DATA utility found that the row with RID=*row-identifier* in table *table-name* violated check constraint *constraint-name*.

System action: Processing continues

User response: Resolve the check pending status.

Severity: 4 (warning)

Related concepts:

Implications of adding parent or foreign keys (DB2 Administration Guide)

Related tasks:

Collecting data for failures after a point-in-time recovery (Collecting data)

Adding referential constraints to existing tables (DB2 Administration Guide)

Related reference:

Advisory or restrictive states (DB2 Utilities)

CHECK-pending status (DB2 Utilities)

CHECK DATA (DB2 Utilities)

Related information:

Recovering from referential constraint violation (DB2 Administration Guide)

DSNU743I   *csect-name* **- LOB IS INVALID. ROWID=**rowid **VERSION=**version

Explanation: The CHECK LOB utility found that the LOB with ROWID *rowid* and VERSION *version* is marked invalid.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to auxiliary warning (AUXW). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

User response: You must use SQL to update the LOB column or delete the row that contains the LOB column to remove the invalid indication. REPAIR LOCATE ROWID VERSION DELETE can be used to remove the entire LOB from the LOB table space.

Severity: 4

DSNU745I   *csect-name* **- ROWID=***rowid* **VERSION=***version* **RID=***rid* **MISSING IN LOB TABLE**
             SPACE=*lob-table-space-name*

Explanation: The CHECK INDEX utility found an index entry key with ROWID *rowid* and VERSION *version* in the index, but not in the LOB table space named *lob-table-space-name*.

System action: The utility returns completion code=4. The index is set Recover Pending.

Programmer response: Use REBUILD INDEX or RECOVER to correct the index on the auxiliary table.

DSNU747I   *csect-name* **- ROWID=***rowid* **VERSION=***version* **MISSING IN INDEX=***index-name*

Explanation: The CHECK INDEX utility found a LOB with ROWID *rowid* and VERSION *version* in the LOB table space but not in the index on the auxiliary table, *index-name*.

System action: Utility returns completion code=4. The index is set Recover Pending.

Programmer response: Use REBUILD INDEX to correct the index on the auxiliary table.

DSNU749I   *csect-name* **CHECK DATA COMPLETE, ELAPSED TIME=***t1*

Explanation: This message is issued when the CHECK DATA utility is complete. The elapsed clock time that the CHECK DATA utility consumed is 't1'.

System action: Processing ends.

Severity: 0 (informational)

DSNU750I   *csect-name* **CATMAINT** *operation* **PHASE** *phase-number* **STARTED**

Explanation: The CATMAINT utility phase *phase-number* processing started for the indicated operation.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU751I   *csect-name* **CATALOG** *operation* **OF TABLE -** *table-name*

Explanation: This message is issued the first time data is operated upon for the designated table.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU752I   *csect-name* **CATMAINT** *operation* **PHASE** *phase-number* **COMPLETED**

Explanation: The CATMAINT utility PHASE *phase-number* processing completed for the indicated operation.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU756I   *csect-name* **CATALOG TABLE IS EMPTY -** *table-name*

Explanation: This message is issued when the named table is empty.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU758I   *csect-name* **- WARNING DURING PROCESSING SQL STATEMENT - SQLCODE IS:** *sql-code* **SQL**
             MESSAGE TEXT: *message-text***. SQL STATEMENT:** *sql-statement* **SQLERRP:** *error-procedure* **SQLERRD:**
             sqlerrd-fields

Explanation: CATMAINT executed an SQL statement as part of CATMAINT UPDATE processing. DB2 returned a non-zero SQLCODE. This message does not indicate a problem in the CATMAINT process.

csect-name
> The name of the control section that issued the message.

sql-code The SQL code that was returned for the SQL statement.

sql-code A numeric value that indicates why the SQL statement executed abnormally.

message-text
> Indicates the problem with the SQL statement.

sql-statement
> The SQL statement that encountered the error.

error-procedure
> The name of the procedure that detected the error. This value corresponds to the SQLERRP SQCLCA field.

sqlerrd-fields
> A series of fields that contain a values that indicate the number of rows that qualified for the SQL statement, the number of rows in the result table, a rough estimate of required resources, or an internal error code. These values correspond to the SQLERRD(*n*) SQLCA fields.

System action: Utility processing continues normally.

User response: No action is required.

Severity: 0

Related reference:

▶ Description of SQLCA fields (DB2 SQL)

DSNU759I   *csect-name* **INCORRECT UNLOAD DATASET HAS BEEN DETECTED**

Explanation: This message indicates an inconsistency between the release level of the unloaded catalog and the DB2 release that is running the CATMAINT utility.

System action: Utility processing terminates.

User response: Review the JCL for the job to ensure that the correct unload data set is specified. Verify that the correct level of DB2 is up.

Severity: 8 (error)

DSNU760I   *csect-name* **SPECIFIED USER NOT DEFINED WITH INSTALLATION SYSADM AUTHORITY**

Explanation: The utility can be run only under a user defined with the installation SYSADM authority.

System action: Utility processing terminates.

User response: Rerun the utility with the proper authorization.

Severity: 8 (error)

DSNU762I   *csect-name* **DUPLICATE RECORD DETECTED FOR TABLENAME -** *table-name* **SEQNO OF THE INPUT RECORD IS -** *seqno* **FOLLOWING IS UP TO 119 BYTES OFRECORD BEING INSERTED**

Explanation: An object created in the target catalog has the same name as an object being migrated from the source catalog.

System action: Utility abends.

User response: User data should not be in the target catalog before running the CATMAINT utility. Free or drop the object from the target catalog and rerun the CATMAINT utility.

Severity: 8 (error)

DSNU764I   **csect-name FOLLOWING IS UP TO 119 BYTES OF RECORDFOR TABLE table-name, SEQNO seqno BEING PROCESSED AT THE TIME OF ERROR**

Explanation: An abend occurred during the execution of the CATMAINT utility, during the processing of the table, 'table-name', with a sequence number of 'seqno'. The cause of the abend is one of the following:

- The record sequence number was invalid.
- An unexpected return code was received from DSNUEXLO.
- The table name was not found in the load table (DSNUELTB).
- The parent record could not be positioned.
- The position OBID was zero (DSNUELTB has incorrect value).
- The record action code is incorrect.

Following this message, up to the first 119 bytes of the recordin question is displayed.

System action: Utility abends.

User response: Verify that the record in the unloaded data set is correct.

Severity: 8 (error)

DSNU765I   *csect-name* **THE DB2 CATALOG IS NOT AT THE REQUIRED LEVEL. CATALOG NOT UPDATED**

Explanation: The DB2 catalog is not at the proper catalog level to migrate the catalog. This error can occur due to the following reasons:

- Attempting to migrate the catalog from a DB2 release that does not support migration
- Attempting to migrate DB2 Stored Procedures prior to a catalog migration

System action: Utility processing terminates

User response: Fall back to the previous release if the catalog is not at the correct level for a migration. Migrate only from DB2 releases that support migration.

Severity: 8 (error)

DSNU766I   *csect-name* **THE CATALOG HAS ALREADY BEEN MIGRATED. NO ACTION WILL BE TAKEN**

Explanation: The catalog was already successfully migrated. There is no reason to rerun the catalog migration job. Therefore, no action is taken to update the DB2 catalog.

System action: Processing ends.

Severity: 0 (informational)

DSNU767I   *csect-name* **- INVALID** *object-name* **IS SPECIFIED, REASON** *reason-code*

Explanation: The CATMAINT utility encountered an invalid object name during update processing.

csect-name
>   The name of the control section that issued the message.

object-name
>   The invalid name, which corresponds to the specific CATMAINT UPDATE option.

>   **CATMAINT UPDATE OWNER**
>>   The name identifies an object owner.

>   **CATMAINT UPDATE SCHEMA**
>>   The name identifies an owner, creator, or schema.

>   **CATMAINT UPDATE VCAT**
>>   The name identifies an integrated catalog facility catalog.

reason-code
>   A numeric value that indicates the reason for the failure.

1	The name cannot be longer than 8 bytes in EBCDIC representation.
2	The name cannot be specified more than once.
3	The name cannot be the qualifier of an existing object.
4	The name "SYSIBM" is not allowed.
5	The name cannot be the qualifier of an existing object that is referenced by:

- A view
- A function
- A materialized query table
- A trigger
- A native SQL procedure
- An expression-based index
- A column mask
- A row permission

6	The OWNER and SCHEMA clauses are mutually exclusive, and cannot be specified in the same CATMAINT UPDATE statement.
7	The OWNER option is specified, but no role is present. The user has to run under a trusted context.
8	A name cannot be referenced in check conditions.
9	The name is too long to fit in PATHSCHEMA. PATHSCHEMA cannot exceed 2048 bytes.

System action: The utility terminates with return code 8.

User response: Remove or modify the invalid name, and then rerun the utility.

Severity: 8 (error)

Related reference:

CATMAINT (DB2 Utilities)

DSNU768I   *csect-name* **CATMAINT CONVERT COMPLETE FOR INDEX** *index-name*

Explanation: The conversion is complete for index *index-name*.

System action: Processing continues with the next index.

Severity: 0 (informational)

DSNU769I   *csect-name* **THE BUILD PHASE FAILED FOR INDEX** *index-name* **REASON** *reason*

Explanation: The CATMAINT CONVERT build phase failed for index *index-name*. This message and others identify the actions that should be taken.

System action: Utility processing continues with the next index and will terminate with return code 8.

Operator response: Note the reason code and the index name. See the Problem Determination section for an explanation of the collected data.

User response: Correct the problem indicated by the reason code. Either resubmit the CATMAINT CONVERT job or recover the index that failed.

Severity: 8 (error)

Related concepts:

DB2 reason codes (DB2 Codes)

Related reference:

CATMAINT (DB2 Utilities)

DSNU770I   *csect-name* **- DBD** *type* **STARTED FOR DATABASE** *database*

Explanation: This message marks the start of the REPAIR DBD type utility. In the message, 'type' can be DIAGNOSE, REBUILD, or TEST. The database name specified in the REPAIR statement is 'database'.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU771I   *csect-name* **- DBD** *type* **COMPLETE FOR DATABASE** *database*

Explanation: This message marks the completion of the REPAIR DBD type utility. In the message, 'type' can be DIAGNOSE, REBUILD, or TEST. The database name specified in the REPAIR statement is 'database'.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU772I   *csect-name* **- THE** *object-type qualifier.object-name* **IS UNAVAILABLE IN THIS RELEASE**

Explanation: This message indicates that the specified object has a dependency on a release from which fall back has occurred, and cannot be operated upon in the current release.

System action: Utility processing terminates.

User response: Check the IBMREQD column of the catalog table associated with the specified object type. If the value in the catalog table is the release dependency mark for a release of DB2 from which fall back has occurred, the utility is not operational for the current release of the targeted database.

Severity: 8 (error)

DSNU773I   *csect-name* **-** *database* **IS A SYSTEM DATABASE AND CANNOT BE PROCESSED**

Explanation: The database indicated is not eligible for processing by the type of the REPAIR DBD utility involved.

System action: Utility processing terminates.

User response: Ensure that the utility control statement specifies the correct database.

Severity: 8 (error)

DSNU774I   *csect-name* **- SPECIFIED DATABASE** *database* **NOT FOUND**

Explanation: The database indicated by the utility control statement was not found in catalog table SYSIBM.SYSDATABASE.

System action: Utility processing terminates.

User response: Ensure that the database indicated by the utility control statement exists in SYSIBM.SYSDATABASE.

Severity: 8 (error)

DSNU775I   *csect-name* **- DROP OF DATABASE** *dbname* **FAILED WITH SQLCODE =** *sql-code*

Explanation: An error was encountered while using the REPAIR DBD DROP utility to drop DATABASE *dbname*. The SQL error code is indicated by *sql-code*.

System action: DATABASE *dbname* is not dropped.

User response: Take the action that is specified for the SQL code that is indicated *sql-code* value.

Severity: 8 (error)

Related concepts:

➡ SQL codes (DB2 Codes)

Related tasks:

➡ Repairing DBDs (DB2 Utilities)

Related reference:

➡ REPAIR (DB2 Utilities)

DSNU776I *csect-name* **CATMAINT** *operation* **ERROR -** *error-text*

Explanation: The CATMAINT utility halted because the catalog contains view(s) on the SYSCOLDIST and/or SYSCOLDISTSTATS catalog tables.

System action: Utility processing halts.

User response: Remove the views on these catalog tables. Then retry the failing CATMAINT *operation*. This message is followed by an abend04E RC00E40601. See the problem determination text of RC00E40601 for additional information.

Severity: 0 (informational)

DSNU777I *csect-name* **CATMAINT** *operation* **STATUS** *status-msg*

Explanation: The CATMAINT utility is processing CATMAINT *operation*. *status-msg* indicates where CATMAINT is in the *operation* process.

System action: Utility processing continues normally.

Severity: 0 (informational)

DSNU778I *csect-name* **- ERROR PROCESSING SQL STATEMENT - SQL CODE IS:** *sql-code* **SQL MESSAGE TEXT:** *message-text* **SQL STATEMENT:** *sql-statement* **SQLERRP:** *error-procedure* **SQLERRD:** *sqlerrd-fields*

Explanation: A utility terminated abnormally in the DB2 database services address space (ssnmDBM1) during CATMAINT UPDATE processing.

csect-name
     The name of the control section that issued the message.

sql-code A numeric value that indicates why the SQL statement execution was unsuccessful.

message-text
     Indicates the problem with the SQL statement.

sql-statement
     The SQL statement that encountered the error.

error-procedure
     The name of the procedure that detected the error. This value corresponds to the SQLERRP SQCLCA field.

sqlerrd-fields
     A series of fields that contain a values that indicate the number of rows that qualified for the SQL statement, the number of rows in the result table, a rough estimate of required resources, or an internal error code. These values correspond to the SQLERRD(*n*) SQLCA fields.

System action: Utility processing terminates abnormally.

User response: This message is followed by an abend04E RC00E40601. The *sql-code*, *message-text*, and *sql-statement* listed in this message are probably sufficient to diagnose and correct the problem. See the problem determination text of RC00E40601 for additional information.

Severity: 0

DSNU780I • DSNU783I

➦ Description of SQLCA fields (DB2 SQL)

Related information:

➦ 00E40601 (DB2 Codes)

DSNU780I   *csect-name* **DSNUTILB LEVEL** *rel1* **INVALID WITH SUBSYSTEM LEVEL** *rel2*

Explanation: JCL has been submitted which executes level 'rel1' of DSNUTILB against a level 'rel2' DB2 subsystem. The levels must match.

System action: The utility terminates with a return code of 8.

User response: Determine the DB2 subsystem against which you want to execute. Determine which STEPLIB data set contains the corresponding DSNUTILB load module. Correct the JCL to specify the correct data set on the STEPLIB DD statement and the correct DB2 subsystem on the EXEC statement and resubmit the job.

Severity: 8 (error)

DSNU781I   *csect-name* **RESTART OF LEVEL** *rel1* **UTILITY AT LEVEL** *rel2* **INVALID**

Explanation: JCL has been submitted which attempts to RESTART a stopped utility. The utility began processing at level *rel1* and cannot be restarted at level *rel2*.

System action: The utility job is abended with reason code X'00E40018'.

User response: Either restart the utility at the same level it was originated or terminate the utility with the TERM command and resubmit it at the new level.

Related concepts:

➦ Restart of an online utility (DB2 Utilities)

Related reference:

➦ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

➦ 00E40018 (DB2 Codes)

DSNU782I   *csect-name* **UNABLE TO RECOVER DUE TO ACTIVITY AT ANOTHER SYSTEM LEVEL**

Explanation: The RECOVER utility is unable to recover the specified table space. Activity at another system level has left all image copies for this table space inconsistent with the DB2 LOG.

System action: The utility terminates with a return code of 8.

User response: The table space, in its entirety can not be recovered. Individual data sets may be recoverable using the DSNUM() keyword.

Severity: 8 (error)

DSNU783I   *csect-name* **INDEXES DSNSCT02, DSNLUX01, DSNLUX02 and DSNLLX02 HAVE SUBPAGES VALUE GREATER THAN 1 AND CANNOT BECOME GROUP BUFFER POOL DEPENDENT.**

Explanation: CATMAINT CONVERT TO TYPE 1 DIRECTORY alters the indexes DSNSCT02, DSNLUX01, DSNLUX02, and DSNLLX02 to type 1 with more than one subpage. In a data sharing environment, there cannot be inter-DB2 R/W interest on type 1 indexes with more than one subpage (they cannot become *group buffer pool dependent*). This means that any read or update request that causes inter-DB2 R/W interest on these directory indexes is denied with a 'resource unavailable' condition.

For directory indexes, only type 2 indexes can have full R/W sharing among data sharing members.

System action: Processing continues.

User response: With very few exceptions, indexes on the directory will have inter-DB2 R/W interest if there are two or more members in your DB2 data sharing group. Use CATMAINT CONVERT TO TYPE 2 DIRECTORY to convert

the indexes to type 2 if you plan to have two or more DB2 members.

Severity: 4 (warning)

DSNU784I   *csect-name* **CATMAINT** *utility-operation* **- CATALOG CONTAINS UNSUPPORTED OBJECTS:** *error-text* **OBJECT NAME -** *object-name***, CREATOR -** *creator-name*

Explanation: The CATMAINT utility halted because the catalog contains an unsupported object or item.

csect-name
> The name of the control section that issued the message.

utility-operation
> The CATMAINT operation that halted.

error-text
> The reason that the operation halted.

object-name
> The name of the object.

creator-name
> The creator of the object.

System action: Utility processing halts. No SYSABEND dump or SVC dump is requested. Code 00E40601 is issued.

User response:
1. Use the DSNTIJPM and DSNTESQ jobs to identify any unsupported items and objects in the subsystem.
2. Remove the unsupported objects and items from the DB2 catalog in the previous release.
3. Retry the failing CATMAINT operation.

Severity: 8 (error)

Related reference:

➡ CATMAINT (DB2 Utilities)

Related information:

➡ 00E40601 (DB2 Codes)

DSNU790I   *csect-name utility-name option* **IS NOT ALLOWED WHEN THERE ARE OUTSTANDING UTILITIES STARTED FROM PRIOR RELEASES.**

Explanation: The utility to be executed may have conflicts with other outstanding utilities started from prior releases.

utility-name is the name of the utility that detected the conflict.

option is the option specified to the utility that detected the conflict.

System action: The request is rejected.

User response: Use the DISPLAY UTILITY command to see what utilities are outstanding. Issue the TERM UTILITY command on the same release as the release on which the outstanding utility was started.

DSNU795I   *csect-name* **- CHECKLOB PHASE COMPLETE - ELAPSED TIME=***time*

Explanation: The CHECK LOB utility CHECKLOB phase was completed with the elapsed time of *time*.

System action: The CHECKLOB phase is completed.

Severity: 0 (informational)

DSNU796I   *csect-name* **- REPRTLOB PHASE COMPLETE - ELAPSED TIME=***time*

Explanation: The CHECK LOB utility REPRTLOB phase was completed with the elapsed time of *time*.

System action: The REPRTLOB phase is completed.

Severity: 0 (informational)

DSNU805I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **IS OUT OF SYNCH ROWID=***rowid*
               VERSION=*version*

Explanation: The CHECK DATA utility found an auxiliary index entry for the LOB column named *column_name* with ROWID, *rowid*, and version, *version*. The base table row with ROWID, *rowid*, has a version value greater then that of the index record for the column specified.

The state changes below take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: If **AUXERROR REPORT** was specified, this error is reported only. The base table space is set to an Aux Check Pending state.

If **AUXERROR INVALIDATE** was specified, the LOB column is set invalid. The base table space is set to an Aux Warning state.

Programmer response: CHECK DATA relies on information in LOB table space and in the auxiliary index being correct. Before running CHECK DATA to check auxiliary columns, first run CHECK LOB and then run REBUILD INDEX or CHECK INDEX. CHECK LOB ensures the validity of the LOB table space. REBUILD INDEX or CHECK INDEX ensures the validity of the auxiliary index.

If **AUXERROR REPORT** was specified, run the CHECK DATA utility with **AUXERROR INVALIDATE** to set the base table column invalid.

If **AUXERROR INVALIDATE** was specified, use SQL to update the LOB column or delete the base table row. Use REPAIR to delete the LOB from the LOB table space using the ROWID and version reported in the message. Rerun the CHECK DATA utility with **AUXERROR INVALIDATE** specified. If all Auxiliary Column Check errors were corrected, Aux Warning is reset.

The REPAIR utility can be used to reset the Aux Warning state.

Severity: 4 (Warning)

DSNU806I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **WAS SET INVALID ROWID=***rowid*
               VERSION=*version*

Explanation: The CHECK DATA utility with **AUXERROR INVALIDATE** specified, set the LOB column named *column_name* invalid. The LOB column was in the row with ROWID, *rowid* in the table named *table_name*. The version for the LOB column was *version*.

System action: Utility returns completion code=4. The base table space is set to a Aux Warning state.

Programmer response: None

Use SQL to update the LOB column or delete the base table row. Rerun the CHECK DATA utility with **AUXERROR INVALIDATE** specified. If all Auxilary Column Check errors were corrected, Aux Warning will be reset.

The REPAIR utility can be used to reset the Aux Warning state.

DSNU807I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **WAS SET VALID ROWID=***rowid*
               VERSION=*version*

Explanation: The CHECK DATA utility with **AUXERROR INVALIDATE** specified, set the LOB column named *column_name* valid. The LOB column was in the row with ROWID, *rowid* in the table named *table_name*. The version for the LOB column was *version*.

System action: Utility returns completion code=4.

Programmer response: None. If all Auxilary Column Check errors were corrected, Aux Warning will be reset.

DSNU809I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **IS MISSING IN INDEX** *index-name* **ROWID=***rowid* **VERSION=***version*

Explanation: The CHECK DATA utility found a record in the table named *table-name* in the row with a ROWID of *rowid*. The column named *column-name* did not have an index entry in the on the auxiliary index named *index-name*. The version for the LOB column was *version*.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: If **AUXERROR REPORT** was specified, this error is reported only. The base table space is set to an Aux Check Pending state.

If **AUXERROR INVALIDATE** was specified, the LOB column is set invalid. The base table space is set to an Aux Warning state.

Utility returns completion code=4.

Programmer response: CHECK DATA relies on information in LOB table space and in the auxiliary index being correct. Before running CHECK DATA to check auxiliary columns, first run CHECK LOB and then run REBUILD INDEX or CHECK INDEX. CHECK LOB ensures the validity of the LOB table space. REBUILD INDEX or CHECK INDEX ensures the validity of the auxiliary index.

If **AUXERROR REPORT** was specified, run the CHECK DATA utility with **AUXERROR INVALIDATE** to set the base table column invalid.

If **AUXERROR INVALIDATE** was specified, use SQL to update the LOB column or delete the base table row. Rerun the CHECK DATA utility with **AUXERROR INVALIDATE** specified. If all Auxiliary Column Check errors were corrected, Aux Warning will be reset.

The REPAIR utility can be used to reset the Aux Warning state.

DSNU811I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **IS INVALID ROWID=***rowid* **VERSION=***version*

Explanation: The CHECK DATA utility found a record in the table named *table-name* in the row with ROWID *rowid*. The column named *column-name* was marked invalid by a prior execution of the CHECK DATA utility with **AUXERROR INVALIDATE** specified. The version for the LOB column was *version*.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: Utility returns completion code=4. The base table space is set to a Aux Warning state.

Programmer response: If **AUXERROR INVALIDATE** was specified, use SQL to update the LOB column or delete the base table row. Rerun the CHECK DATA utility with **AUXERROR INVALIDATE** specified. If all Auxiliary Column Check errors were corrected, Aux Warning will be reset.

The REPAIR utility can be used to reset the Aux Warning state.

DSNU813I   *csect-name* **- LOB IN TABLE SPACE=***lob_table_space_name* **WITH ROWID=***rowid* **AND VERSION=***version* **HAS NO BASE TABLE ROW**

Explanation: The CHECK DATA utility found a LOB in the LOB table space named *lob_table_space_name* but did not find a row in the base table for this LOB. The LOB has a ROWID of *rowid* and version of *version*.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: Utility returns completion code=4.

Programmer response: CHECK DATA relies on information in the index on the auxiliary table and LOB table space being correct. Before running CHECK DATA to check auxiliary columns, first run CHECK LOB and then run REBUILD INDEX or CHECK INDEX. CHECK LOB ensures the validity of the LOB table space. REBUILD INDEX or CHECK INDEX ensures the validity of the auxiliary index.

The LOB can be deleted using the REPAIR utility. Specify LOCATE ROWID VERSION DELETE.

DSNU814I   *csect-name* **- TABLE=***table_name* **COLUMN=***column_name* **IS ZERO OR NULL BUT ENTRY IN INDEX** *index-name* **FOUND ROWID=***rowid* **VERSION=***version*

Explanation: The CHECK DATA utility found a record in the table named *table-name* in the row with a ROWID of *rowid*. The column named *column-name* was indicated null or zero length, but an index entry with the same ROWID was found in the index on the auxiliary table named *index-name*. The version for the LOB column was *version*.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: The fact that a LOB column entry exists in the index on the auxiliary table when the LOB column is marked null or has a null length does not hinder LOB column processing.

If **AUXERROR INVALIDATE** was specified, the LOB column is not set invalid for this error.

Programmer response: CHECK DATA relies on information in the index on the auxiliary table and LOB table space being correct. If you did not run CHECK LOB followed by REBUILD INDEX or CHECK INDEX prior to running CHECK DATA, then before running CHECK DATA again, run CHECK LOB followed by REBUILD INDEX or CHECK INDEX. CHECK LOB checks the validity of the LOB table space and REBUILD INDEX or CHECK INDEX ensures that the auxiliary index is valid.

The LOB can be deleted from the LOB table space by using **REPAIR LOCATE TABLESPACE ROWID DELETE**

Severity: 4 (warning)

DSNU815I   *csect-name* **- TABLESPACE** *table-space-name* **IS IN AUX CHECK PENDING STATE**

Explanation: If the CHECK DATA utility issued the message then the utility found *auxiliary check* errors. *AUXERROR REPORT* was specified, so errors were left as found.

Auxiliary check pending is a prohibitive state.

Previous error messages report the errors found. Regarding other utilities that can tolerate *auxiliary check pending*, the pending state will be reported.

System action: Utility returns completion code 4 or 8.

Programmer response: Use the CHECK DATA with *AUXERROR INVALIDATE*. This will modify the *auxiliary check* errors so they can be corrected and set the state of the table space to *auxiliary warning*.

After running CHECK DATA with *AUXERROR INVALIDATE*, use SQL to update the LOB column or delete the base table row. Rerun the CHECK DATA utility with *AUXERROR INVALIDATE* specified. If all *auxiliary check* errors were corrected, *auxiliary warning* will be reset.

The *auxiliary check pending* state can also be reset using REPAIR with *SET NOAUXCHKP*. Resetting *auxiliary check pending* with REPAIR will have unpredictable results.

DSNU816I   *csect-name* **- TABLESPACE** *table-space-name* **IS IN AUX WARNING STATE**

Explanation: The CHECK DATA utility found *auxiliary check* errors. *AUXERROR INVALIDATE* was specified, so LOB columns in error were marked invalid.

Auxiliary warning is a non-prohibitive state. Updating the LOB column or deleting the row containing the LOB column by SQL is permitted. Valid LOBs may be accessed, however any attempt to retrieve an invalid modified LOB column by SQL will result in a -904 return code.

Previous error messages report the errors found.

System action: Processing continues.

Programmer response: Use SQL to update or delete the LOB columns in error. Use CHECK DATA with AUXERROR INVALIDATE to verify all errors were corrected and to reset the *auxiliary warning* state.

The *auxiliary warning* state can also be reset using REPAIR with *SET NOAUXWARN*.

Severity: 0 (informational)

DSNU817I   *csect-name-* **TABLE=***table-name* **COLUMN=***column-name* **IS COMPLETELY INLINE BUT ENTRY IN INDEX** *index-name* **FOUND ROWID=***row-id* **VERSION=***version*

Explanation: The CHECK DATA utility found a record for a row whose column was indicated to be completely inline, but an index entry with the same ROWID was found in the index on the auxiliary table.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

column-name
> The name of the column.

index-name
> The name of the index on the auxiliary table.

row-id   The ROWID of the row.

version   The version for the LOB column.

System action: LOB column processing is unaffected. If the AUXERROR INVALIDATE option was specified, the LOB column is not set to an invalid status for this error.

Programmer response: CHECK DATA relies on information in the index on the auxiliary table and LOB table space being correct. To ensure that this information is correct:

1. Run CHECK LOB to check the validity of the LOB table space.
2. Run either REBUILD INDEX or CHECK INDEX to ensure that the auxiliary index is valid.
3. Run CHECK DATA.

Severity: 4 (warning)

Related reference:

➡️ CHECK DATA (DB2 Utilities)

➡️ CHECK INDEX (DB2 Utilities)

➡️ CHECK LOB (DB2 Utilities)

➡️ REBUILD INDEX (DB2 Utilities)

DSNU820I   **XML COLUMN** *column-name* **IN TABLE** *tbcreator*, *tbname* **INVALID. DOCID** *DocId hex-string*

Explanation: This message is issued when an XML column value is marked invalid. The column was marked invalid during previous CHECK DATA processing for **AUXERROR INVALIDATE**, or **XMLERROR INVALIDATE**.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: The XML table space is set to *Aux Warning*.

Programmer response: You must replace the XML column, or delete the row containing the XML column. Use SQL to make the XML valid.

Severity: 4 (warning)

DSNU822I   **XML COLUMN** *column-name* **IN TABLE** *tbcreator*, *tbname* **IS MISSING IN INDEX** *ixcreator*, *ixname* **DOCID** *DocId-hex-string*

Explanation: This message is issued when a base table row contains an XML column value, but no index entry is found for the XML column value in the NODEID index.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: If you specified **AUXERROR REPORT**, then only this error is reported. The base table space is set to

an *auxiliary check pending* state. If you specified **AUXERROR INVALIDATE**, then the XML column is set to invalid. The base table space is set to an *auxiliary warning* state.

Programmer response: Before running CHECK DATA to check the auxiliary columns, run REBUILD INDEX or CHECK INDEX. REBUILD INDEX or CHECK INDEX ensures the validity of the auxiliary index.

If you specified **AUXERROR REPORT**, run the CHECK DATA utility with **AUXERROR INVALIDATE** in order to set the base table column to invalid.

If you specified **AUXERROR INVALIDATE**, use SQL to update the XML column, or delete the base table row. Run the CHECK DATA utility again, with **AUXERROR INVALIDATE** specified.

If all auxiliary column check errors were corrected, *auxiliary warning* will be reset.

Severity: 4 (warning)

DSNU823I   *csect-name* **ORPHANED XML DOCUMENT FOUND FOR XML COLUMN** *column-name* **IN BASE TABLE** *creator-name.table-name*, **DOCID** *document-id*

Explanation: The CHECK DATA utility found an XML document in the XML table space, but the document is not referenced by a base table row.

csect-name
> The name of the control section that issued the message.

column-name
> The name of the XML column.

creator-name.table-name
> The name of the table, qualified by the name of the creator of the table.

document-id
> The hexadecimal document identifier that is specified for the XML document.

System action: Processing continues until you receive completion code 4.

The state changes take place if SHRLEVEL REFERENCE is specified or is the default option. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

This is an informational message. XML processing is not hindered because an XML column entry exists in the node ID index, but not in the base table.

Programmer response: To delete the orphaned XML document, run REPAIR LOCATE TABLESPACE *xml-tablespace* DOCID *document-id* DELETE.

Severity: 4 (warning)
Related reference:

➡ REPAIR (DB2 Utilities)

DSNU824I   **XML COLUMN** *column-name* **IN TABLE** *crname*, *tbname* **IS NULL BUT ENTRY IN INDEX** *crname*, *ixname* **FOUND. DOCID** *DocId-hex-string*

Explanation: This message is issued when a row in an XML table exists, but it is not referenced by a base table row.

The state changes take place if SHRLEVEL REFERENCE is specified or defaulted. If SHRLEVEL CHANGE is specified, the state of the table space remains unchanged.

System action: Processing continues. This is an informational message. XML processing is not hindered because an XML column entry exists in the auxiliary index, but the XML column in the base table is marked null or has a null length.

Programmer response: Before running CHECK DATA to check the auxiliary columns, run REBUILD INDEX or CHECK INDEX. REBUILD INDEX or CHECK INDEX ensures the validity of the auxiliary index.

Severity: 4 (warning)

DSNU826I   **XML COLUMN** *column name* **IN TABLE** *tbcreator*, *tbname* **HAS BEEN MARKED INVALID BY THE CHECK UTILITY. DOCID** *DocId hex string*

Explanation: This message is issued when an XML column value is marked invalid during CHECK DATA processing for **AUXERROR INVALIDATE** or **XMLERROR INVALIDATE**.

System action: The base table space is set to Aux Warning.

Programmer response: You must replace the XML column or delete the row containing the XML. Use SQL to remove the invalid indication.

Severity: 4 (warning)

DSNU827I   *csect-name*- **XML COLUMN***column-name* **FOR TABLE** *creator-name*.*table-name* **WAS SET VALID. DOCID** *document-id*

Explanation: An XML column value that had previously been marked "invalid" has been changed to "valid" during CHECK DATA utility processing for either AUXERROR INVALIDATE or XMLERROR INVALIDATE.

csect-name
>   The name of the control section that issued the message.

column-name
>   The name of the XML column.

creator-name.*table-name*
>   The name of the table, qualified by the name of the creator of the table.

document-id
>   The hexadecimal document identifier.

System action: The auxiliary warning status is reset if no invalid columns remain.

User response: No action is required.

Severity: 4 (warning)

Related reference:

⇨ CHECK DATA (DB2 Utilities)

⇨ Auxiliary warning status (DB2 Utilities)

DSNU830I   *csect-name* - **INDEX=** *index-name* **IS IN REBUILD PENDING STATE**

Explanation: If the RECOVER utility issued this message, the corresponding table space to this index was recovered to a point in time with keyword TOCOPY, TOLOGPOINT, or TORBA. Thus, the index was placed in rebuild pending status.

If the REBUILD INDEX utility issued this message, the index is in rebuild pending status. Either duplicate index entries existed during the BUILD phase of the REBUILD INDEX, or invalid KEY/RIDs, which do not belong to the partition of the cluster index, were detected. The duplicate index entries or invalid KEY/RID entries must be resolved before starting the REBUILD INDEX utility again to rebuild the index and remove the index from rebuild pending status.

If the REORG utility with SHRLEVEL REFERENCE or CHANGE (or -TERM UTILITY) issued this message, the index is in rebuild pending status. During the renaming of a data set during REORG, either a data set with the source name did not yet exist or a data set with the target name already existed.

User response: If the RECOVER or REORG utility (or -TERM UTILITY) issued this message, use the REBUILD INDEX utility to rebuild this index.

If the REBUILD INDEX utility issued this message resolve the duplicate index entries or invalid keys and rerun the REBUILD INDEX utility.

Severity: 4 (warning) or 8 (error)

DSNU831I   *csect-name* **ALL INDEXES OF TABLESPACE=** *table space-name* **ARE IN REBUILD PENDING**

Explanation: The RECOVER TORBA, TOLOGPOINT, or TOCOPY utility issues this message to inform the user of the state of all indexes belonging to the recovered table space. All indexes were put in rebuild pending status to prevent data and index inconsistency. All indexes should be rebuilt before they are used.

System action: The RECOVER utility completed successfully.

User response: Use the REBUILD utility with the ALL option to rebuild all indexes over the recovered table space.

Severity: 4 (warning)

DSNU832I   *csect-name* **- INDEX=** *creator.index-name* **PART=** *partition-number* **IS IN REBUILD PENDING STATE**

Explanation: If the RECOVER utility issued this message, one partition of a partitioned table space is being recovered to a point in time with keyword TOCOPY, TOLOGPOINT, or TORBA. The corresponding index for that partition was placed in rebuild pending status.

If the REBUILD INDEX utility issued this message, the index or the partition of a cluster index is in rebuild pending status. Either duplicate index entries existed during the BUILD phase of the REBUILD INDEX or INDEX PART, or invalid KEY/RIDs, which do not belong to the partition of the cluster index, were detected. The duplicate index entries or invalid KEY/RID entries must be resolved before starting the REBUILD INDEX utility again to rebuild the index and remove the index from rebuild pending status.

If the REORG utility with SHRLEVEL REFERENCE or CHANGE (or -TERM UTILITY) issued this message, one partition of a partitioned index was placed in rebuild pending status. During the renaming of a data set during REORG, either a data set with the source name did not yet exist or a data set with the target name already existed.

User response: If the RECOVER or REORG utility (or -TERM UTILITY) issued the message, rebuild this index with the REBUILD INDEX utility.

If the REBUILD INDEX utility issued this message resolve duplicate index entries or invalid keys and rerun the REBUILD INDEX utility.

Severity: 4 (warning) or 8 (error)

DSNU833I   *csect-name* **- UNABLE TO RECOVER** *object-type object-name* **DUE TO ALTER COLUMN**

Explanation: RECOVER utility is unable to recover the index or index space as requested because an ALTER COLUMN VARCHAR has been performed. The current column definition is no longer consistent with the image copy and log records which would be used to perform the requested recovery.

User response: Use the REBUILD INDEX utility to rebuild the index from the tablespace.

Severity: 8 (error)

DSNU834I   *csect-name* **- UNABLE TO RECOVER** *object-type object-name* **DSNUM** *dsnum* **DUE TO ALTER COLUMN**

Explanation: RECOVER utility is unable to recover the index or index space partition as requested because an ALTER COLUMN VARCHAR has been performed. The current column definition is no longer consistent with the image copy and log records which would be used to perform the requested recovery.

User response: Use the REBUILD INDEX utility to rebuild the index partition from the tablespace.

Severity: 8 (error)

DSNU840I   *csect-name* **BASE TABLE SPACE** *database-name*.*space-name* **IS MISSING IN TABLE SPACE LIST FOR XML TABLE SPACE** *database-name*.*space-name*. **BASE TABLE=***creator-name*.*table-name*, **XML COLUMN=***column-name*

Explanation: An XML table space is specified by the INCLUDE XML TABLESPACES clause of the CHECK DATA utility, but no corresponding base table space is specified.

csect-name
    The name of the control section that issued the message.

database-name.space-name
> The name of the table space, qualified by the name of the database. This information is provided first for the base table space and then for the XML table space.

creator-name.table-name
> The name of the base table, qualified by the name of the creator of the table.

column-name
> The name of the XML column.

System action: Processing continues, but the indicated XML table space is not checked.

System programmer response: Either add the missing base table space to the TABLESPACE keyword or remove the XML table space from the INCLUDE XML TABLESPACES list.

Severity: 4 (warning)

Related reference:

➡ CHECK DATA (DB2 Utilities)

DSNU841I   *csect-name* **XML TABLE SPACE** *database-name.space-name* **IS NOT CHECK PENDING. BASE TABLE=***creator-name.table-name***, XML COLUMN=***column-name*

Explanation: During processing with the INCLUDE XML TABLESPACES and SCOPE PENDING options, the CHECK DATA utility determined that an XML table space is not in a check pending state.

csect-name
> The name of the control section that issued the message.

database-name.space-name
> The name of the XML table space, qualified by the name of the database.

creator-name.table-name
> The name of the base table, qualified by the name of the creator of the table.

column-name
> The name of the XML column.

System action: The utility skips the XML table space and continues processing.

System programmer response: No action is required.

Severity: 4 (warning)

Related reference:

➡ CHECK DATA (DB2 Utilities)

DSNU842I   *csect-name* **CHECKING XML TABLE SPACE** *database-name.space-name* **OF XML COLUMN** *column-name*

Explanation: The CHECK DATA utility is checking the XML table space for structurally defective documents.

csect-name
> The name of the control section that issued the message.

database-name.space-name
> The name of the XML table space that is being checked, qualified by the name of the database.

column-name
> The name of the XML column.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

CHECK DATA (DB2 Utilities)

DSNU843I *csect-name* **CHECKING XML TABLE SPACE** *database-name.space-name* **OF XML COLUMN** *column-name*
WAS COMPLETED SUCCESSFULLY

Explanation: The CHECK DATA utility has successfully completed the check of the XML table space for structurally defective documents.

csect-name
> The name of the control section that issued the message.

database-name.space-name
> The name of the XML table space that was checked, qualified by the name of the database.

column-name
> The name of the XML column.

System action: Processing continues with the next XML table space.

User response: No action is required.

Severity: 0

Related reference:

CHECK DATA (DB2 Utilities)

DSNU844I *csect-name* **CORRUPTED XML DOCUMENT IN XML TABLE SPACE** *database-name.space-name* **FOR**
XML COLUMN *column-name* **FOUND. DOCID** *document-id*

Explanation: The CHECK DATA utility found a corrupted XML document.

csect-name
> The name of the control section that issued the message.

database-name.space-name
> The name of the XML table space that was checked, qualified by the name of the database.

column-name
> The name of the XML column.

document-id
> The hexadecimal document identifier.

System action: The system action depends on options that were specified for the CHECK DATA utility:

SHRLEVEL REFERENCE and either AUXERROR REPORT or XMLERROR REPORT
> This error is reported, and the base table space is set to an auxiliary check pending state if CHECK DATA is configured to place table spaces in a check pending state (subsystem parameter CHECK_SETCHKP=YES).

SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or XMLERROR INVALIDATE
> The corrupted XML document is deleted from the XML table space, the XML column is set to an invalid state, and the base table space is set to an auxiliary warning state.

User response: Take the appropriate action for the CHECK DATA option that was specified:

SHRLEVEL REFERENCE and either AUXERROR REPORT or XMLERROR REPORT
> Run the CHECK DATA utility with either the AUXERROR INVALIDATE option or the XMLERROR INVALIDATE option to delete the invalid XML document and set the base table XML column to an invalid state.

SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or XMLERROR INVALIDATE
> Either update the XML column or delete the base table row. Then run the CHECK DATA utility with the AUXERROR INVALIDATE option. If all XML documents are structurally valid, the auxiliary warning status will be reset.

Severity: 4 (warning)

Related reference:

⮕ CHECK DATA (DB2 Utilities)

DSNU845I   *csect-name* **XML DOCUMENT IS MISSING FOR XML COLUMN** *column-name* **IN BASE TABLE**
creator-name.*table-name*. **DOCID** *document-id*

Explanation: A base table row contains an XML document, but the XML document is not found in the
corresponding XML table.

csect-name
> The name of the control section that issued the message.

column-name
> The name of the XML column.

creator-name.*table-name*
> The name of the base table, qualified by the name of the creator of the table.

document-id
> The hexadecimal document identifier.

System action: The system action depends on the SHRLEVEL option that was specified:

SHRLEVEL REFERENCE and either AUXERROR REPORT or XMLERROR REPORT
> This error is reported, and the base table space is set to an auxiliary check pending state if CHECK DATA is
> configured to place table spaces in a check pending state (subsystem parameter CHECK_SETCHKP=YES).

SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or XMLERROR INVALIDATE
> The XML column is set to an invalid state, and the base table space is set to an auxiliary warning state.

Programmer response: Take the appropriate action for the SHRLEVEL option that was specified:

SHRLEVEL REFERENCE and either AUXERROR REPORT or XMLERROR REPORT
> Run the CHECK DATA utility with the SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or
> XMLERROR INVALIDATE options to set the base table XML column to an invalid state.

SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or XMLERROR INVALIDATE
> Either update the XML column or delete the base table row. Then run the CHECK DATA utility with the
> SHRLEVEL REFERENCE and either AUXERROR INVALIDATE or XMLERROR INVALIDATE options. If all
> XML errors were corrected, the auxiliary warning status will be reset.

Severity: 4 (warning)

Related concepts:

⮕ XML schema validation with an XML type modifier (DB2 Programming for XML)

Related reference:

⮕ CHECK DATA (DB2 Utilities)

DSNU846I   *csect-name* **XML DOCUMENTS DELETED FROM TABLE** *creator-name*.*table-name* **FOR XML COLUMN**
column-name

Explanation: This message is issued by the CHECK DATA utility when XML documents have been deleted because
of XML schema validation errors.

csect-name
> The name of the control section that issued the message.

creator-name.*table-name*
> The name of the table, qualified by the name of the creator of the table.

column-name
> The name of the column.

DSNU848I • DSNU849I

The number of XML documents deleted from the table is equal to the number of XML documents inserted into the corresponding XML exception table. Message DSNU848I provides the name of the corresponding XML exception table. XML documents of one XML column will copied to their corresponding XML exception table.

System action: Processing continues.

System programmer response: No action is required.

Severity: 4 (warning)

Related reference:

➦ CHECK DATA (DB2 Utilities)

Related information:
"DSNU848I"

DSNU848I   *csect-name* **XML EXCEPTION TABLE** *creator-name.table-name* **CREATED FOR XML COLUMN**
column-name **IN BASE TABLE** *creator-name.table-name*

Explanation: When running with SHRLEVEL REFERENCE and AUXERROR/XMLERROR INVALIDATE, or with SCOPE XMLSCHEMAONLY, CHECK DATA performs XML schema validation against XML documents of XML columns having set an XML type modifier. In case of an XML schema validation error, CHECK DATA implicitly creates an XML exception table (XET), which contains a copy of the affected XML document. Each XML column will have its own corresponding XET object.

csect-name
> The name of the control section that issued the message.

creator-name.table-name
> The name of the table, qualified by the name of the creator of the table. This information is provided for the exception table and for the base table.

column-name
> The name of the column.

System action: Processing continues.

System programmer response: No action is required.

Severity: 0

DSNU849I   *csect-name* **XML DOCUMENT WITH DOCID** *document-id* **IS NOT VALID AGAINST THE XML TYPE MODIFIER OF XML COLUMN** *column-name* **FOR TABLE** *creator-name.table-name*

Explanation: The CHECK DATA utility found an XML document that is not valid for the XML type modifier of an XML column.

csect-name
> The name of the control section that issued the message.

document-id
> The hexadecimal document identifier.

column-name
> The name of the XML column.

creator-name.table-name
> The name of the table, qualified by the name of the creator of the table.

System action: The system action depends on options that were specified for the CHECK DATA utility:

AUXERROR REPORT
> This error is reported, and the base table space is set to an auxiliary check pending state if CHECK DATA is configured to place table spaces in a check pending state (subsystem parameter CHECK_SETCHKP=YES).

AUXERROR INVALIDATE
> The XML document that is in violation of the schema is copied to the XML exception table and is deleted from the XML table space. The base table XML column is set to an invalid state, and the base table space is set to an auxiliary warning state.

User response: Take the appropriate action for the CHECK DATA option that was specified:

AUXERROR REPORT
>Run the CHECK DATA utility with the AUXERROR INVALIDATE option to copy the XML document that is in violation to the XML exception table, delete it from the XML table space, and set the base table XML column to an invalid state.

AUXERROR INVALIDATE
>Either update the XML column with a document that is valid for the XML type modifier, or delete the base table row. Then run the CHECK DATA utility with the AUXERROR INVALIDATE option. If all XML documents are valid for their XML type modifiers, the auxiliary warning status will be reset.

Severity: 4 (warning)

Related concepts:

XML schema validation with an XML type modifier (DB2 Programming for XML)

Related reference:

CHECK DATA (DB2 Utilities)

DSNU850I   *csect-name* **INDEX** *creator-name*.*index-name* **SHOULD BE REBUILT**

Explanation: An inconsistent index was found.

csect-name
>The name of the control section that issued the message.

creator-name.*index-name*
>The name of the index, qualified by the name of the creator of the index.

System action: The status of the index is not changed.

User response: Run the REBUILD INDEX utility on the index.

Severity: 4 (warning)

Related reference:

REBUILD INDEX (DB2 Utilities)

DSNU851I   *csect-name* **RECORD** *record-id* **CANNOT BE LOCATED USING HASH ACCESS ON TABLE** *table-name*, **REASON CODE:** *reason-code*

Explanation: The CHECK DATA utility identified a problem with data in a table that is organized by hash. The indicated record could not be found using hash access.

csect-name
>The name of the control section that issued the message.

record-id
>The record ID that could not be found.

table-name
>The name of the table.

reason-code
>A numeric value that indicates the reason for the failure.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

Related concepts:

DB2 reason codes (DB2 Codes)

Related reference:

CHECK DATA (DB2 Utilities)

DSNU852I   *csect-name* **RECORD** *record-id* **FOR PARTITION** *partition-number* **CANNOT BE LOCATED USING**
HASH ACCESS ON TABLE *table-name*, **REASON CODE:** *reason-code*

Explanation: The CHECK DATA utility identified a problem with hash data. The indicated record could not be
found using hash access.

csect-name
        The name of the control section that issued the message.

record-id
        The record ID that could not be found.

partition-number
        The partition number.

table-name
        The name of the table.

reason-code
        A numeric value that indicates the reason for the failure.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

DSNU853I   *csect-name* **THE LOCATED RECORD** *record-id* **HAS A DIFFERENT RID OR KEY THAN THE**
UNLOADED ROW, REASON CODE: *reason-code*

Explanation: The CHECK DATA utility identified a problem with hash data. The unloaded record has a different
record ID (RID) or key than the record that was found using hash access.

csect-name
        The name of the control section that issued the message.

record-id
        The record ID that could not be found.

reason-code
        A numeric value that indicates the reason for the failure.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

Related concepts:

DB2 reason codes (DB2 Codes)

Related reference:

CHECK DATA (DB2 Utilities)

DSNU860I   *csect-name* **DIAGNOSE UTILITY COMPLETE**

Explanation: This message is issued in response to the normal completion of the DIAGNOSE utility.

System action: This message indicates the normal completion of the DIAGNOSE utility. Processing continues to the
next utility statement in the utility job step.

Severity: 0 (informational)

DSNU861I   DISPLAY MEPL FOR SYSTEM *name*

Explanation: This is the header message for output of the DIAGNOSE DISPLAY MEPL function. The contents of the MEPL is dumped to SYSPRINT.

The contents of the MEPL for subsystem *name* is dumped to SYSPRINT.

System action: This message indicates the beginning of DIAGNOSE DISPLAY MEPL processing.

Severity: 0 (informational)

DSNU862I   DISPLAY AVAILABLE UTILITIES. MAP: *bit-map*

Explanation: This message is the end-result of a DIAGNOSE DISPLAY AVAILABLE control statement. It displays the utilities that are available on this subsystem. The message contains a machine-readable bit-map, and precedes a table of utility names. They are two representations of the same information. An available utility is represented by the utility name that appears in the table and a "1" in the corresponding bit position in the bit-map. An unavailable utility is represented by a blank table entry and a "0" in the corresponding bit position in the bit-map.

Utilities may be unavailable if separately-priced features JDB771K and/or JDB771M are not installed, or installed improperly.

The table position and bit position are fixed for any given utility, and maintenance cannot alter them. Bit positions correspond to table entries when you read the table across each row, left to right, starting with the top row. For example, bit 1 corresponds to CATMAINT, bit 2 to CHECK, and bit 9 to OPTIONS.

The following example shows the partial output of this message. Due to space constraints, the sample has been truncated and broken into two rows. The actual table contains eight columns of identity names, and will display all eight columns horizontally across:

```
DSNU862I  DSNUDIAG - DISPLAY AVAILABLE UTILITIES.
  MAP: 111111111111111111111110000000000000
-------------------------------------------------
|CATMAINT |CHECK    |COPY     |DIAGNOSE |LISTDEF|
|OPTIONS  |QUIESCE  |REBUILD  |RECOVER  |REORG  |
|STOSPACE |TEMPLATE |UNLOAD   |COPYTOCOP|EXEC   |
|         |         |         |         |       |
-------------------------------------------------


-------------------------------------------------
|LOAD     |MERGE    |COPY     |MODIFY   |       |
|REPAIR   |REPORT   |RUNSTATS |         |       |
|BACKUP   |RESTORE  |         |         |       |
|         |         |         |         |       |
-------------------------------------------------
```

System action: The DIAGNOSE DISPLAY AVAILABLE utility control statement has completed normally.

Severity: 0 (informational)

DSNU863I   *csect-name* DISPLAY OBD *dbname***.***tsname*

Explanation: This message is the heading message for DIAGNOSE DISPLAY OBD output.

System action: This message indicates the start of the output for the DIAGNOSE DISPLAY OBD function.

Severity: 0 (informational)

DSNU864I   *csect-name* DISPLAY OBD, DBID=*nn* **OBID=***nn* **OBD TYPE=***type*

Explanation: This message is the heading message for the OBD being displayed.

System action: This message identifies the OBD being displayed.

Severity: 0 (informational)

DSNU865I   *csect-name* **DISPLAY OBD** *dbname.tsname* **COMPLETE**

Explanation: This message is issued in response to normal completion of DISPLAY OBD processing.

System action: This message indicates the end of DISPLAY OBD processing.

Severity: 0 (informational)

DSNU866I   *csect-name* **DISPLAY SYSUTIL JOB INFORMATION**

Explanation: This message is the header message for the SYSUTIL JOB INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output. This section displays the job information area of the SYSUTIL record and contains information which identifies the utility the SYSUTIL record is referencing.

System action: This message identifies the SYSUTIL JOB INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output.

Severity: 0 (informational)

DSNU867I   *csect-name* **DISPLAY SYSUTIL CHECKPOINT/RESTART INFORMATION**

Explanation: This message is the header message for the SYSUTIL UTILITY CHECKPOINT/RESTART INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output. This section displays the utility checkpoint/restart area of the SYSUTIL record and contains information pertaining to utility processing needed for the restart of the utility.

System action: This message identifies the SYSUTIL UTILITY CHECKPOINT/RESTART INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output.

Severity: 0 (informational)

DSNU868I   *csect-name* **DISPLAY SYSUTIL UTILITY DEPENDENT AREA**

Explanation: This message is the header message for the SYSUTIL UTILITY DEPENDENT INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output. This section displays the utility dependent area of the SYSUTIL record and contains information particular to the utility which the SYSUTIL record is referencing.

System action: This message identifies the SYSUTIL UTILITY DEPENDENT INFORMATION section of the DIAGNOSE DISPLAY SYSUTIL output.

Severity: 0 (informational)

DSNU869I   *csect-name* **DISPLAY SYSUTIL COMPLETE**

Explanation: This message is issued in response to normal completion of DISPLAY SYSUTIL processing.

System action: This message indicates the end of DISPLAY SYSUTIL processing.

Severity: 0 (informational)

DSNU870I   *csect-name* **NO SYSUTIL RECORD FOUND**

Explanation: This message is issued if no SYSUTIL records are found when DISPLAY SYSUTIL is specified.

System action: This message indicates the end of DISPLAY SYSUTIL processing.

Severity: 4 (warning)

DSNU871I   *csectname* **DISPLAY DBET DATABASE** *dbname*

Explanation: This message is the heading for display output of the database exception table (DBET) entry associated with database *dbname*.

System action: DIAGNOSE DISPLAY DBET processing started for the named database.

Severity: 0 (informational)

DSNU872I   *csectname* **DISPLAY DBET TABLE SPACE** *tsname*

Explanation: This message is the heading for display output of the database exception table (DBET) entry associated with table space *tsname*.

System action: DIAGNOSE DISPLAY DBET processing started for the named table space.

Severity: 0 (informational)

DSNU873I   *csectname* **DISPLAY DBET INDEX** *index-name*

Explanation: This message is the heading for display output of the database exception table (DBET) entry associated with index *index-name*.

System action: DIAGNOSE DISPLAY DBET processing started for the named index.

Severity: 0 (informational)

DSNU874I   *csectname* **NO DBET ENTRY FOUND FOR THE SPECIFIED OBJECT**

Explanation: There is no database exception table (DBET) entry associated with the specified object.

System action: Processing continues normally.

Severity: 4 (warning)

DSNU875I   *csectname* **DIAGNOSE WAIT AT MESSAGE=**msgno **INSTANCE=**nn **CORR ID=**id1 **CONN ID=**id2 **REPLY TO CONT**

Explanation: The DIAGNOSE WAIT function was invoked to wait at a specified message 'msgno' on the nn'th time that message was issued. The CORR ID is the correlation identifier of the thread associated with the waiting application. The CONN ID is the connection identifier of the thread.

System action: Processing is stopped until the user responds to this message.

Operator response: After diagnosis procedure in progress is completed, reply to this message to continue execution. Any reply is valid.

User response: See operator response.

Severity: 4 (warning)

DSNU876I   *csectname* **DIAGNOSE WAIT AT TRACEID=X'trcid' INSTANCE=**nn **CORR ID=**id1 **CONN ID=**id2 **REPLY TO CONT**

Explanation: The DIAGNOSE WAIT function was invoked to wait at a specified traceid 'trcid' on the nn'th time that traceid was encountered. The CORR ID is the correlation identifier of the thread associated with the waiting application. The CONN ID is the connection identifier of the thread.

System action: Processing is stopped until the user responds to this message.

Operator response: After diagnosis procedure in progress is complete, reply to this message to continue execution. Any reply is valid.

User response: See operator response.

Severity: 4 (warning)

DSNU877I   **event modulename+offset**

Explanation: Event 'event' occurred at module 'modulename' plus 'offset'. The DIAGNOSE utility issues this message to aid in diagnosing utility problems.

System action: Processing continues normally.

Severity: 0 (informational)

DSNU900I *csect-name* **- DBD X'xxxx' COMPARISON STARTING**

Explanation: This message marks the start of a DBD comparison for the REPAIR DBD TEST or REPAIR DBD DIAGNOSE utility. The DBID of the DBD being compared against the DB2 Catalog is 'xxxx'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU901I *csect-name* **- OBD X'xxxx'** *obd-type* **HAS THE FOLLOWING DIFFERENCES**

Explanation: This message marks the start of an OBD comparison for the REPAIR DBD TEST or REPAIR DBD DIAGNOSE utility. This message is displayed when a difference in the OBDs has been detected. The OBID of the OBD being compared against the DB2 catalog is 'xxxx'. The type of OBD being displayed is 'obd-type'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU902I *csect-name* **- EXISTING OBD AT OFFSET X'xxxxxxxx' IN EXISTING DBD**

Explanation: Following this message is a hexadecimal dump of the OBD as it exists in the DSNDB01.DBD01 page set. In the message, 'xxxxxxxx' is the hexadecimal offset from the beginning of the existing DBD as it appears later in the REPAIR DBD DIAGNOSE output.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU903I *csect-name* **- REBUILT OBD AT OFFSET X'xxxxxxxx' IN REBUILT DBD**

Explanation: Following this message is a hexadecimal dump of the OBD that was rebuilt from the DB2 Catalog. In the message, 'xxxxxxxx' is the hexadecimal offset from the beginning of the rebuilt DBD as it appears later in the REPAIR DBD DIAGNOSE output.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU904I *csect-name* **- EXISTING VALUE X'xx..xx'**

Explanation: This message shows a hexadecimal string which is the value of a field in the OBD as it exists on the DSNDB01.DBD01 page set. The hexadecimal string is 'xx..xx'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU905I *csect-name* **- REBUILT VALUE X'xx..xx'**

Explanation: This message shows a hexadecimal string which is the value of a field in the OBD which was rebuilt from the DB2 catalog. The hexadecimal string is 'xx..xx'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU906I *csect-name* **- EXISTING VALUE 'cc..cc'**

Explanation: This message shows a character string which is the value of a field in the OBD as it exists on the DSNDB01.DBD01 page set. The character string is 'cc..cc'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU907I   *csect-name* **- REBUILT VALUE 'cc..cc'**

Explanation: This message shows a character string which is the value of a field in the OBD which was rebuilt from the DB2 catalog. The character string is 'cc..cc'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU908I   *csect-name* **- EXISTING OBD IS** *obd-type*

Explanation: This message shows the OBD type of the OBD which exists on the DSNDB01.DBD01 page set.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU909I   *csect-name* **- REBUILT OBD IS** *obd-type*

Explanation: This message shows the OBD type of the OBD which was rebuilt from the DB2 catalog.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU910I   *csect-name* **- EXISTING DBD**

Explanation: Following this message is a hexadecimal dump of the DBD which exists on the DSNDB01.DBD01 page set.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU911I   *csect-name* **- REBUILT DBD**

Explanation: Following this message is a hexadecimal dump of the DBD which was rebuilt from the DB2 Catalog.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU912I   *csect-name* - *phase-completion-message* **--** *nn* **ERRORS WERE FOUND**

Explanation: This message marks the end of a processing phase in the DBD comparison for a REPAIR DBD TEST or REPAIR DBD DIAGNOSE utility. In the message, 'nn' indicates the number of errors detected while processing. Informational and warning messages are not added into the error count.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU913I   *csect-name* **- OBD X'xxxx' COMPARISON COMPLETE --** *nn* **ERRORS WERE FOUND**

Explanation: This message marks the end of an OBD comparison for the REPAIR DBD TEST or REPAIR DBD DIAGNOSE utility. This message is displayed only when a difference has been detected. If there is no difference, this message will not be displayed. The OBID of the OBD being compared against the DB2 catalog is 'xxxx'. The number of errors in this OBD is 'nn'. Informational and warning messages are not added into the error count.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU914I *csect-name* **- DBD X'xxxx' COMPARISON COMPLETE --** *nn* **ERRORS WERE FOUND**

Explanation: This message marks the end of a DBD comparison for the REPAIR DBD TEST or REPAIR DBD DIAGNOSE utility. The DBID of the DBD being compared against the DB2 catalog is 'xxxx'. The total number of errors found in this DBD is 'nn'.

System action: DBD checking completed.

Severity: 0 (informational)

DSNU915I *csect-name* **- OBD X'obid' E-X'xxxxxxxx' R-X'yyyyyyyy'** *obd-description*

Explanation: This message is used when listing the OBDs in the REPAIR DBD DIAGNOSE utility. The OBID of the OBD being listed is 'obid'. The offset to the OBD within the EXISTING DBD which is displayed later in the output is 'xxxxxxxx'. The offset to the OBD within the REBUILT DBD which is displayed later in the output is 'yyyyyyyy'. A short description of the state of the OBD is 'obd-description'.

System action: DBD checking completed.

Severity: 0 (informational)

DSNU916I *csect-name* **- OBDS DIFFER AT OFFSET X'xxxxxxxx'** *field-name* **- INFORMATION NOT IN DB2 CATALOG**

Explanation: This message indicates the value of a field in the OBD differs from the DBD which exists on the DSNDB01.DBD01 page set and the DBD rebuilt from the DB2 catalog. This is information which is not kept in the DB2 catalog. This is an acceptable difference and does not indicate that the OBD is damaged. The hexadecimal offset from the beginning of the OBD is 'xxxxxxxx'. The name of the field is 'field-name'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU917I *csect-name* **- DBDS DIFFER AT OFFSET X'xxxxxxxx'** *field-name* **- INFORMATION NOT IN DB2 CATALOG**

Explanation: This message indicates that the value of a field in the DBD differs from the DBD which exists on the DSNDB01.DBD01 page set and the DBD rebuilt from the DB2 Catalog. This is information which is not kept in the DB2 catalog. This is an acceptable difference and does not indicate that the DBD is damaged. The hexadecimal offset from the beginning of the DBD is 'xxxxxxxx'. The name of the field is 'field-name'.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU919I *csect-name* **- DBD X'xxxx' OBD X'yyyy' -** *details*

Explanation: This message gives detailed information about the specified OBD. 'xxxx' is the DBID of the DBD. 'yyyy' is the OBID of the OBD or the PSID of the OBD when the *details* indicates a PSID is being processed. *details* provides additional information.

System action: DBD checking continues normally.

Severity: 0 (informational)

DSNU920I *csect-name* **- OBDS DIFFER AT OFFSET X'xxxxxxxx'** *field-name*

Explanation: This message indicates that the value of a field in the OBD differs from the DBD which exists on the DSNDB01.DBD01 page set and the DBD rebuilt from the DB2 catalog. The DB2 Catalog and DBD/OBD are inconsistent. The hexadecimal offset from the beginning of the OBD is 'xxxxxxxx'. The name of the field is 'field-name'.

System action: DBD checking continues normally.

Severity: 4 (warning)

DSNU921I   *csect-name* **- OBD X'xxxx' MISSING IN EXISTING DBD**

Explanation: This message indicates that an OBD which was rebuilt from the DB2 catalog does not exist in the DBD which exists on the DSNDB01.DBD01 page set. The DB2 catalog and DBD/OBD are inconsistent. The OBID of the missing OBD is 'xxxx'.

System action: DBD checking continues normally.

Severity: 4 (warning)

DSNU922I   *csect-name* **- OBD X'xxxx' MISSING IN REBUILT DBD**

Explanation: This message indicates that an OBD in the DBD which exists on the DSNDB01.DBD01 page set does not exist in the DBD which was rebuilt from the DB2 catalog. The DB2 catalog and DBD/OBD are inconsistent. The OBID of the missing OBD is 'xxxx'.

System action: DBD checking continues normally.

Severity: 4 (warning)

DSNU923I   *csect-name***-OBDTYPE FOR OBD X'xxxx' DOES NOT MATCH -NO FURTHER CHECKING DONE FOR THIS OBD**

Explanation: The object descriptor (OBD) that exists on the DSNDB01.DBD01 page set is different from the type of OBD that was rebuilt from the DB2 catalog. No further processing can be done because the fields of one OBD type are different from the fields of another. The DB2 catalog and database descriptor (DBD)/OBD are inconsistent. The object identifier (OBID) of the OBD that is inconsistent is 'xxxx'.

System action: Checking for OBD X'xxxx' stops. Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU924I   *csect-name***-DBDS DIFFER AT OFFSET X'xxxxxxxx'** *field-name*

Explanation: The value of a field in the database descriptor (DBD) differs from the value of the corresponding field in the DBD that exists on the DSNDB01.DBD01 page set and differs from the value of the corresponding field in the DBD that was rebuilt from the DB2 catalog.

xxxxxxxx
    The hex offset from the beginning of the DBD.

field-name
    The name of the field.

System action: Checking of the DBD continues normally.

Severity: 0 (informational)

DSNU925I   *csect-name***-OBDDFILE AT OFFSET X'xxxxxxxx' IN EXISTING DBD HAS OBID X'nnnn' WHICH IS NOT A FILE OBD**

Explanation: The field that points to the first FILE OBD (OBDDFILE) in the database descriptor (DBD) that exists on the DSNDB01.DBD01 page set does not point to a FILE OBD. The DB2 catalog and DBD/OBD (object descriptor) are inconsistent.

xxxxxxxx
    The hex offset from the beginning of the existing DBD.

nnnn   The object identifier (OBID) of the OBD which should be a FILE OBD.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU926I *csect-name*-**OBDDFILE AT OFFSET X'xxxxxxxx' IN REBUILT DBD HAS OBID X'nnnn' WHICH IS NOT A FILE OBD**

Explanation: The field that points to the first FILE OBD (OBDDFILE) in the database descriptor (DBD) that was rebuilt from the DB2 catalog does not point to a FILE OBD. The DB2 catalog and DBD/OBD (object descriptor) are inconsistent.

xxxxxxxx
> The hex offset from the beginning of the rebuilt DBD.

nnnn The object identifier (OBID) of the OBD that should be a FILE OBD.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU927I *csect-name*-**OBD X'xxxx' IS A HOLE IN EXISTING DBD**

Explanation: The object descriptor (OBD) that exists on the DSNDB01.DBD01 page set has been logically deleted, OBDHOLE = '1'B. The DB2 catalog and database descriptor (DBD)/OBD are inconsistent. The object identifier (OBID) of the OBD that is a hole is 'xxxx'.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU928I *csect-name*-**OBD X'xxxx' IS A HOLE IN REBUILT DBD**

Explanation: The object descriptor (OBD) that was rebuilt from the DB2 catalog has been logically deleted, OBDHOLE = '1'B. The DB2 catalog and database descriptor (DBD)/OBD are inconsistent. The object identifier (OBID) of the OBD that is a hole is 'xxxx'.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU929I *csect-name*-**OBD FIELD** *field-name* **HAS VALUE OUT OF RANGE AT OFFSET X'xxxxxxxx' IN OBD X'nnnn'**

Explanation: The object descriptor (OBD) field 'field-name' has an unexpected value. The DB2 catalog and database descriptor (DBD)/OBD are inconsistent.

field-name
> The name of the field in the OBD.

xxxxxxxx
> The hex offset from the beginning of the OBD.

nnnn The object identifier (OBID) of the OBD.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU930I *csect-name*- **INDEX** *index-name* **(OBD X'xxxx') MUST BE REBUILT**

Explanation: The index 'index-name' must be either recovered or dropped and re-created to ensure that the index is consistent with the description in the INDEX FANSET OBD. The object descriptor (OBD) is not damaged. However, if the index is not either recovered or dropped and re-created, incorrect output can be produced when the index is used as an access path.

index-name
> The name of the index found in SYSIBM.SYSINDEXES.

xxxx The object identifier (OBID) of the OBD.

System action: Checking of the DBD continues normally.

Severity: 8 (error)

DSNU931I   *csect-name* **- OBDDMAX FIELD OF DBD IS TOO LARGE**

Explanation: The OBDDMAX field (maximum number of ID in the DBD) is too large. It is larger than the largest possible number of ID based upon the DBD size.

System action: Checking of the DBD continues but using the largest possible number of ID instead of the incorrect OBDDMAX.

Severity: 8

DSNU932I   **REPAIR CAN NOT DROP THE DATABASE.**

Explanation: The system cannot drop the database because the name that was specified for the database and the DBID that was specified do not match.

System action: The utility job is halted.

User response: Correct the utility statement, and resubmit the job.

DSNU933I   *csect-name* **- TABLESPACE** *tablespace-name* **MUST BE REORGANIZED**

Explanation: The specified table space was reorganized to materialize pending definition changes. After that REORG job was run, a RECOVER job was run to recover the data to a point in time before the REORG job ran. A subsequent REORG job must be run to complete the point-in-time recovery process.

tablespace-name
        The name of the table space.

System action: Utility processing is not initiated.

User response: Run REORG with SHRLEVEL REFERENCE on the entire table space to remove the REORG-pending (REORP) restrictive state, and to complete the point-in-time recovery process.

Severity: 8 (error)

Related reference:

➡ REORG TABLESPACE (DB2 Utilities)

DSNU935I   *csect-name* **DATABASE** *database-name* **IS NOT STARTED FOR UTILITY PROCESSING ONLY**

Explanation: This message is issued when the REPAIR utility with the DBD option is run on a database that is not started for utility processing only.

System action: REPAIR DBD terminated.

User response: Stop the database. Then start the database with the ACCESS(UT) option. Resubmit the REPAIR DBD job.

Severity: 8

DSNU936I   *csect-name* **DATABASE** *database-name* **IS IN USE**

Explanation: This message is issued when the REPAIR utility with the DBD option is run on a database that is in use by others.

System action: REPAIR DBD terminated.

User response: Wait until all users are finished, then resubmit the REPAIR DBD job.

Severity: 8

DSNU938I   *csect-name*-**OBD X'xxxx' HAD DIFFERENCES WHICH WERE NOT IN A NAMED FIELD**

Explanation: There is a difference between the object descriptor (OBD) in the database descriptor (DBD) that exists on the DSNDB01.DBD01 page set and the DBD that was rebuilt from the DB2 catalog. This difference does not exist in a named field. Therefore, the detailed search by field name fails to find the difference. The DB2 catalog and DBD/OBD are inconsistent. The object identifier (OBID) of the OBD in error is 'xxxx'.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU939I   *csect-name*-**DBD HEADER HAD DIFFERENCES WHICH WERE NOT IN A NAMED FIELD**

Explanation: There is a difference between the DBD header in the DBD that exists on the DSNDB01.DBD01 page set and the DBD that was rebuilt from the DB2 catalog. This difference does not exist in a named field. Therefore, the detailed search by field name fails to find the difference. The DB2 catalog and database descriptor (DBD)/object descriptor (OBD) are inconsistent.

System action: Checking of the DBD continues normally.

Severity: 4 (warning)

DSNU971I   *csect-name* **TABLESPACE** *database-name*.*tablespace-name* **IS CHECK PENDING OR AUX CHECK PENDING CORRELATION-ID** *correlation-identifier* **CONNECTION-ID** *connection-identifier* **LUW-ID** *luw-identifier*

Explanation: This message is sent to the console when TABLESPACE *tsname* is set to CHECK pending or auxiliary CHECK pending state. The CORRELATION-ID *correlation-identifier* is the correlation identifier of the thread requesting CHECK pending or auxiliary CHECK pending status. Correlation IDs beginning with '0*nn*', where *nn* is a number from 01 to 28, identify system agents. The CONNECTION-ID *connection-identifier* is the correlation identifier of the thread. The LUW-ID *luw-identifier* is the logical-unit-of-work identifier of the thread. If the LUW-ID is an '*', the thread has no LUW-ID. If it is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of logical-unit-of-work-id in any DB2 command that accepts logical-unit-of-work-id as input.

System action: Processing continues.

User response: Resolve the CHECK pending status.

Severity: 4 (warning)

Related concepts:

[→] Implications of adding parent or foreign keys (DB2 Administration Guide)

Related tasks:

[→] Collecting data for failures after a point-in-time recovery (Collecting data)

[→] Adding referential constraints to existing tables (DB2 Administration Guide)

Related reference:

[→] Advisory or restrictive states (DB2 Utilities)

[→] CHECK-pending status (DB2 Utilities)

Related information:

[→] Recovering from referential constraint violation (DB2 Administration Guide)

DSNU972I   *csect-name* **CATALOG UPDATING FAILURE WHILE SETTING OR RESETTING CHECK PENDING CORRELATION-ID** *id1* **CONNECTION-ID** *id2* **LUW ID=***luw-id*

Explanation: When the check pending state is set on or off, the updating of the catalog tables (SYSTABLESPACE, SYSTABLES, and SYSTABLEPART) fails. The CORRELATION-ID, *id1*, is the correlation identifier of the thread associated with the application requesting that the check pending state be set on or off. The CONNECTION-ID, *id2*, is the correlation identifier of the application. Correlation IDs beginning with '0nn', where 'nn' is a number from 01 to 28, identify system agents.

System action: Processing continues.

Problem determination: The utility being processed requested the setting of check pending either on or off. The updating of the STATUS column of SYSTABLESPACE, and of the CHECKFLAG and CHECKRID columns of SYSTABLES and SYSTABLEPART failed. Processing continues normally. However, those columns for the rows being updated now contain invalid values. Subsequent utility processing ignores these values, and the next setting or resetting of check pending for that table space corrects the error in one of the following ways:

- If the table space is set to check pending and the catalog fields are in error, the scope of check pending is escalated in the catalog. (The STATUS column of SYSTABLESPACE is set to 'P', and the CHECKFLAG and CHECKRID columns of SYSTABLES and SYSTABLEPART are set to 'C' and binary zeros.)

- If the table space is set to not check pending and the catalog fields are in error, the scope of check pending is set off in the catalog. (The STATUS column of SYSTABLESPACE is set to an 'A', and the CHECKFLAG and CHECKRID columns of SYSTABLES and SYSTABLEPART are set to blanks.)

After the catalog fields are corrected, the setting or resetting of check pending continues as specified.

Usually message DSNT500I or DSNT501I is also issued, because the probable cause of the error is RESOURCE UNAVAILABLE. If the reason for the failure was not RESOURCE UNAVAILABLE, an SVC dump is requested and processing continues normally. To determine the cause of the failure, refer to the dump listing.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- SYSPRINT output, including JCL, for the application program or batch job, and system messages that were issued. Make sure that MSGLEVEL=(1,1) on the JOB statement so that all diagnostic information is sent to SYSPRINT.

Severity: 0 (informational) or 4 (warning)

DSNU973I   *csect-name* **CHECK PENDING, AUX CHECK PENDING OR BOTH HAS BEEN RESET FOR TABLESPACE** *database-name***.***tablespace-name* **CORRELATION-ID** *correlation-identifier* **CONNECTION-ID** *connection-identifier* **LUW-ID** *luw-identifier*

Explanation: This message is sent to the console when CHECK pending, auxiliary CHECK pending or both states were reset for the *tablespace-name* table space. The CORRELATION-ID *correlation-identifier* is the correlation identifier of the thread requesting that the CHECK pending state, the auxiliary CHECK pending state or both states be reset. Correlation IDs beginning with '0nn', where *nn* is a number from 01 to 28, identify system agents. The CONNECTION-ID *connection-identifier* is the correlation identifier of the thread. The LUW-ID *luw-identifier* is the logical-unit-of-work identifier of the thread. If the LUW-ID is an '*', the thread has no LUW-ID. If it is not an '*', the value is a logical-unit-of-work-id that is assigned to the thread followed by an '=' sign and a token value, which can be used in place of the logical-unit-of-work-id in any DB2 command that accepts a logical-unit-of-work-id as input.

System action: Processing continues.

User response: Resolve the CHECK pending status.

Severity: 0 (informational)

DSNU993I • DSNU997I

Related concepts:

▶ Implications of adding parent or foreign keys (DB2 Administration Guide)

Related tasks:

▶ Collecting data for failures after a point-in-time recovery (Collecting data)

▶ Adding referential constraints to existing tables (DB2 Administration Guide)

Related reference:

▶ Advisory or restrictive states (DB2 Utilities)

▶ CHECK-pending status (DB2 Utilities)

Related information:

▶ Recovering from referential constraint violation (DB2 Administration Guide)

DSNU993I   *csect-name* **IMAGE COPY RESTORE TERMINATED**

Explanation: Catalog and directory initialization (image copy restore) did not complete successfully.

System action: Processing terminates.

User response: Examine other error messages associated with this job.

Severity: 8 (error)

DSNU994I   *csect-name* **IMAGE COPY RESTORE COMPLETED SUCCESSFULLY**

Explanation: Catalog and directory initialization (image copy restore) completed successfully.

Severity: 0 (informational)

DSNU995I   *csect-name* **VSAM PUT ERROR,RPLERREG =** *return-code*, **RPLERRCD =** *reason-code*

Explanation: VSAM PUT failed with the indicated *return-code* and *reason-code*.

System action: Processing terminates.

System programmer response: Follow standard VSAM diagnostic procedure. Refer to *z/OS DFSMS: Macro Instructions for Data Sets*. See also *MVS/DFP V3R3 Diagnosis Guide* for more information about the VSAM procedures.

User response: Notify system programmer.

Severity: 8 (error)

DSNU996I   *csect-name* **VSAM OPEN ERROR, ACBERRFLG =** *error-code*

Explanation: VSAM OPEN failed with the indicated ACB *error-code*.

System action: Processing terminates if the error code is greater than or equal to 128; otherwise, processing continues normally.

System programmer response: Follow standard VSAM diagnostic procedure. Refer to *z/OS DFSMS: Macro Instructions for Data Sets*. See also *MVS/DFP V3R3 Diagnosis Guide* for more information about the VSAM procedures.

User response: Notify the system programmer.

Severity: 8 (error) if error is greater than or equal to 128; otherwise, 4 (warning)

DSNU997I   *csect-name* **OUTPUT DATA SET NAME =** *data-set-name*

Explanation: The indicated *data-set-name* is used for output.

Severity: 0 (informational)

DSNU998I   *csect-name* **INPUT DATA SET NAME =** *data-set-name*

Explanation: The indicated *data-set-name* is used for input.

Severity: 0 (informational)

DSNU999I   *csect-name* **START OF IMAGE COPYRESTORE FOR JOB** *job-name step-name*

Explanation: The catalog and directory initialization (image copy restore) started.

Severity: 0 (informational)

DSNU1000I   *csect-name* **- PROCESSING CONTROL STATEMENTS IN PREVIEW MODE**

Explanation: The PREVIEW option has been specified either on the OPTIONS control statement or as a JCL parameter. Control statements will be previewed in this execution for TEMPLATE and LISTDEF use but the utility control statements will not be executed. PREVIEW via JCL parameter will continue to the end of the job step. PREVIEW via OPTIONS PREVIEW will continue until the end of the job step or another OPTIONS control statement.

System action: Utility control statements are previewed but not executed.

User response: If the return code is 8, PREVIEW detected errors which must be corrected. If the return code is 4, PREVIEW detected potential problems, but the previewed statements are valid.

Examine the preview output shown in informational messages DSNU1010I, DSNU1020I, DSNU1021I, DSNU1022I, DSNU1023I and DSNU1025I, plus any error messages. Correct any statements that are in error and remove the PREVIEW option to execute the control statements.

Severity: 0 (informational)

DSNU1001I   *csect-name* **- LISTDEF** *list-name* **CLAUSE IDENTIFIES NO NEW OBJECTS:** *clause-type object-type*
object-qualifier.object-name

Explanation: An INCLUDE or EXCLUDE clause of a LISTDEF control statement returned no new objects.

csect-name
> The name of the control section that issued the message.

list-name
> The name of the list.

clause-type
> The type of clause: INCLUDE or EXCLUDE.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

This message is caused by the following conditions:
- The clause evaluated to no objects at all.
- The clause did not return any objects that had not already been included or excluded by previous clauses.
- The clause contains only undefined objects. Prior to Version 10, undefined objects were included in the list. However, beginning with Version 10, undefined objects (objects that have been created with DEFINE NO) are, by default, not included in the list.

System action: Utility processing continues.

The total number of objects covered by the LISTDEF did not change as a result of this clause.

User response: No immediate response is required. To prevent this message in the future, update the clause.

To include undefined objects in the list, specify DEFINED ALL on the LISTDEF control statement.

Severity: 4 (warning)

Related reference:

➦ LISTDEF (DB2 Utilities)

DSNU1002I *csect-name* **- INVALID KEYWORD SPECIFIED ON CLAUSE:** *clause-type object-type qualifier.name*

Explanation: An INCLUDE or EXCLUDE clause of a LISTDEF control statement has an invalid keyword specified.

Catalog and directory objects may only be specified via their TABLESPACE or INDEXSPACE names. DATABASE, TABLE and INDEX may not be specified. Catalog and directory object names may not contain pattern matching characters. They must be included or excluded explicitly using their complete names. Those include or exclude clauses may not contain the keywords RI, BASS, LOB, ALL.

System action: Utility processing stops.

User response: Correct the invalid LISTDEF clause and rerun the utility.

Severity: 8 (error)

DSNU1003I *csect-name* **- INVALID LIST FOR** *utility-name* **UTILITY**

Explanation: The list name referenced on this control statement is not valid for the specified utility. The list either contains no objects or contains objects which cannot be processed by this specific utility.

System action: Utility processing is not initiated.

User response: Correct the LISTDEF in error, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➦ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➦ LISTDEF (DB2 Utilities)

DSNU1004I *csect-name* **- DUPLICATE** *object-type object-name* **FOUND IN SYSIN** *dd-name*

Explanation: Duplicate LISTDEF or TEMPLATE statements (duplicate names) were detected in the SYSIN, listdef, or template data set.

System action: Utility processing is not initiated.

User response: Eliminate the duplicate LISTDEF or TEMPLATE names and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➦ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➦ LISTDEF (DB2 Utilities)

➦ TEMPLATE (DB2 Utilities)

DSNU1005I *csect-name* **- UTILITY** *name* **NOT ALLOWED IN** *dd-type* **DD** *dd-name*

Explanation: The utility parser detected a control statement other than TEMPLATE in the SYSTEMPL data set or LISTDEF in the SYSLISTD data set.

System action: Utility processing is not initiated.

User response: Remove the invalid control statements and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➡ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➡ LISTDEF (DB2 Utilities)

➡ TEMPLATE (DB2 Utilities)

DSNU1006I *csect-name* **-** *object-type object-name* **NOT FOUND**

Explanation: A utility control statement used a LIST or TEMPLATE that was not found in SYSIN or either of the SYSTEMPL or SYSLISTD data sets. These DD names may vary if OPTIONS LISTDEFDD or OPTIONS TEMPLATEDD were specified.

System action: Utility processing is not initiated.

User response: Provide the missing LISTDEF or TEMPLATE definition in SYSIN prior to its use and resubmit the utility job. If the definition utility exists in another data set, provide the SYSTEMPL or SYSLISTD DD card for the data set.

Severity: 8 (error)

Related tasks:

➡ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➡ LISTDEF (DB2 Utilities)

➡ TEMPLATE (DB2 Utilities)

DSNU1007I *csect-name* **- DATE/TIME VALUES MAY CHANGE BEFORE EXECUTION**

Explanation: PREVIEW processing has detected DSN date/time variables on a TEMPLATE control statement being used. These values may change before the actual utility execution, resulting in different data set names.

System action: Preview processing continues normally.

Severity: 4 (warning)

DSNU1008I *csect-name* **- LISTDEF** *list-name* **CONTAINS NO OBJECTS**

Explanation: The indicated LISTDEF list definition expands to contain no objects.

csect-name
      The name of the control section that issued the message.

list-name
      The name of the list.

Prior to Version 10, undefined objects were included in the list. However, beginning with Version 10, undefined objects (objects that have been created with DEFINE NO) are, by default, not included in the list.

System action: Utility processing is not initiated.

User response: Correct the LISTDEF definition, and resubmit the utility job.

To include undefined objects in the list, specify DEFINED ALL on the LISTDEF control statement.

Severity: 4 (warning)

Related reference:

➡ LISTDEF (DB2 Utilities)

DSNU1009I   *csect-name* **- TEMPLATE** *name* **DSN=** *dataset-name*

Explanation: PREVIEW processing has expanded a referenced TEMPLATE data set name to the value shown. If the data set name contains date/time variables, the date/time values might change before actual utility execution.

System action: PREVIEW processing continues.

User response: If the data set names shown are not acceptable, correct the TEMPLATE control statements, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➦ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➦ LISTDEF (DB2 Utilities)

➦ TEMPLATE (DB2 Utilities)

DSNU1010I   *csect-name* **- LISTDEF** *name* **EXPANDS TO THE FOLLOWING OBJECTS:**

Explanation: This message is followed by an equivalent LISTDEF control statement with all pattern matching characters removed and all INCLUDE and EXCLUDE clauses evaluated. For static applications, this expanded LISTDEF may be saved to a data set and reused to avoid list expansion overhead.

System action: PREVIEW processing continues.

User response: Examine the list, correct the LISTDEF if necessary and resubmit the utility job.

Severity: 0 (informational)

Related tasks:

➦ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➦ LISTDEF (DB2 Utilities)

DSNU1011I   **WORKFILE NOT ALLOWED IN LISTDEF**

Explanation: A workfile database cannot be used in a LISTDEF list because utilities do not process them.

System action: Utility processing is not initiated.

User response: Correct the LISTDEF in error, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➦ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➦ LISTDEF (DB2 Utilities)

DSNU1012I   *csect-name* **- LISTDEF MUST BEGIN WITH AN INCLUDE CLAUSE**

Explanation: A LISTDEF control statement begins with something other than an INCLUDE clause. The LISTDEF control statement must begin with an INCLUDE clause.

System action: Utility processing is not initiated.

User response: Correct the LISTDEF in error, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➡ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➡ LISTDEF (DB2 Utilities)

DSNU1013I   *csect-name* **-** *dd-type* **DD CARD** *dd-name* **NOT FOUND**

Explanation: An OPTIONS control statement specifies a DD name '*dd-name*' that does not exist in the JCL.

System action: Utility processing is not initiated.

User response: Correct the DD name on the OPTIONS statement or include the missing JCL DD card, and resubmit the utility job.

Severity: 8 (error)

Related reference:

➡ OPTIONS (DB2 Utilities)

DSNU1014I   *csect-name* **- INDEXSPACE** *name* **NOT FOUND**

Explanation: The specified INDEXSPACE does not exist.

System action: Utility processing is not initiated.

User response: Correct the operand in error, and resubmit the utility job.

Severity: 8 (error)

Related concepts:

➡ Including objects in a list (DB2 Utilities)

➡ DB2 online utilities (DB2 Utilities)

Related tasks:

➡ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➡ LISTDEF (DB2 Utilities)

DSNU1015I   *csect-name* **- ERROR ALLOCATING DATA SET DSN=***name* **CODE=X'***reason-code***'**

Explanation: An error was returned by SVC 99 during TEMPLATE data set dynamic allocation. This message is followed by text which explains the reason-code shown.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, execution will HALT.

User response: Correct the error preventing dynamic allocation, and resubmit the utility job. In some cases, TEMPLATE dynamic allocation may not be possible and a JCL DD card may be required. See DYNALLOC (SVC 99) text unit documentation for limitations.

Severity: 8 (error)

DSNU1016I   *csect-name* **- TEMPLATE DSN VARIABLE** *var-name* **HAS NO VALUE**

Explanation: A utility has attempted to build a data set name based on a TEMPLATE that specifies DSN variables. The variable shown has no value in this particular use. For example, **PART** when the object is not partitioned or **ICTYPE** when an image copy is not being allocated.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, execution will HALT.

User response: Correct the TEMPLATE DSN specification to eliminate the invalid variable or use a different TEMPLATE, and resubmit the utility job.

Severity: 8 (error)

Related concepts:

Using the OPTIONS utility with LISTDEF (DB2 Utilities)

Related reference:

OPTIONS (DB2 Utilities)

TEMPLATE (DB2 Utilities)

DSNU1017I *csect-name* **- TEMPLATE DSN VARIABLE** *var-name* **TRUNCATED TO** *value*

Explanation: While performing dynamic allocation of a TEMPLATE data set the DSN variable shown was truncated to the value shown.

System action: Utility processing continues.

Severity: 4 (warning)

DSNU1018I *csect-name* **- USER** *user-id* **NOT AUTHORIZED TO ACCESS CATALOG TABLE** *qualifier*. *name*

Explanation: The user executing the utility does not have the required authority to access the catalog table shown. This access is required to expand the LISTDEF control statement being used.

System action: Utility processing is not initiated.

User response: GRANT the required authority to access the DB2 catalog to the user executing the utility.

Severity: 8 (error)

Related reference:

LISTDEF (DB2 Utilities)

GRANT (DB2 SQL)

DB2 catalog tables (DB2 SQL)

DSNU1019I *csect-name* **- INVALID TEMPLATE NAME** *template-name*

Explanation: A TEMPLATE control statement specifies an invalid TEMPLATE name. The names UTPRINT, SORTLIB and any name starting with SORTWK or SYS may not be used.

System action: Utility processing terminates.

User response: Correct the TEMPLATE name in error, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

TEMPLATE (DB2 Utilities)

LISTDEF (DB2 Utilities)

DSNU1020I *csect-name* **- EXPANDING LISTDEF** *list-name*

Explanation: PREVIEW processing is performing the catalog lookup necessary to expand the specified LISTDEF.

System action: PREVIEW processing continues.

Severity: 0 (informational)

DSNU1021I *csect-name* **- PROCESSING** *clause-type* **CLAUSE** *object-type qualifier.name*

Explanation: PREVIEW processing is expanding the specified clause of a LISTDEF. A DSNU1022I message follows this message to indicate how many objects are identified by each INCLUDE or EXCLUDE clause and message DSNU1023I to show the final expanded list..

System action: PREVIEW processing continues.

Severity: 0 (informational)

DSNU1022I *csect-name* **- CLAUSE IDENTIFIES** *n* **OBJECTS**

Explanation: PREVIEW processing has expanded one INCLUDE or EXCLUDE clause of a LISTDEF control statement. The clause expanded to the indicated *n* number of objects. The sum of the values shown on the DSNU1022I messages may not match the quantity shown on DSNU1023I due to INCLUDEs, EXCLUDEs and list overlap.

System action: PREVIEW processing continues.

Severity: 0 (informational)

DSNU1023I *csect-name* **- LISTDEF** *name* **CONTAINS** *n* **OBJECTS**

Explanation: PREVIEW processing has finished expanding a LISTDEF. The final list contains the indicated number of objects. The value shown may not match the sum of previous DSNU1022I quantities due to INCLUDEs, EXCLUDEs and list overlap.

System action: PREVIEW processing continues.

Severity: 0 (informational)

DSNU1024I *csect-name* **- OPTIONS EVENT WARNING HAS SET RETURN CODE TO** *return-code*

Explanation: Utility processing has ended with WARNING messages and a return code of 4. However, due to a previous OPTIONS EVENT WARNING control statement, the return code has been changed to the indicated return code.

System action: Utility processing halts if WARNING RC8 was specified, or continues if WARNING RC0 was specified.

User response: Use OPTIONS WARNING RC0 only on repetitive utility executions where WARNING messages are expected and acceptable.

Severity: 0 (informational) or 8 (warning)

DSNU1025I *csect-name* **-** *name* **IS A DD NAME**

Explanation: PREVIEW processing has checked the current utility control statement for TEMPLATE dynamic allocation and has found the specified name refers to a JCL DD card, not a template. In all cases the presence of a DD card takes precedent over any TEMPLATE of the same name.

System action: PREVIEW processing continues.

User response: If a TEMPLATE and DD card have the same name, the DD card always takes precedence. If the indicated name should not reference a DD card, perform one of the following actions:
- Remove the DD card
- Change the TEMPLATE name on the TEMPLATE control statement and the utility control statement that refers to it

If the reference to the DD card is correct, no action is necessary.

Severity: 0 (informational)

Related reference:

➡ TEMPLATE (DB2 Utilities)

DSNU1026I   *csect-name* **- LIST** *using-list* **REUSES LIST** *used-list*

Explanation: A LISTDEF list is INCLUDING or EXCLUDING another list that has already been used in expansion of this clause. This "circular reference" is not allowed. For example, the following specification is not allowed: LISTDEF A INCLUDE LIST B LISTDEF B INCLUDE LIST A

System action: Utility processing does not initiate.

User response: Correct one or more LISTDEF control statements to eliminate the circular reference and resubmit the utility job.

Severity: 8 (error)

Related tasks:

➡ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

➡ LISTDEF (DB2 Utilities)

DSNU1027I   *csect-name* **- PROCESSING CONTINUES DUE TO OPTIONS ITEMERROR SKIP**

Explanation: An error has occurred during utility processing, as indicated by previous error messages. However, because an OPTIONS ITEMERROR SKIP control statement is in force, processing continues despite the error.

csect-name
>   The name of the control section that issued the message.

System action: Message DSNU1027I is issued with a severity level that describes at what point in processing the condition occurred.

Severity 4
>   The list is being expanded, and processing of items in the list has not started. Utility processing continues on the current utility control statement and continues on to the next utility control statement in the job step.

Severity 8
>   Processing of items in the list has started. Utility processing continues on the current utility control statement. Subsequent utility control statements in the same job step are not processed.

Subsequent error conditions might prevent utility processing from completing.

User response: Take the action indicated by the severity level associated with the message.

Severity 4
>   In most cases, no action is required.

Severity 8
>   In most cases, correct the previous error, and resubmit the utility for the objects in error. Then resubmit the utility control statements that follow the control statement that ended with return code 8.
>
>   If the utility experiencing an error is QUIESCE LIST and a single quiesce point is required for the entire list, you must resubmit the entire list.

Severity: 4 (warning) or 8 (error)

Related concepts:

➡ DB2 online utilities (DB2 Utilities)

DSNU1028I   *csect-name* **- INVALID KEYWORD** *invalid-keyword* **FOR DASD TEMPLATE**

Explanation: A DASD TEMPLATE control statement cannot include the tape TEMPLATE keyword *invalid-keyword*.

System action: Utility processing does not initiate.

User response: DB2 uses the UNIT operand to determine the data set type. Either remove the tape keyword from the TEMPLATE control statement, or correct the UNIT operand to refer to a tape unit.

Severity: 8 (error)

Related concepts:

▶ Copies of lists of objects (DB2 Utilities)

Related tasks:

▶ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

▶ LISTDEF (DB2 Utilities)

▶ TEMPLATE (DB2 Utilities)

▶ Syntax and options of the TEMPLATE control statement (DB2 Utilities)

DSNU1029I *csect-name* **- INVALID KEYWORD** *invalid-keyword* **FOR TAPE TEMPLATE**

Explanation: A tape TEMPLATE control statement cannot include the DASD TEMPLATE keyword *invalid-keyword*.

System action: Utility processing does not initiate.

User response: DB2 uses the UNIT operand to determine the data set type. Either remove the DASD keyword from the TEMPLATE control statement, or correct the UNIT operand to refer to a DASD unit.

Severity: 8 (error)

Related concepts:

▶ Copies of lists of objects (DB2 Utilities)

Related tasks:

▶ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

▶ LISTDEF (DB2 Utilities)

▶ TEMPLATE (DB2 Utilities)

▶ Syntax and options of the TEMPLATE control statement (DB2 Utilities)

DSNU1030I *csect-name* **- QUALIFIER** *invalid-qualifier* **TOO LONG**

Explanation: The TEMPLATE DSN qualifier '*invalid-qualifier*' is greater than eight characters.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default execution will HALT.

User response: Correct the TEMPLATE DSN expression and resubmit the utility job.

Severity: 8 (error)

Related tasks:

▶ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

▶ OPTIONS (DB2 Utilities)

▶ TEMPLATE (DB2 Utilities)

DSNU1031I *csect-name* **- DSN** *name* **TOO LONG**

Explanation: The DSN operand in the TEMPLATE utility control statement results in a data set name that is too long for the type of data set that is being allocated.

csect-name
>   The name of the control section that issued the message.

name   The data set name.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the next object proceeds or halts based on the ITEMERROR specification in the OPTIONS control statement. By default, execution halts.

User response: Correct the TEMPLATE DSN operand in error, and resubmit the utility job.

Severity: 8 (error)

Related tasks:

▶ Referencing LISTDEF lists in other utility jobs (DB2 Utilities)

Related reference:

▶ OPTIONS (DB2 Utilities)

▶ TEMPLATE (DB2 Utilities)

DSNU1032I *csect-name* **- INVALID QUALIFIER** *qualifier* **IN TEMPLATE** *name*

Explanation: The TEMPLATE DSN qualifier either contains invalid characters or does not start with an alpha character.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, execution will HALT.

User response: Correct the TEMPLATE DSN operand in error, and resubmit the utility job. Common causes of this error message are:

- Omission of the trailing period from a TEMPLATE DSN variable name. All TEMPLATE DSN variable names are of the form &xyz. and must contain the trailing period.

- Invalid use of a partitioned data set.

Severity: 8 (error)

Related concepts:

▶ DB2 online utilities (DB2 Utilities)

▶ DB2 stand-alone utilities (DB2 Utilities)

Related reference:

▶ Syntax and options of the TEMPLATE control statement (DB2 Utilities)

DSNU1033I *csect-name* **- PROCESSING LIST ITEM:** *object-type qualifier***.**name*

Explanation: While processing a LISTDEF list, utility processing has begun on the indicated object.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU1034I *csect-name* **-** *primary-secondary* **SPACE** *units quantity* **TOO LARGE FOR DATA SET** *name*

Explanation: The TEMPLATE PRIMARY or SECONDARY SPACE quantity indicated is too large. It is not possible to dynamically allocate a data set larger than 16 777 215 units.

System action: Utility processing for the current object halts. If a list is being processed, utility processing of the

next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, execution will HALT.

User response: The DB2-calculated required space is too large to be dynamically allocated in the units shown. If the primary allocation is too large, use the PCTPRIME keyword to reduce the primary and increase the secondary. If the secondary allocation is too large, use the NBRSECND keyword to increase the number of secondary allocations. It is not possible to use TEMPLATE dynamic allocation for data sets larger than 16×(16 777 215) MB. If an object requires a data set of this size it must be processed individually using a DD card, not a template.

Severity: 8 (error)

DSNU1035I *csect-name* - *statement-type* **STATEMENT PROCESSED SUCCESSFULLY**

Explanation: DB2 issues this message following the successful execution of an OPTIONS, LISTDEF, or TEMPLATE utility control statement.

System action: DB2 processed and retained the options or definitions specified on the control statement for use on subsequent utility control statements.

Severity: 0 (informational)

DSNU1036I *csect-name* - **UNABLE TO ESTIMATE SPACE REQUIREMENTS FOR** *dataset-use*

Explanation: While processing a TEMPLATE control statement which does not include a SPACE keyword, DB2 could not calculate the default data set size. DB2 bases default calculations on the size of existing data sets. In this situation, DB2 could not determine the size of existing data sets because one or more of those existing data sets are:

- in-stream data sets,
- members of partitioned data sets,
- on tape,
- or results sets generated by DECLARE CURSOR executed with an EXEC SQL utility statement.

System action: DB2 halts utility processing for the current object. If the utility is processing a list of objects, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, processing will HALT.

User response: Add the SPACE keyword and its operands to the TEMPLATE control statement in error and rerun the utility. If the utility is the LOAD utility, you must add SPACE keywords for all TEMPLATE control statements referenced in the LOAD, except for the input data set.

Severity: 8 (error)

DSNU1037I *csect-name* - **PREVIEW PROCESSING CONTINUES DUE TO JCL PARAMETER**

Explanation: An OPTIONS utility control statement has attempted to turn off PREVIEW processing. PREVIEW processing continues because the PREVIEW JCL parameter has been specified. This parameter applies to the entire job step and can not be overridden by an OPTIONS control statement.

System action: Utility processing continues in PREVIEW mode.

User response: If PREVIEW processing must be turned off in the middle of a utility job step, remove the PREVIEW JCL parameter. Activate or deactivate PREVIEW processing by using the OPTIONS utility control statement, not the PREVIEW JCL parameter.

Severity: 0 (informational)

DSNU1038I *csect-name* - **DATASET ALLOCATED. TEMPLATE=***template-name* **DDNAME=***dd-name* **FILE SEQUENCE=***file-sequence* **DSN=***ds-name*

Explanation: The specified data set was successfully allocated based on a TEMPLATE control statement. *file-sequence* is shown for only tape data sets.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1039I • DSNU1044I

Related reference:

➡ TEMPLATE (DB2 Utilities)

DSNU1039I   *csect-name* **- PROCESSING LIST ITEM:** *object-type qualifier.name* **PARTITION** *partnum*

Explanation: While processing a LISTDEF list, utility processing has begun on the indicated object.

System action: Utility processing continues.

Severity: 0 (informational)

DSNU1040I   *csect-name* **- UNABLE TO LOCATE DSN** *dsname* **ON RESTART**

Explanation: DB2 could not find data set *dsname* when the utility was restarted. The data set is required for restart to continue.

System action: The utility step abends. The utility job step is in the stopped state.

User response: Refer to the DB2 abend code x'00E400C3' for more information.

DSNU1041I   *csect-name* **- UNABLE TO CREATE GDG BASE** *name reason*

Explanation: A TEMPLATE could not allocate a generation data group (GDG) data set, but the base GDG did not exist. The utility was unable to define the GDG base.

System action: Utility processing halts for the current object. If a list is in process, utility processing of the next object will proceed or halt based on the OPTIONS ITEMERROR control statement. By default, execution will HALT.

User response: Review the reason value to determine why the GDG base could not be defined. The possible reasons and suggested responses include:
- UNABLE TO LOAD IDCAMS PROGRAM. Ensure that the utility can invoke the IDCAMS program. Ensure that the utility can invoke the IDCAMS program.
- IDCAMS ABEND *xxx*. Use *xxx*, the abend code from the IDCAMS program, to determine why the IDCAMS program abended.
- ERROR IN IDCAMS. Review the IDCAMS message output that is listed between messages DSNU1042I and DSNU1043I to determine why the GDG base could not be defined.

Severity: 8 (error)

DSNU1042I   *csect-name* **- START OF IDCAMS MESSAGES**

Explanation: This message indicates the beginning of the IDCAMS messages or other related non-DB2 messages. DSNU1043I indicates the end.

User response: No action is required, but subsequent IDCAMS messages may require some action. See the *z/OS MVS System Messages Volumes 1-10* or other appropriate documentation for more information.

DSNU1043I   *csect-name* **- END OF IDCAMS MESSAGES**

Explanation: This message indicates the end of the IDCAMS or other related non-DB2 messages. DSNU1042I indicates the beginning.

User response: No action is required, but preceding messages may require some action. See the *z/OS MVS System Messages Volumes 1-10* for more information.

DSNU1044I   *csect-name* **- PROCESSING** *ddname* **AS EBCDIC**

Explanation: The utility control statement input data set begins with an EBCDIC character and is assumed for the duration of this job step to contain only EBCDIC characters.

System action: Processing of EBCDIC utility control statements continues.

User response: If processing of utility control statements as EBCDIC is incorrect make sure the first character in the

input data set is UNICODE, specifically UTF-8 CCSID 1208. This will cause the entire input data set to be processed as UNICODE.

Severity: 0 (informational)

DSNU1045I *csect-name* **- PROCESSING** *ddname* **AS UNICODE**

Explanation: The utility control statement input data set begins with a UNICODE character, specifically UTF-8 CCSID 1208, and is assumed for the duration of this job step to contain only UNICODE characters.

System action: Processing of UNICODE utility control statements continues.

Attention: UTF-8 quoted strings may not print correctly in utility error messages. All messages are printed in EBCDIC, and the quoted strings remain in UTF-8, causing the output to display poorly in EBCDIC.

User response: If processing of utility control statements as UNICODE is incorrect make sure the first character in the input data set is EBCDIC. This will cause the entire input data set to be processed as EBCDIC.

Severity: 0 (informational)

DSNU1046I *csect-name* **- SWITCH-TO-TEMPLATE** *template-name* **NOT FOUND**

Explanation: The LIMIT option of a TEMPLATE utility control statement refers to a switch-to-template name that was not found in SYSIN or SYSTEMP data set.

System action: Utility processing continues with original template.

User response: Provide the missing TEMPLATE definition in SYSIN prior to its use and resubmit the utility job. If the TEMPLATE definition exists in another data set, provide the SYSTEMPL DD card for the data set.

Severity: 4 (warning)

Related reference:

➡ TEMPLATE (DB2 Utilities)

DSNU1047I *csect-name* **- PROCESSING SKIPPED FOR** *object-type object-name* **REASON CODE=***code*

Explanation: The utility has skipped processing of the indicated object. The reason code shown indicates why processing was skipped.

1 - CLONE was specified and this object has no CLONE.

2 - This object was created with DEFINE NO and this utility does not process DEFINE NO objects.

System action: Processing continues for the remaining objects.

Severity: 0 (informational)

DSNU1048I *csect-name* **- PROCESSING SKIPPED FOR** *n* **OF** *m* **OBJECTS**

Explanation: This message summarizes DSNU1047I messages issued earlier during the execution of this utility control statement. It indicates how many objects were skipped for the list being processed.

System action: Processing continues for the remaining objects.

Severity: 0 (informational)

DSNU1049I *csect-name* **- THE** *utility-name* **UTILITY WAS NOT INVOKED. THERE WERE NO OBJECTS TO BE PROCESSED IN LISTDEF** *list-name***.**

Explanation: A utility detected an empty LISTDEF list. The utility that was processing the indicated list was not invoked.

csect-name
> The name of the control section that issued the message.

list-name
> The name of the list.

utility-name
    The name of the utility.

System action: The utility job ends with return code 4.

User response: If the empty LISTDEF list will always be empty, modify the LISTDEF parameters to eliminate the warning message. See the previous DSNU1008 message and other accompanying messages to determine why the LISTDEF is empty.

A common cause of this warning message is the LISTDEF DEFINED keyword default value of YES. If all objects in the list are undefined, an empty list is the result. Specify DEFINED ALL or DEFINED NO on the LISTDEF statement if undefined objects need to be processed.

Severity: 4 (warning)

Related reference:

➡ LISTDEF (DB2 Utilities)

➡ Syntax and options of the LISTDEF control statement (DB2 Utilities)

Related information:
"DSNU1008I" on page 825

DSNU1050I *csect-name* **- INVALID SUBSTRING NOTATION IN EXPRESSION** *string* **ON TEMPLATE** *template name*

Explanation: Invalid substring notation was detected on the TEMPLATE DSN operand. The portion of the DSN operand which is in error and the TEMPLATE name on which it occurred are shown.

System action: Utility processing is terminated.

User response: Correct the TEMPLATE control statement which is in error. Common causes of this error are incorrect syntax of the (s,l) clause and values of "s" or "l" which fall outside of the variable value at the time of execution.

Severity: 8 (error)

DSNU1051I *csect-name* **TEMPLATE SUBSYS IS NOT SUPPORTED TO ALLOCATE FILE FOR** *util-name* **UTILITY**

Explanation: Dynamically allocating a BatchPipes® or z/OS UNIX System Services file is not supported for *util-name* UTILITY.

User response: In the TEMPLATE utility control statement, remove the SUBSYS or PATH keyword and resubmit the job.

Severity: 8 (error)

DSNU1052I *csect-name util-name* **UTILITY DOES NOT SUPPORT RESTART WHEN TEMPLATE SUBSYS IS USED**

Explanation: The utility that uses the BatchPipes or z/OS UNIX System Services file cannot be restarted.

User response: Use the DB2 command TERM UTILITY to terminate the stopped utility, and then rerun the job.

Severity: 8 (error)

DSNU1053I *csect-name* **KEYWORD** *k1* **INVALID WITH KEYWORD** *k2* **ON** *utility-name*

Explanation: The use of KEYWORD *k1* is not allowed due to a conflicting keyword *k2* on another utility control statement. The conflicting LISTDEF or TEMPLATE name is shown.

User response: Correct one or both control statements in error and resubmit the job.

Severity: 8 (error)

| **DSNU1054I** **PARTITIONS of** *object-type object-name* **ARE NOT ALL IN THE SAME RBALRSN PAGE FORMAT.** **KEYWORD** *keyword-name* **REQUIRES KEYWORD PARTLEVEL.**

Explanation: The BASIC and EXTENDED keywords can be used on an entire partitioned object only when all partitions of that object are in the same format. Partitions of the indicated object exist in both BASIC and EXTENDED format.

object-type
    The type of object.

object-name
    The name of the object.

keyword-name
    The name of the keyword.

System action: Processing terminates.

User response: Either remove the indicated keyword, or add the PARTLEVEL keyword.

Severity: 8 (error)

Related reference:

➡ LISTDEF (DB2 Utilities)

| **DSNU1055I** **KEYWORD** *keyword-name* **IGNORED FOR ONE OR MORE 'DEFINE NO' OBJECTS**

Explanation: The indicated keyword cannot be evaluated for DEFINE NO objects. The keyword is ignored and the requested filtering is not performed for the DEFINE NO objects. The keywords remain part of the resulting LISTDEF INCLUDE or EXCLUDE result.

keyword-name
    The name of the keyword.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

➡ LISTDEF (DB2 Utilities)

DSNU1100I *csect-name* **RECORD 'n' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO** *field-name* **CONVERSION ERROR FOR** *table-name* **ERROR CODE** *cc*

Explanation: The load utility found the input data provided for a field to be incorrect. The corresponding input data set record 'n' from the input data set specified for partition *m* will be placed in the DISCARDS data set.

The error code, *cc*, identifies the problem. See DSNU334I for the list of error codes and their meaning.

System action: The input data set record is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set.

Severity: 4 (warning)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

Related information:

"DSNU334I" on page 674

DSNU1101I *csect-name* **RECORD '***n***' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO INVALID** *field-name*
COLUMN SPECIFICATION

Explanation: The load utility calculated the location of an input data value using the field specifications supplied and VARCHAR length fields present within the input data set record. The calculated position for the field was determined to be incorrect.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data, if one exists. Otherwise, correct the input field specification. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU1102I *csect-name* **RECORD '***n***' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO ERROR =**
X'*reason-code***' FROM EXIT ROUTINE** *procedure-name* **FOR** *table-name*

Explanation: The exit routine provided by your installation blocked the load of record '*n*' for partition *m* into the table indicated. The reason-code was set by the exit routine.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The record in error is placed in the DISCARDS data set. The LOAD utility continues to process additional records in the input data set.

System programmer response: Verify that the validation routine is correct.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data, if one exists. Ensure that the exit routine provided is correct. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU1103I *csect-name* **RECORD '***record-num***' FOR PARTITION** *part-num* **WILL BE DISCARDED DUE TO PART**
OR HIGH KEY VALUESPECIFICATION(S)

Explanation: The discarded record does not belong to the partition *part-num* being loaded. For a LARGE table space, this might be because the key is higher than the high key value specified for the last partition.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

System programmer response: Verify that the validation routine is correct.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data, if one exists. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set.

Severity: 4 (warning)

Related reference:

➡ LOAD (DB2 Utilities)

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU1104I *csect-name* **RECORD 'n' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO WHEN SPECIFICATION(S)**

Explanation: The discarded record does not satisfy the WHEN selection expression provided for partition *m*.

System action: The input data set record is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data, if one exists. Submit a RESUME(YES) LOAD invocation with the DISCARDS data set as the input data set. If FORMAT UNLOAD was specified on the load statement, the WHEN specifications are built internally. This ensures that the rows are loaded back into the same object from which they were unloaded.

Severity: 4 (warning)

Related reference:

▶ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU1105I *csect-name* **RECORD 'n' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO UNEXPECTED ERROR**

Explanation: The LOAD of input data set record number 'n' for partition *m* was attempted but was not successful due to an unexpected error.

System action: If the error was caused because a resource was not available, a DSNT500I message will be issued.

The utility step will be abended with a system X'04E' and an abend reason code of '00E40318'. If a DSNT500I message is not issued, an SVC dump will be requested. A SYSABEND dump of batch storage is also requested.

Operator response: An operator with SYSOPR authority can issue a TERM UTIL to free any held resources if that is necessary.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: If a DSNT500I message was issued, refer to the description under that message number. If the unavailable resource becomes available, the utility step is restartable.

If there is no DSNT500I message, the error is an unexpected internal error and you should consult your system programmer.

Severity: 8 (error)

Related reference:

▶ -TERM UTILITY (DB2) (DB2 Commands)

Related information:
"DSNT500I" on page 555

▶ 00E40318 (DB2 Codes)

DSNU1106I *csect-name* **RECORD 'n' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO INVALID** *field-name* **COLUMN SPECIFICATION FOR** *table-name*

Explanation: The load utility calculated the location of an input data value using the field specifications supplied and VARCHAR length fields present within the input data set record 'n' for partition *m*. The calculated position for the field was determined to be incorrect.

System action: The input data set record number is noted and will be discarded in the DISCARD phase. The LOAD utility continues to process additional records in the input data set.

User response: Following the completion of the current LOAD utility invocation, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data, if one exists. Otherwise, correct the input field specification. Submit a RESUME(YES) LOAD invocation, which uses the DISCARDS data set and the verified LOAD utility statement and field specifications.

Severity: 4 (warning)

Related reference:

➡ Syntax and options of the LOAD control statement (DB2 Utilities)

DSNU1107I   *csect-name* **- UNIQUE INDEX KEY DUPLICATES KEY OF INDEXED ROW AT RID** *record-id* **INDEX = *index-name* TABLE = *table-name* RECNO = *record-number* FOR PART = *partition* RID = *duplicate-id***

Explanation: A duplicate key or duplicate hash key was identified during a parallel LOAD operation. The indicated input data set record was loaded into a table partition, but it duplicates either a hash key or the key value of an existing record, which causes a unique index violation.

csect-name
        The name of the control section that issued the message.

record-id
        The record ID where the input data set was loaded.

index-name
        The name of the index.

table-name
        The name of the table.

record-number
        The number for the record in the input data set to load.

duplicate-id
        The record ID of the duplicate key value.

System action: Processing continues.

No index entry is created for the record at RID *duplicate-id*. The record will be deleted in a subsequent phase of processing.

User response: No action is required.

Severity: 4 (warning)

DSNU1108I   *csect-name* **UNIQUE INDEX KEY DUPLICATES KEY FROM INPUT DATASET RECORD** *iiii* **FOR PART** *m* **LOADED AT RID** *rrrr* **INDEX = *index-name* TABLE = *table-name* RECNO = *recno* RID = *duprid***

Explanation: Input data set record 'recno' was loaded into table 'table-name', partition *m*, at RID 'duprid', but it causes a unique index violation for index 'index-name'. It duplicates the key value of another record in the load job: the record loaded at RID 'rrrr' from input data set record 'iiii'.

System action: Processing continues. Index entries are not created for either record. Both records will be deleted in a subsequent phase of processing.

Severity: 4 (warning)

DSNU1109I   *csect-name* **RECORD '*n*' FOR PARTITION** *m* **WILL BE DISCARDED DUE TO CHECK CONSTRAINT** *constraint-name* **VIOLATION ON TABLE** *table-name*

Explanation: The discarded record violates the check constraint named *constraint-name* defined on table *table-name*. The record is not loaded and is discarded during the DISCARD phase.

System action: The LOAD utility continues to process additional records in the input data set.

User response: After the LOAD utility completes, examine the record in the DISCARDS data set for partition *m*. Correct the error in the discarded input data. Submit a RESUME(YES) LOAD job with the DISCARDS data set as the input data set.

Severity: 4 (warning)

DSNU1110I *csect-name* **RECORD 'n' FOR PARTITION** *m* **VIOLATES CHECK CONSTRAINT** *constraint-name* **ON TABLE** *table-name*

Explanation: The LOAD utility cannot load record *n* into table *table-name* partition *m* because the check constraint *constraint-name* would be violated. The record is not loaded.

System action: Processing terminates. An abend is requested so that the LOAD can be restarted after the error condition is corrected.

User response: Find the failing record in the input data set for partition *m*, correct the condition that caused the check constraint to be violated, and restart the LOAD utility.

Severity: 8 (error)

DSNU1111I *csect-name* **FOREIGN KEY HAS NO PRIMARY KEY FOR RELATIONSHIP=***r1* **INPUT RECORD NUMBER=***nnnn*, **PART** *m*, **RID=X'rrrr'**

Explanation: Input record 'nnnn' for partition *m* caused a row to be loaded into table 'table' at RID 'rrrr' with no primary key for the relationship with the name 'r1'.

System action: Processing continues.

User response: None required. The constraint violation described will be rectified later in processing.

Severity: 4 (warning)

DSNU1112I *csect-name* **DISCARD PHASE STATISTICS -** *nnn* **INPUT DATA SET RECORDS DISCARDED FOR PARTITION** *m*

Explanation: Input data set records were rejected by the LOAD utility for one or several reasons. The rejected records have been placed in a discards data set. The number of records rejected for partition *m* is 'nnn'. Message DSNU399I summarizes the errors causing the records to be rejected.

System action: Processing continues.

User response: Examine the LOAD error summary and decide which of the discarded records, if any, should be corrected and reapplied to the table space.

Severity: 4 (warning)

DSNU1113I *csect-name* **- LOAD UTILITY ERROR SUMMARY REPORT**

Explanation: This message summarizes errors detected during the load job. A report is issued only if unique index or referential constraint violations were detected. The report contains one line for each error. The errors are listed first by error severity and within that by *idsrn*.

The report fields have the following meanings:

sev

> The severity of the error reported on this line:

Severity	Type of error	Explanation
1	Primary	Primary referential integrity errors and all nonreferential integrity errors.
2	Secondary	Secondary referential integrity errors (that is, errors that result from a primary referential integrity error in an ancestor).

idsrn The relative record number in the input data set, for the partition identified by *pno*, of the record causing the error reported on this line.

pno The partition number for the input data set of the record causing the error reported on this line.

ddsrn The relative record number in the input data set of the record causing the error reported on this line, if discard processing is in effect. Otherwise, *ddsrn* is blank.

type A description of the type of error reported on this line, as follows:

- Conversion
- Field specification
- Validation
- Part Specification
- Unexpected
- Column specification
- Referential integrity
- Check constraint
- Duplicate key.

name1 The name of the table for which the record causing the error was targeted, if this information is pertinent for the type of error reported on this line. Otherwise, *name1* is blank. This field has nonblank values for the following error types:

- Conversion
- Validation
- Column specification
- Referential integrity (primary).
- Check constraint

name2 The name of the field or fanset involved in the error reported on this line, if this information is pertinent for the type of error reported. Otherwise, *name2* is blank. This field contains a field name for the following types of errors:

- Conversion
- Field specification
- Column specification.

For primary referential integrity errors, *name2* is a relationship name. For check constraint errors, *name2* is a check constraint name.

idsrn2 The input data set record number of a related error. For duplicate key errors, it gives the input data set record number of the record whose key is duplicated. For secondary referential integrity errors, it gives the input data set record number of the record whose primary referential integrity error cascaded to cause this secondary error. For other types of errors, this field is unused.

System action: Processing continues.

Severity: 4 (warning)

DSNU1114I *csect-name* **(RE)LOAD PHASE STATISTICS NUMBER OF RECORDS LOADED =** *nnnn* **FOR TABLE** *table-name*

Explanation: When the SHRLEVEL CHANGE option is selected, DB2 issues this message upon normal completion of the (RE)LOAD phase of the LOAD utility. It indicates the number of records loaded into TABLE *table-name* from the source record data set (that is, the data set identified by the JCL DD statement identified in the INDDN option for the LOAD. This message is repeated for each TABLE loaded by the current LOAD utility statement

System action: Processing of the RELOAD phase continues.

User response: Verify that the number of record loaded reported is consistent with expectations. For example, if four records are processed, the user should expect to see NUMBER OF RECORD LOADED = 4.

Severity: 0 (informational)

DSNU1115I *csect-name nnnn* **RESTART PHASE NOT ALLOWED IN THE (RE)LOAD PHASE, PHASE CHANGED TO CURRENT**

Explanation: This message is issued when the RESTART PHASE is requested for a LOAD SHRLEVEL CHANGE utility. RESTART PHASE is not allowed for LOAD SHRLEVEL CHANGE.

System action: Processing of the RELOAD phase continues.

User response: No action is necessary. If another RESTART is required, use the RESTART option instead of RESTART PHASE.

Severity: 4 (warning)

DSNU1116I   *csect-name* **FOREIGN KEY HAS NO PRIMARY KEY FOR RELATIONSHIP=***r1* **INPUT RECORD NUMBER=***nnnn*

Explanation: Input record *nnnn* has no primary key for the relationship with the name *r1*. The row is not loaded.

System action: Processing continues.

User response: None required.

Severity: 4 (warning)

DSNU1117I   *csect-name* **- UNIQUE INDEX KEY DUPLICATES KEY OF INDEXED RECORD AT RID** *existing-RID* **INDEX =** *index-name* **TABLE =** *table-name* **RECNO =** *duplicate-RID*

Explanation: An input data set record that was loaded into a table duplicates the key value of a record that is already in the table, which causes a unique index violation.

csect-name
> The name of the control section that issued the message.

existing-RID
> The record ID that was already in the table.

index-name
> The name of the index.

table-name
> The name of the table.

duplicate-RID
> The new record ID, which would cause the unique index violation.

System action: Processing stops if the LOAD utility with the SHRLEVEL CHANGE option was used on a table space that is defined with the NOT LOGGED attribute. If the table space is partitioned, all partitions that had at least one row loaded by the current LOAD job are placed into the recovery pending (RECP) and logical page list (LPL) states.

Otherwise, processing continues. No index entry is created for the record at RID *duplicate-RID*.

User response: No action is required.

Severity: 4 (warning)

8 (error)

DSNU1118I   *csect-name* **(RE)LOAD PHASE STATISTICS -** *nnnn* **DUPLICATE KEY ERRORS ENCOUNTERED**

Explanation: This message is issued by the (RE)LOAD phase of the LOAD utility when the SHRLEVEL CHANGE option is used. The number of duplicate key errors that were encountered during processing is *nnnn*.

System action: Processing continues.

Severity: 0 (informational)

DSNU1119I   *csect-name* **(RE)LOAD PHASE STATISTICS** *violation-count* **CONSTRAINT VIOLATIONS ENCOUNTERED**

Explanation: The LOAD utility found the indicated number of referential constraint violations.

This informational message is issued by the LOAD utility during the RELOAD phase when the SHRLEVEL CHANGE option is used.

csect-name
> The name of the control section that issued the message.

violation-count
> The number of referential constraint violation errors that were encountered during processing.

System action: Processing continues.

User response: Correct any referential constraint violations.

Severity: 0

Related concepts:

▶ Referential constraints (Introduction to DB2 for z/OS)

▶ Referential constraint violations (DB2 Utilities)

Related reference:

▶ LOAD (DB2 Utilities)

DSNU1120I *lob-or-xml* **TABLE NOT DEFINED**

Explanation: The LOB or XML table space *dbname.tsname* has been created, but no auxiliary table has been created in it.

System action: Utility operation is terminated with return code 8.

User response: Complete the definition of the LOB table space before resubmitting the utility.

Severity: 8 (error)

DSNU1121I *csect-name* **(RE)LOAD PHASE STATISTICS - NUMBER OF RECORDS LOADED =** *nnnn* **FOR TABLE** *table-name* **PART** *m*

Explanation: When the SHRLEVEL CHANGE option is selected, DB2 issues this message upon normal completion of the (RE)LOAD phase of the LOAD utility. It indicates the number of records that were loaded into part *m* of table *table-name* from the source record data set. The message repeats for each INTO TABLE PART clause in the LOAD utility statement.

System action: Processing continues.

User response: Verify that the number of records loaded reported is consistent with expectations.

Severity: 0 (informational)

DSNU1122I *csect-name* **JOB** *job-name* **PERFORMING** *util-type* **WITH UTILID** *util-id* **UNABLE TO DRAIN** *object-name*. **RETRY** *n* **OF** *m* **WILL BE ATTEMPTED IN** *s* **SECONDS**

Explanation: A utility with the RETRY option specified has been unable to drain a table space, table space part, or index. The utility will make another attempt to acquire the drain after a delay.

System action: The utility job continues.

User response: The utility job attempts to acquire a restrictive access to the referenced object, and will retry if unsuccessful. Optionally, the user can determine who has claims on the object using the -DIS DATABASE command with CLAIMERS option, and cancel or end those threads.

Severity: 0 (informational)

DSNU1124I *csect-name* **- UNIQUE INDEX KEY DUPLICATES KEY OF INDEXED RECORD AT RID** *existing-RID* **INDEX =** *index-name* **TABLE =** *table-name* **RECNO =** *duplicate-RID* **FOR PART =** *part-number*

Explanation: The input data set record indicated for PART *part-number* of table *table-name* causes a unique index violation for index *index-name*. It duplicates the key value of a record already in the table at RID *existing-RID*. The record was not loaded.

csect-name
> The name of the control section that issued the message.

existing-RID
> The record ID that was already in the table.

index-name
> The name of the index.

table-name
> The name of the table.

duplicate-RID
> The new record ID, which would cause the unique index violation.

part-number
> The value that identifies the partition or data set.

System action: Processing stops if the LOAD utility with the SHRLEVEL CHANGE option was used on a table space that is defined with the NOT LOGGED attribute. All partitions that had at least one row loaded by the current LOAD job are placed into the recovery pending (RECP) and logical page list (LPL) states.

Otherwise, processing continues. No index entry is created for the record at RID *duplicate-RID*.

User response: No action is required.

Severity: 4 (warning)

8 (error)

DSNU1125I *csect-name* **FOREIGN KEY HAS NO PRIMARY KEY FOR RELATIONSHIP=***r1* **INPUT RECORD NUMBER=***nnnn***,FOR PART** *m*

Explanation: Input record *nnnn* for partition *m* for table *table* has no primary key for the relationship with the name *r1*. The row is not loaded.

System action: Processing continues.

User response: None required.

Severity: 4 (warning)

DSNU1126I *csect-name* **PARTITION** *n* **IS SPECIFIED IN MORE THAN ONE INTO TABLE PART CLAUSE, PARTITION PARALLELISM IS DISALLOWED**

Explanation: LOAD partition parallelism was requested via INDDN or INCURSOR following INTO TABLE PART clauses. However, the same partition was specified on more than one INTO TABLE PART clause. Partition parallelism is not supported for this case.

System action: Utility operation is terminated with return code 8.

User response: Correct the LOAD statement in one of the following ways:
- Specify a different partition number on each INTO TABLE PART clause
- Remove the parallelism by removing the INDDN or INCURSOR keywords from the INTO TABLE PART clauses.

Severity: 8 (error)

DSNU1127I *csect-name op* **OPERATION FOUND ON RECORD REMOVED BY DISCARD CRITERIA**

Explanation: Operation *op* was detected during a REORG SHRLEVEL CHANGE on a record removed during the UNLOAD phase.

System action: Utility processing is terminated.

User response: Rerun the utility when data being discarded by REORG is not being updated or deleted by applications or SQL.

Severity: 8 (error)

DSNU1128I *csect-name* **FEWER PAGES THAN PARTS FOR REBALANCE**

Explanation: In attempting to rebalance partitions during REORG, it is estimated that there are fewer total pages than the number of partitions being rebalanced.

System action: Remove the REBALANCE keyword form the REORG until there is sufficient data to populate pages for every partition.

Severity: 8 (error)

| **DSNU1129I** *csect-name* **- PARTITION RANGE NOT CONTIGUOUS FOR REBALANCE**

Explanation: The partition range that was specified on a REORG REBALANCE does not represent a set of contiguous logical partitions. Due to a series of ALTER TABLE ADD PART and ALTER TABLE ROTATE PART statements, the physical partition numbers do not match the logical partition numbers.

The physical partition numbers that were specified as the REORG partition range do not represent a contiguous set of logical partitions. For example, physical partitions 1 and 2 might represent logical partitions 2 and 4.

System action: The REORG terminates with return code 8.

User response: Use the DISPLAY DATABASE command to identify a range of physical partition numbers that represent a contiguous set of logical partitions, and resubmit the REORG REBALANCE while specifying that physical partition range. Alternatively, omit the partition range on the REORG REBALANCE so that the entire table space is reorganized.

Severity: 8 (error)

DSNU1130I *csect-name* **- NOT ALL PARTITIONS POPULATED BY REBALANCE - PROCESSING TERMINATES**

Explanation: The table space that is being reorganized has many rows with duplicate partitioning key values, and there were insufficient unique values to allow distributing rows across all partitions during a rebalance.

System action: The utility is terminated with return code 8. For REORG SHRLEVEL NONE, the table space is left in the recover pending restricted state.

User response: If the REORG was REORG SHRLEVEL NONE, recover the table space to the point in time before the REORG.

Severity: 8 (error)

DSNU1132I *csect-name* **COPYDICTIONARY OPERAND CANNOT BE A PARTITION THAT IS BEING REPLACED**

Explanation: LOAD COPYDICTIONARY was specified on a partition that was also being replaced. COPYDICTIONARY is not supported for this case.

System action: Utility operation is terminated with return code 8.

User response: Correct the LOAD statement in one of the following ways:
- Specify a different partition number on the INTO TABLE PART clause.
- Specify a different partition number on the COPYDICTIONARY option.

Severity: 8(error)

DSNU1133I *csect-name* **- INDEX BUILD DEFERRED FOR INDEX** *index-name*. **INDEX WILL BE LEFT IN RBDP STATE.**

Explanation: The LOAD utility specified the INDEXDEFER option, and the index build for the specified index or index part was deferred.

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index. The index is left in the REBUILD PENDING (RBDP) state.

System action: Utility processing continues. The index build for the specified index is deferred and the index or index part is left in RBDP.

User response: Use the REBUILD INDEX utility after the LOAD job completes.

Severity: 0

Related reference:

▶ LOAD (DB2 Utilities)

▶ REBUILD INDEX (DB2 Utilities)

DSNU1134I   *csect-name* **- INDEX BUILD NOT DEFERRED FOR INDEX** *index-name*. **REASON CODE=***reason-code*

Explanation: The LOAD utility specified the INDEXDEFER option, but the index build for the specified index was not deferred.

csect-name
> The name of the control section that issued the message.

index-name
> The name of the index.

reason-code
> A numeric value that indicates the reason for the failure:
>
> **1**   The specified index is an indexed foreign key and ENFORCE CONSTRAINTS was specified.

System action: Utility processing continues. The index build for the specified index is not deferred.

User response: No action is required.

Severity: 4 (warning)

Related reference:

▶ LOAD (DB2 Utilities)

DSNU1135I   **EXTENDED PAGE FORMAT CONVERSION ON XML TABLESPACE** *object-type object-name* **IS IGNORED**

Explanation: The conversion to extended page format for the target XML table space is ignored. The message occurs in the following situations:

- A partition-level REORG on the XML table space
- A REORG TABLESPACE SHRLEVEL NONE on the XML table space
- LOAD REPLACE on the base table space
- REBUILD on the node id index if the XML table space is not in extended page format

object-type
> The type of object: TABLESPACE or NODE INDEX.

object-name
> The name of the object:
> - Database and table space name for the target of the conversion
> - Creator name and index name for the target of the index conversion

System action: The request is ignored.

User response: Run a REORG TABLESPACE SHRLEVEL REFERENCE or CHANGE on the entire XML table space to convert it to extended page format.

Severity: 4 (warning)

Related reference:

▶ REORG TABLESPACE (DB2 Utilities)

| **DSNU1136I** *csect-name* **SYSLGRNX IS NOT USED FOR LOG READ DUE TO** *reason-text*

| **Explanation:** The REORG utility is not using the SYSLGRNX records to determine the log read range.

| *csect-name*
| The name of the control section that issued the message.

| *reason-text*
| The reason that the SYSLGRNX object is not used:
| • LOGRANGES NO SPECIFICATION
| • COPY NO INDEX
| • DIRECTORY OBJECT
| • SYSCOPY CATALOG OBJECT

| **System action:** Utility processing continues.

| **Operator response:** No action is required.

| **Severity:** 0

| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

| ➡ REORG INDEX (DB2 Utilities)

| **DSNU1137I** *csect-name* **ALTER OF MAXRO DEFER TO** *new-value* **DETECTED**

| **Explanation:** A REORG utility statement that is specified with the MAXRO DEFER keywords has detected that the ALTER UTIL command changed the MAXRO value.

| *csect-name*
| The name of the control section that issued the message.

| *new-value*
| The new MAXRO value that was changed by the user using the ALTER UTIL command.

| **System action:** Utility processing continues.

| **Operator response:** No action is required.

| **Severity:** 0

| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

| ➡ REORG INDEX (DB2 Utilities)

| **DSNU1138I** *csect-name* **DRAIN** *claimer-class* **WITH START TIME** *start-time* **HAS COMPLETED SUCCESSFULLY**

| **Explanation:** REORG has successfully drained the specified class in its processing.

| *csect-name*
| The name of the control section that issued the message.

| *claimer-class*
| Specifies the class of claimers that REORG has drained successfully.

| *start-time*
| Specifies the starting timestamp of this draining process.

| **System action:** Utility processing continues.

| **Operator response:** No action is required.

| **Severity:** 0

| Related reference:

| [→] REORG TABLESPACE (DB2 Utilities)

| [→] REORG INDEX (DB2 Utilities)

| **DSNU1139I** *csect-name* **FINAL LOG ITERATION STATISTICS. NUMBER OF LOG RECORDS =** *quantity-read*

| **Explanation:** The LOG phase of the utility has completed.

| *csect-name*
| The name of the control section that issued the message.

| *quantity-read*
| The number of log records that were read. Application of some of these log records might be unnecessary.
| Therefore, the LOG phase might apply only a subset of these log records to the shadow data sets.

| **System action:** Utility processing continues.

| **Operator response:** No action is required.

| **Severity:** 0

DSNU1140I *csect-name* **MESSAGE LIMIT OF** *max-messages* **HAS BEEN REACHED FOR** *message-number*
MESSAGES. NO MORE WILL BE ISSUED FOR THIS UTILITY

Explanation: The number of instances of the indicated message that is issued during a utility operation has
exceeded the maximum number that is permitted.

csect-name
The name of the control section that issued the message.

max-messages
The maximum number of messages that are issued before this message is issued. This value is controlled by
the utility that issues message DSNU1140I.

message-number
The message number that has reached the limit and is now suppressed.

Examples:

LOAD example
The LOAD utility issues message DSNU1140I after 2000 instances of any of the following messages:
DSNU313I, DSNU314I, DSNU1103I, DSNU1104I DSNU344I,DSNU345I, DSNU1107I, or DSNU1108I. Any
rows that would have triggered the suppressed message are still discarded.

UNLOAD example
The UNLOAD utility issues message DSNU1140I after 2000 instances of message DSNU1239I when both of
the following conditions are true:

- The UNLOAD statement includes either the FROMCOPY or FROMCOPYDDN option.
- The value of the MAXERR option in the UNLOAD statement is either a number greater than 2000 or the
number 0.

Any rows that would have triggered the suppressed message are still not unloaded.

System action: Processing continues.

No additional instances of *message-number* messages are issued. Only the issuance of the message is suppressed. Any
action that is associated with the suppressed message number still occurs.

User response: No action is required.

Severity: 0

Related reference:

LOAD (DB2 Utilities)

Syntax and options of the UNLOAD control statement (DB2 Utilities)

Related information:

"DSNU313I" on page 668

"DSNU314I" on page 669

"DSNU1103I" on page 838

"DSNU1104I" on page 839

"DSNU344I" on page 679

"DSNU345I" on page 680

"DSNU1107I" on page 840

"DSNU1108I" on page 840

"DSNU1239I" on page 872

DSNU1141I *csect-name* **INVALID DELIMITER CHARACTER** *invalid-delimiter*

Explanation: The delimiter character *invalid-delimiter*, which was specified as the operand of a COLDEL, CHARDEL, or DECPT keyword, is not a valid delimiter.

System action: Utility processing is not initiated.

User response: Correct the invalid delimiter characters and resubmit the job.

Severity: 8 (error)

Related reference:

Delimited file format (DB2 Utilities)

DSNU1142I *csect-name* **RECORD** *n* **DELIMITER MISMATCH FOR COLUMN** *colname*

Explanation: A column or character delimiter character that marks the end of input for column *colname* was not found before the end of the record.

System action: Processing terminates. An abend is requested so that the LOAD can be restarted after the error is corrected.

User response: Find the failing record in the input data set, correct the condition that caused the error, and restart the LOAD utility.

Severity: 8 (error)

DSNU1143I *csect-name* **RECORD** *n* **WILL BE DISCARDED DUE TO DELIMITER MISMATCH FOR COLUMN** *colname*

Explanation: A column or character delimiter character that marks the end of input for column *colname* was not found before the end of the record. The record is not loaded, and is discarded during the DISCARD phase.

System action: The LOAD utility continues to process records in the input data set.

User response: After the LOAD utility completes, examine the records in the discard data set. Correct the errors in the discarded input data, and submit a LOAD RESUME YES job with the DISCARD data set as the input.

Severity: 4 (warning)

DSNU1144I *csect-name* **FORMAT DELIMITED REQUIRES ONE INTO TABLE**

Explanation: LOAD FORMAT DELIMITED supports loading only a single table. LOAD FORMAT DELIMITED was specified in a LOAD statement with more than one INTO TABLE clause.

System action: Processing terminates.

User response: Correct the LOAD statement and resubmit the utility.

Severity: 8 (error)

| **DSNU1145I**   **THE OBJECT CONVERSION TO BASIC PAGE FORMAT IS PREVENTED BY THE NOBASIC**
| **SETTING OF SUBSYSTEM PARAMETER UTILITY_OBJECT_CONVERSION**

| **Explanation:** Either an attempt was made to convert an object to basic 6-byte page format when it is in extended
| 10-byte page format or to keep the object in basic 6-byte page format instead of converting it to extended 10-byte
| format. This action is not allowed when the UTILITY_OBJECT_CONVERSION subsystem parameter is set to
| NOBASIC.

| **System action:** Utility processing is terminated.

| **User response:** Resubmit the job without the RBALRSN_CONVERSION keyword or with the
| RBALRSN_CONVERSION EXTENDED keywords. If the object of the utility needs to be converted to basic 6-byte
| page format or stay in basic 6-byte page format, contact the database administrator to change the
| UTILITY_OBJECT_CONVERSION subsystem parameter.

| **Severity:** 8 (error)

| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

DSNU1147I   *csect* **- (RE)LOAD PHASE STATISTICS - TOTAL NUMBER OF RECORDS LOADED=***nnnn* **FOR**
TABLESPACE *dbname.tsname*

Explanation: This message is issued at the normal completion of the RELOAD phase of the LOAD utility. It
indicates the total number of records loaded into the table space.

System action: The utility continues.

User response: The user can verify that the number of records matches expectations. The total includes all records
loaded into all tables for a segmented or linear table space, and all records loaded into partitions for a partitioned
table space.

Severity: 0

DSNU1148I   *csect* **- (RE)LOAD PHASE STATISTICS - NUMBER OF INPUT RECORDS PROCESSED=***nnnn* **FOR**
PART *n*

Explanation: The message is issued from the LOAD utility when LOAD partition parallelism is being used and the
loading of partition *n* has been completed. The message indicates the total number of records that were processed
from the input file for the partition. The message is issued for each partition being loaded.

System action: The utility continues.

User response: The user can verify that the number of records matches expectations.

Severity: 0

DSNU1149I   *csect* **- (RE)LOAD PHASE STATISTICS - NUMBER OF INPUT RECORDS NOT LOADED=***nnnn*
FOR PART *n*

Explanation: The message is issued from the LOAD utility when LOAD partition parallelism is being used and the
loading of partition *n* has been completed. It indicates the total number of records from the input file for partition *n*
that were not loaded. The records might not have been loaded because they were excluded by a WHEN clause,
because their partitioning key did not match partition *n*, or because there was an error processing the record.

The message is not issued if *nnnn* is zero, or if the job was restarted from the last commit point.

System action: The utility continues.

User response: The user can verify that the number of records matches expectations. For example, if this message
appears it might be an indication that some rows in the input file belong in other partitions, and should have been
loaded into those partitions.

Severity: 0

DSNU1150I *csect* **- (RE)LOAD PHASE STATISTICS - NUMBER OF INPUT RECORDS NOT LOADED=***nnnn*

Explanation: The message is issued at the normal completion of the RELOAD phase of the LOAD utility. It indicates the total number of records from the input file or files that were not loaded. The records might not have been loaded for one of the following reasons:

- They were excluded by a WHEN clause
- They were being loaded into a specific partition with INTO TABLE PART but their partitioning key did not match that partition
- There was an error processing the record

The message is not issued if *nnnn* is zero, or if the job was restarted from the last commit point.

System action: The utility continues.

User response: The user can verify that the number of records matches expectations. For example, when loading into a single partition, if this message appears it might be an indication that some rows in the input file belong in other partitions, and should have been loaded into those partitions.

Severity: 0

DSNU1151I **csect-name REORGLOB PHASE COMPLETE - NUMBER OF RECORDS PROCESSED = nnnnnnnn**

Explanation: This message is issued at the normal completion of the REORGLOB phase of the REORG utility. It indicates the number of records processed by the utility.

User response: Verify the number of records loaded into the LOB table space.

Severity: 0 (informational)

DSNU1152I *csect-name utility-name* **SHRLEVEL** *option* **OF** *object-type object-qualifier.object-name*

Explanation: The SHRLEVEL option was specified incorrectly for a utility.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility: REORG TABLESPACE, REORG INDEX, or REBUILD INDEX.

option The SHRLEVEL option: CHANGE, REFERENCE, or NONE. The combination of *option* and *utility-name* indicates what has occurred.

> **CHANGE**
>> If *utility* is REORG TABLESPACE: SHRLEVEL CHANGE was specified when REORG TABLESPACE was run on a NOT LOGGED table space, or on a LOB table space for which the base table space is a NOT LOGGED table space.
>>
>> If *utility* is REBUILD INDEX or REORG INDEX: SHRLEVEL CHANGE was specified when REBUILD INDEX or REORG INDEX was run on an index that is defined on a NOT LOGGED table space.

> **REFERENCE**
>> SHRLEVEL REFERENCE was specified when REORG TABLESPACE with PART was run on a NOT LOGGED table space on which nonpartitioned indexes are defined.

> **NONE** SHRLEVEL NONE cannot be specified in the following situations:
>> - For a table space with hash organization.
>> - When REORG TABLESPACE is run against either DSNDB06.SYSTSVOL or DSNDB06.SYSTSSTG and the table space has a user-defined index that uses a storage group that is managed by DB2.

object-type
> The type of object: INDEX, INDEXSPACE, or TABLESPACE.

object-qualifier.object-name
> The name of the object, as a qualified name.

System action: If the utility was operating on a list of objects and OPTIONS EVENT(ITEMERROR,SKIP) was specified, then processing continues to the next object in the list. Otherwise, processing terminates.

Operator response: Take the appropriate action for the *object-type* value:

NOT LOGGED TABLESPACE
>   Remove the SHRLEVEL(CHANGE) or SHRLEVEL(REFERENCE) option from the utility statement.

HASH TABLESPACE
>   Specify either SHRLEVEL(CHANGE) or SHRLEVEL(REFERENCE) option in the utility statement

Severity: 8 (error)

DSNU1153I   *csect-name* **LOG YES SPECIFIED FOR THE NOT LOGGED** *obj-type objqual.obj-name* **WILL BE IGNORED**

Explanation: LOG YES was specified for a LOAD, REORG, CHECK DATA, or REPAIR utility job that refers to an object with a NOT LOGGED logging attribute. Logging is not supported for a not logged table space.

System action: Processing continues, the LOG YES option is ignored.

User response: Remove the LOG YES option from the Utility input.

Severity: 0 (informational)

DSNU1154I   *csect-name* **RESTART OF THE UTILITY IS NOT ALLOWED AGAINST THE NOT LOGGED TABLE SPACE** *objqual.obj-name*

Explanation: You cannot restart the online load (LOAD RESUME YES SHRLEVEL CHANGE) against the table space with the NOT LOGGED attribute. The failed LOAD job could not roll back to the previous commit point because the log records that are required to perform that function were not written.

System action: Processing terminates.

User response: Terminate the failed LOAD job. Before rerunning the job, the table space must be recovered. DB2 recovers to the last image copy or to the point when CREATE or ALTER NOT LOGGED was done, whichever is most recent.

Severity: 8 (error)

DSNU1155I   *csect-name* **- LOB TABLE SPACES WILL BE REORGANIZED IN PARALLEL WITH THE ASSOCIATED BASE TABLE SPACE OBJECTS**

Explanation: The REORG TABLESPACE utility will reorganize auxiliary LOB table spaces along with the associated base table space objects that were specified for the REORG.

csect-name
>   The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

 REORG TABLESPACE (DB2 Utilities)

DSNU1156I   *csect-name* **- REORG TABLESPACE SHRLEVEL NONE OR AUX NO CANNOT BE RUN ON PARTITIONED TABLE SPACE IN REORG PENDING WITH LOB COLUMN DEFINED**

Explanation: REORG TABLESPACE has been executed with either SHRLEVEL NONE or AUX NO against a partitioned table space that has LOB columns defined. If the base table space is in a REORG PENDING state, then all auxiliary LOB table spaces must be reorganized along with the base table space. When REORG TABLESPACE is executed with either SHRLEVEL NONE or AUX NO, and the base table space is in a REORG PENDING state, the reorganization cannot happen.

csect-name
> The name of the control section that issued the message.

System action: Processing of the REORG utility job ends.

User response: Execute REORG TABLESPACE against the partitioned table space and specify either SHRLEVEL REFERENCE or CHANGE AUX YES.

Severity: 8 (error)

Related reference:

▶ REORG TABLESPACE (DB2 Utilities)

DSNU1157I *csect-name* **- INLINE IMAGE COPIES ARE NOT CREATED FOR AUXILIARY TABLE SPACES REORGANIZED AND ARE LEFT IN COPY PENDING**

Explanation: A REORG TABLESPACE job that uses the AUX YES option (either explicitly or as the default option) is unable to create inline image copy data sets for one or more LOB table spaces that were reorganized.

csect-name
> The name of the control section that issued the message.

System action: Processing of the REORG TABLESPACE utility job continues. This condition does not prevent the REORG from completing successfully, but the LOB table spaces are left with a COPY pending status.

User response: Create an image copy on the affected LOB table spaces after REORG completion.

Severity: 8 (error)

Related reference:

▶ REORG TABLESPACE (DB2 Utilities)

DSNU1158I *csect-name* **- REORG TABLE SPACE SHRLEVEL NONE OR AUX NO CAN NOT BE RUN ON TABLE SPACE WITH ALTERED INLINE LOB COLUMNS**

Explanation: The auxiliary LOB table space must be reorganized along with the base table space because one or more LOB columns have an altered INLINE LENGTH that must be materialized.

csect-name
> The name of the control section that issued the message.

System action: The REORG utility is terminated.

Programmer response: Execute the REORG TABLESPACE utility and specify either the SHRLEVEL REFERENCE or SHRLEVEL CHANGE keyword and the AUX YES keyword.

Related reference:

▶ REORG TABLESPACE (DB2 Utilities)

DSNU1159I *csect-name* **- THE POINT-IN-TIME RECOVERY PROCESS WAS NOT COMPLETED ON TABLESPACE** *tablespace-name*

Explanation: The specified base table space and the associated LOB table space were reorganized to materialize pending definition changes. After that REORG job was run, a RECOVER job was run to recover the data to a point in time before the REORG job ran. A subsequent REORG job must be run on the base table space and the LOB table space to complete the point-in-time recovery process. However, the REORG statement was executed on the base table space first, so the LOB table space could not be reorganized.

tablespace-name
> The name of the table space.

System action: Utility processing is not initiated.

User response: Run REORG on the LOB table space first, and then run REORG on the base table space to complete the point-in-time recovery process.

Severity: 8 (error)

Related reference:

REORG TABLESPACE (DB2 Utilities)

DSNU1160I   *csect-name* **- PARTITIONS WILL BE UNLOADED/RELOADED IN PARALLEL, NUMBER OF TASKS=***n*

Explanation: The REORG TABLESPACE utility is able to unload and reload partitions in parallel. The number of tasks started is *n*.

System action: REORG processing continues with the UNLOAD phase.

Severity: 0 (informational)

DSNU1161I   *csect-name* **- NON-PARTITIONED INDEXES WILL BE UNLOADED IN PARALLEL, NUMBER OF TASKS=***n*

Explanation: The REORG TABLESPACE SHRLEVEL CHANGE or REFERENCE utility is operating on a subset of the partitions of a partitioned table space. The table space has one or more NPIs, and the utility has determined that the NPIs can be unloaded in parallel with the unload of the table space partitions. The number of tasks started is *n*.

System action: REORG processing continues with the UNLOAD phase.

Severity: 0 (informational)

DSNU1162I   *csect-name* **- LOG APPLIES WILL BE PERFORMED IN PARALLEL, NUMBER OF TASKS=***n*

Explanation: The REORG TABLESPACE SHRLEVEL CHANGE or REFERENCE utility has determined that data partition and index log records can be applied to the shadow data sets in parallel. The number of tasks started is *n*.

System action: REORG processing continues with the LOG phase.

Severity: 0 (informational)

DSNU1163I   *csect-name* **- APPLYING PENDING DEFINITION CHANGES COMPLETE FOR** *database-name.object-name*.

Explanation: Pending definition changes are being applied.

csect-name
> The name of the control section that issued the message.

database-name.object-name
> The name of the object, qualified by the name of the database.

This message is for informational purposes only.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

DSNU1164I   *csect-name* **- PENDING DEFINITION CHANGES ARE NOT APPLIED FOR***database-name.object-name* **BECAUSE THE ENTIRE OBJECT WAS NOT SPECIFIED.**

Explanation: Pending definition changes at the table space or index level were not applied because the entire object was not specified. This message is issued in response to the REORG utility.

csect-name
> The name of the control section that issued the message.

database-name.object-name
> The name of the object, qualified by the name of the database.

System action: Utilities continue to process.

User response: Run the utility on the entire object to apply pending definition changes.

Severity: 4 (warning)

DSNU1165I *csect-name* **- REORG WITH SHRLEVEL NONE OR FASTSWITCH NO DOES NOT APPLY PENDING DEFINITION CHANGES for** *database-name.object-name***.**

Explanation: REORG with SHRLEVEL NONE reorganized the object but no pending definition changes were materialized.

csect-name
> The name of the control section that issued the message.

database-name.object-name
> The name of the object, qualified by the name of the database.

System action: Utilities continue to process.

User response: Run the REORG utility with SHRLEVEL REFERENCE or SHRLEVEL CHANGE against the object to materialize the pending definition changes.

Severity: 4 (warning)

DSNU1166I *csect-name* **- SOME PARTITION STATISTICS MAY HAVE BECOME OBSOLETE ON** *database-name.object-name***.**

Explanation: Some partition statistics might be no longer accurate.

csect-name
> The name of the control section that issued the message.

database-name.object-name
> The name of the object, qualified by the name of the database.

Partition statistics can become obsolete because of the materialization of pending definition changes. The following partition statistics might be obsolete:
- COLGROUP statistics
- KEYCARD statistics
- HISTOGRAM statistics
- Frequency statistics with NUMCOLS > 1
- Statistics for extended indexes, where applicable.

System action: Utilities continue to process.

User response: Run the RUNSTATS utility to collect the partition statistics again.

Severity: 4 (warning)

Related reference:

➡ RUNSTATS (DB2 Utilities)

DSNU1167I *csect-name* **- UNABLE TO ACCOMMODATE THE EXISTING USER DATA IN TABLESPACE** *database-name.tablespace-name* **BECAUSE THE NEW SEGSIZE OR THE NEW DSSIZE IS TOO SMALL.**

Explanation: The new SEGSIZE or the new DSSIZE in one of the pending ALTER definition changes is too small to accommodate the existing data.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

tablespace-name
> The name of the table space.

System action: Utility processing stops.

User response: Adjust the space attributes for the table space so that all existing data will fit, then run the utility again.

Severity: 8 (error)

DSNU1168I *csect-name* **- KEYWORD PART IS INVALID FOR RESETTING AREOR DBET STATE.**

Explanation: The keyword PART is not valid when NOAREORPEND is specified because the AREOR state can be reset only for the entire table space or index space.

csect-name
> The name of the control section that issued the message.

System action: Utility processing stops.

User response: Correct the utility statement.

Related reference:

➡ REPAIR (DB2 Utilities)

DSNU1169I *object-type object-name partition* **CONVERTED BY** *conversion-cause* **TO** *new-format* **RBA/LRSN FORMAT**

Explanation: LOAD, REBUILD, REORG, or RECOVER is converting the RBA or LRSN format of a database object.

object-type
> The object type: TABLESPACE or INDEXSPACE.

object-name
> The two-part object name.

partition
> PARTITION or DSNUM and an integer value. The value is omitted for nonpartitioned objects.

conversion-cause
> The cause of the conversion:

> **ZPARM**
>> Requested by UTILITY_OBJECT_CONVERSION subsystem parameter.

> **KEYWORD**
>> Requested by RBALRSN_CONVERSION keyword.

> **IMAGE COPY**
>> Restored from an image copy.

> **LOG LIMIT**
>> Forced by 6-byte logging limit.

new-format
> The new format of the object: BASIC or EXTENDED.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1170I *csect-name* **UNABLE TO EXTEND PARTITION-BY-GROWTH TABLESPACE** *database-name***.***tablespace-name* **BECAUSE OF** *reason*

Explanation: The partition-by-growth table space could not be extended.

csect-name
> The name of the control section that issued the message.

*database-name***.***tablespace-name*
> The name of the database and table space.

reason The reason that the partition could not be extended. *reason* can have one of the following values:

PARTITION RANGE

> The space that exists in the current range of the partition is not large enough to hold all of the data.

LOB COLUMNS

> Rows with LOB columns changed partitions during REORG. This change is not allowed. However, this partition change occurred, because a change has been made to the attributes of the table space so that a row with LOB columns cannot be reloaded to the partition to which it belongs.

MAXPARTITIONS LIMIT

> The partition could not be extended because the number of partitions has reached the maximum partitions limit.

System action: Utility processing is discontinued.

User response: Take one of the actions that corresponds to the *reason* value:

PARTITION RANGE

- Correct the space attributes of the partition to fit all the data that you need in that partition. Then, run the utility job again.
- Reorganize more partitions, especially those partitions with free space.

LOB COLUMNS

- In your REORG TABLESPACE utility control statement, set the AUX option to YES so that the associated LOB table spaces are also reorganized. Then, run the utility job again.

MAXPARTITIONS LIMIT

- Increase the maximum number of partitions by using ALTER TABLESPACE.

Related reference:

↪ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

↪ ALTER TABLESPACE (DB2 SQL)

DSNU1171I *csect-name*- **KEY EVALUATION ERROR OCCURRED DURING PROCESSING AT RID** *rid* **INDEX =** *index-name* **TABLE =** *table-name* **RECNO=** *input-dataset-record-number for-part* **=** *part-num*

Explanation: A key evaluation error occurred when the extended index was built. The record was not loaded.

System action: Processing continues.

Severity: 4 (warning)

DSNU1172I *csect-name*- **KEY EVALUATION ERROR OCCURRED DURING PROCESSING AT RID** *rid* **INDEX =** *index-name* **TABLE =** *table-name* **RECNO=** *input-dataset-record-number for-part* **=** *part-num*

Explanation: A key evaluation error occurred when the extended index was built. The record was not loaded.

System action: Processing terminates. An abend is requested so that the LOAD can be restarted after the error condition is corrected.

User response: Find the failing record in the input data set, correct the condition that caused the key evaluation error, and restart the LOAD utility. You can optionally include a SYSDISC DD statement in the JCL input to hold copies of records that are not loaded.

Severity: 8 (error)

DSNU1173I *csect-name* **TABLE SPACE** *database-name*.*space-name* **WILL BE CONVERTING FROM** *old-format* **TO** *new-format*

Explanation: REORG TABLESPACE or LOAD REPLACE is converting the row format of a table space or partition as specified.

csect-name

> The name of the control section that issued the message.

database-name
> The name of the database.

space-name
> The name of the table space or partitions.

old-format
> The row format before the conversion.

new-format
> The row format after the conversion.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1174I *csect-name* **TABLE SPACE** *database-name.space-name* **WILL REMAIN IN** *old-format* **DUE TO** *explanation-text*

Explanation:

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

space-name
> The name of the table space or partitions.

old-format
> The row format before the conversion attempt.

explanation-text
> The reason why the conversion did not complete.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1175I *csect-name* **INPUT FIELD(S) - CONTAIN ONE OR MORE INSTANCES WHICH WHEN CONVERTED TO DECFLOAT RESULTED IN ONE OF THESE CONDITIONS (OVERFLOW, UNDERFLOW, SUBNORMAL, INEXACT), AND THE RESULT VALUE MAY BE ROUNDED**

Explanation: The utility for LOAD or UNLOAD converts the input value, but the converted value is out of the normal range, and it has possibly been rounded.

System action: The utility for LOAD or UNLOAD continues processing.

Severity: 4 (warning)

DSNU1176I *csect-name* **- FUNCTION** *function* **IS NOT SUPPORTED FOR LOB OR XML TABLE SPACES**

Explanation: The function specified, *function*, is not supported for LOB/XML table spaces.

System action: Utility processing continues.

Programmer response: When using COPYDDN/RECOVERYDDN in LOAD for a table with LOB/XML columns, an inline copy is taken only of the base table space, not the LOB/XML table space. You must perform the tasks separately.

When using STATISTICS in LOAD for a table with LOB/XML columns, inline statistics are collected on the base table space, but not on the LOB/XML table space. You must perform the tasks separately.

Severity: 0 (informational)

| **DSNU1177I** *csect-name* **TABLE SPACE WILL BE LOADED IN PARALLEL, NUMBER OF TASKS =** *number-of-tasks*

| **Explanation:** This message is issued by the RELOAD phase of the LOAD utility. The message indicates that the
| LOAD utility is able to load the table space in parallel.

| *csect-name*
| The name of the control section that issued the message.

| *number-of-tasks*
| The number of utility tasks that are used to load the table space in parallel.

| If you specified the SHRLEVEL CHANGE option for LOAD, this number references the degree of both
| conversion and insertion parallelism. If you specified the SHRLEVEL NONE option, this number references
| only the degree of conversion parallelism.

| **System action:** Processing of the RELOAD phase continues.

| **User response:** You can verify that the reported number of tasks is consistent with your expectations.

| **Severity:** 0

| **Related reference**:

| ➡ LOAD (DB2 Utilities)

DSNU1179I *csect-name* **THE SQL STATEMENT EXCEEDS THE MAXIMUM ALLOWED LENGTH**

Explanation: The statement cannot be executed because it exceeds the system limits length.

System action: The statement cannot be executed.

User response: Break the statement into shorter, or less complex, SQL statements.

Severity: 8 (error)

DSNU1180I *csect-name* **SQLCODE=0, SUCCESSFUL EXECUTION**

Explanation: The SQLCODE was 0, indicating successful execution of the function. SQLWARN indicators might note some warnings.

csect-name
        The name of the control section that issued the message.

System action: Utility processing continues.

Operator response: No action is required.

Severity: 0

Related concepts:

➡ SQL communication area (SQLCA) (DB2 SQL)

Related reference:

➡ Description of SQLCA fields (DB2 SQL)

DSNU1184I *csect-name* **- SQLCODE =**ize*sqlcode*, *explanation*

Explanation: This message contains an SQL return code and a brief explanation, with the text inserted from the SQLERRM field. The SQL return code is positive, indicating a warning condition.

System action: Utility processing continues.

Severity: 4 (warning)

Related concepts:

Successful execution SQL codes (DB2 Codes)

Related reference:

Description of SQLCA fields (DB2 SQL)

DSNU1188I   *csect-name* **- SQLCODE = -***sqlcode***,** *explanation*

Explanation: This message contains an SQL return code and a brief explanation, with the text inserted from the SQLERRM field. The SQL return code is negative, indicating that an error has occurred.

System action: Further utility processing is terminated.

User response: Determine the cause for the SQL error by using information about that specific SQLCODE. Correct the error. Rerun the application program or SQL statement.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- Listing of the results produced by the SQL statements.
- Source listing of the failing application program.

Severity: 8 (error)

Related concepts:

SQL error codes (DB2 Codes)

Related reference:

Description of SQLCA fields (DB2 SQL)

DSNU1189I   *csect-name* **- CURSOR** *cursor-name* **ALREADY DECLARED**

Explanation: The indicated *cursor-name* was declared previously in the SYSIN data set.

System action: Utility processing is not initiated.

User response: Eliminate the duplicate DECLARE CURSOR and resubmit the utility job.

Severity: 8 (error)

Related reference:

EXEC SQL (DB2 Utilities)

Syntax and options of the EXEC SQL control statement (DB2 Utilities)

DSNU1190I   *csect-name* **- CURSOR** *cursor-name* **NOT DECLARED**

Explanation: A utility control statement used a cursor that was not found in SYSIN.

System action: Utility processing is not initiated.

User response: Provide the missing DECLARE CURSOR in SYSIN prior to its use and resubmit the utility job.

Severity: 8 (error)

Related reference:

EXEC SQL (DB2 Utilities)

Syntax and options of the EXEC SQL control statement (DB2 Utilities)

DSNU1191I *csect-name* **SQLCAID =** *xxxx***, SQLCA IDENTIFIER**

Explanation: The SQLCA ID field (SQLCAID) is not the expected value, C'SQLCA'.

System action: Utility processing continues.

User response: The SQLCA area is invalid.

Problem determination: This error is an internal error.

Severity: 0 (informational)

DSNU1192I *csect-name* **SQLCABC =** *xxxx***, SQLCA BYTE COUNT**

Explanation: The size of the SQLCA is not the expected value (136 bytes).

System action: Utility processing continues.

User response: The SQLCA area is invalid.

Problem determination: This error is an internal error.

Severity: 0 (informational)

DSNU1193I *csect-name* **SQLCODE =** *xxxx***, INVALID SQLCODE**

Explanation: The SQL return code is not a valid value.

System action: Utility processing continues.

User response: Either the SQLCA area is invalid or the maintenance was not coordinated correctly. As a result, the SQL return code was added to the DB2 subsystem but not to the formatting routine. If the maintenance was not coordinated correctly, the DSNTIAM CSECT was probably not replaced in the current STEPLIB or JOBLIB.

Problem determination: If DSNTIAM is current then this error is an internal error.

Severity: 0 (informational)

DSNU1194I *csect-name* **SQLERRM =** *xxxx***, SQLCA ERROR MESSAGE INSERTION TEXT**

Explanation: This message results from an SQL return code that is not recognized by DSNTIAR. SQLERRM contains error message test.

System action: Utility processing continues.

User response: Either the SQLCA area is invalid or the maintenance was not coordinated correctly. As a result, the SQL return code was added to the DB2 subsystem but not to the formatting routine. If the maintenance was not coordinated correctly, the DSNTIAM CSECT was probably not replaced in the current STEPLIB or JOBLIB.

Problem determination: If DSNTIAM is current then this error is an internal error.

Severity: 0 (informational)

DSNU1195I *csect-name* **SQLERRP =** *xxxx***, SQL PROCEDURE DETECTING ERROR**

Explanation: The name of the procedure detecting the error is provided in the SQLERRP.

System action: Utility processing continues.

User response: Correct the error if one is noted in other messages. The SQL procedure detecting the error might be of help in diagnosing an error or warning. Rerun the SQL statement.

Problem determination: If DSNTIAM is current then this error is an internal error.

Severity: 0 (informational)

DSNU1196I **SQLERRD =** *field-1***,** *field-2***,** *field-3***,** *field-4***,** *field-5***,** *field-6* **SQL DIAGNOSTIC INFORMATION**

Explanation: One or more of the six diagnostic fields do not have the expected value. The diagnostic fields should contain one of the following expected sets of values:

- 00, 00, 00, 00, 00, 00
- 00, 00, 00, -1, 00, 00

field-1   DB2 unique internal relational data system code.

field-2   DB2 internal data manager error code.

field-3   One of the following conditions:

- Number of rows inserted, updated, or deleted following an INSERT, UPDATE, MERGE or DELETE statement.

    Attention: A -1 in this field indicates that a mass delete of a table in a segmented table space was performed, wherein DB2 does not access nor count individual rows.

- For SQLCODE -911 or -913, a reason code that indicates whether a deadlock or timeout occurred.

    Tip: The most common reason codes are:
    - 00C90088 - deadlock
    - 00C9009E - timeout

field-4   Reserved.

field-5   Position or column of a syntax error for PREPARE or EXECUTE IMMEDIATE.

field-6   DB2 internal buffer manager error code.

System action: Utility processing continues.

User response: The SQL diagnostic information might help in diagnosing an error or in indicating the results of successful execution. If this error or warning occurs, correct the error. Rerun the program or SQL statement.

Severity: 0

Related reference:

⏩ GET DIAGNOSTICS (DB2 SQL)

⏩ Description of SQLCA fields (DB2 SQL)

Related information:

⏩ -911 (DB2 Codes)

⏩ -913 (DB2 Codes)

⏩ 00C90088 (DB2 Codes)

⏩ 00C9009E (DB2 Codes)

DSNU1197I **SQLWARN***n-n* **SQL WARNINGS**

Explanation: At least one of the eleven (11) warning values is not the expected value (a blank). A nonblank value in one of the 11 warning fields has one of the following meanings:

SQLWARN0
> Any other warning code is set.

SQLWARN1
> String truncation.

SQLWARN2
> Null values eliminated.

SQLWARN3
> Number of host variables provided did not equal degree of the table.

SQLWARN4

No WHERE clause on UPDATE or DELETE.

SQLWARN5

Not a DB2 SQL statement.

SQLWARN6

Date or timestamp value adjusted.

SQLWARN7

Conversion of decimal number from precision 31 to precision 15 caused truncation of fractional digits.

SQLWARN8

Untranslatable character replaced with substitute.

SQLWARN9

Arithmetic exceptions have been ignored during processing of COUNT DISTINCT.

SQLWARNA

At least one character field of the SQLCA is invalid due to a string translation error.

System action: Utility processing continues.

User response: The SQL warning information might be of help in diagnosing an error or in indicating the results of successful execution. If this warning should occur, correct the error and rerun the program or SQL statement.

Severity: 0 (informational)

DSNU1198I **SQLSTATE =** *sqlstate*, **SQLSTATE RETURN CODE**

Explanation: The SQLSTATE is a return code that indicates the outcome of the most recently executed SQL statement. The execution of every SQL statement sets SQLSTATE to a five-digit code in the range of 00000 to 65535. The SQLSTATE has no effect on the existing use of any other field in the SQLCA.

System action: Utility processing continues.

User response: Use the return code value to determine the problem. If necessary, correct the SQL statement and issue it again.

Severity: 0 (informational)

Related tasks:

Checking the execution of SQL statements by using SQLCODE and SQLSTATE (DB2 Application programming and SQL)

Related reference:

SQLSTATE values and common error codes (DB2 Codes)

SQLSTATE (DB2 SQL)

DSNU1199I *csect-name text*

Explanation: Unexpected message was returned from DSNTIAR.

System action: Utility processing continues.

User response: This error is an internal error.

Severity: 0 (informational)

DSNU1201I *csect-name* **- PARTITIONS WILL BE UNLOADED IN PARALLEL, NUMBER OF TASKS =** *nnnn*

Explanation: The message indicates that the UNLOAD utility is able to unload partitions in parallel. The number of utility tasks used to unload partitions is indicated by *nnnn*.

System action: Processing of the UNLOAD phase will continue.

Severity: 0 (informational)

DSNU1202I *csect-name* **- SAMPLING LIMIT HAS BEEN REACHED FOR TABLE** *table-name*

Explanation: In the UNLOAD statement, the LIMIT option is specified for table *table-name* and the number of unloaded rows from the table has reached the specified LIMIT value.

System action: No more rows will be unloaded from the table, but the unloading operation will continue for the previously unloaded tables if they exist.

Severity: 0 (informational)

DSNU1203I *csect-name* **- GENERATION OF LOAD STATEMENT IS BYPASSED FOR TABLE** *table-name*

Explanation: The PUNCHDDN specification is omitted in a UNLOAD statement when partition parallelism is activated. The LOAD statement will not be generated for the indicated table.

System action: The UNLOAD job will continue.

Severity: 0 (informational)

DSNU1204I **DUPLICATE ROWS MIGHT HAVE BEEN UNLOADED**

Explanation: The image copy contains duplicate pages. All rows from the duplicate pages are unloaded

System action: Processing continues.

Severity: 4 (warning)

DSNU1211I *csect-name* **- INPUT DATA SET IS NOT AN IMAGE COPY**

Explanation: The input data set specified by the FROMCOPY or FROMCOPYDDN option of the UNLOAD utility was not identified as an image copy data set.

System action: The utility job terminates.

User response: Specify the correct image copy data set name to the FROMCOPY option, or to the DD-name that is referred to by the FROMCOPYDDN option.

Severity: 8 (error)

DSNU1212I *csect-name* **- INVALID FIELD NAME** *field-name* **FOR TABLE** *table-name*

Explanation: The specified field name in the field specification list of the UNLOAD utility is:
* Not a column name of the specified table, or
* An existing column name for the CONSTANT option.

The field name is not valid.

System action: The utility job terminates.

User response: For a CONSTANT field, specify a non-existing field name. Otherwise, specify the field name as a column name of the indicated table.

Severity: 8 (error)

DSNU1213I *csect-name* **- INVALID FIELD TYPE FOR FIELD** *field-name* **FOR TABLE** *table-name*

Explanation: The specified field type in the field specification list of the UNLOAD utility is incompatible with the base type of the corresponding column of the table.

System action: The utility job terminates.

User response: Specify only a compatible type allowed for the unload operation in the field specification list.

Severity: 8 (error)

DSNU1214I • DSNU1219I

DSNU1214I *csect-name* **- INVALID FIELD POSITION SPECIFICATION FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: For a UNLOAD statement, specified field positions are not in ascending order in the field specification list of the indicated table.

System action: The utility job terminates.

User response: Specify the field positions in the ascending order in the field specification list. If a certain order of the output fields is needed, change the order of the field in the list so that the field positions are in the ascending order.

Severity: 8 (error)

DSNU1217I *csect-name* **- CONFLICTING FIELD POSITION AND/OR LENGTH SPECIFICATION FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: For a UNLOAD statement, the specified field position for the indicated field caused a conflict with the previous field. This is caused either by

• The position value specified for the field is too small.

• The length specification for the previous field is too large.

System action: The utility job terminates.

User response: Specify the correct position or length parameters in the field specification list so that there are enough space for the output fields.

Severity: 8 (error)

DSNU1218I *csect-name* **- LOGICAL RECORD LENGTH OF OUTPUT RECORD EXCEEDED THE LIMIT FOR TABLE** *table-name*

Explanation: The maximum length of the output records exceeded the physical limit for the UNLOAD utility.

System action: The utility job terminates.

User response: Consider one or more of the following options that can reduce the length of the output record, if possible:

• Use the NOPAD option.

• Unload only the necessary fields using the field specification options.

• Specify the LENGTH option with TRUNCATE and/or STRIP options for the applicable fields.

If the above options can not be applied, you will have to use an application to unload the data from the table.

Problem determination: This problem will most likely occur when BLOB, CLOB, or DBCLOB columns are selected from a table to be unloaded.

Severity: 8 (error)

DSNU1219I *csect-name* **- THE NUMBER OF RECORDS IN ERROR REACHED THE LIMIT** *nnnn*

Explanation: For the UNLOAD utility, the number of records in error reached the specified MAXERR limit.

System action: The utility job terminates.

User response: Investigate the individual error message and determine the cause of the errors. The corrective action will differ by the type of the error. In certain situations, this condition may be anticipated. In this case, the total number of records unloaded before the job terminates can be adjusted by changing the MAXERR option value.

Problem determination: This problem will occur when the number of errors in the output record formatting, including data conversion errors, detected by the UNLOAD utility reached the limit specified by the MAXERR option. For each record in error, the corresponding message has been issued. The records in error will not be unloaded to the output data set. This message is issued only once, even though there can be a situation in which the number of individual error messages exceeds the specified MAXERR value. This situation occurs only when the UNLOAD utility is running in the partition parallel mode and the multiple tasks have reached the MAXERR value concurrently.

Severity: 8 (error)

DSNU1220I   **UNLOAD OF** *lob-or-xml* **FROM IMAGE COPY NOT SUPPORTED**

Explanation: For the UNLOAD utility, one or more XML, BLOB, CLOB, and/or DBCLOB columns are selected on the indicated table when the source object is an image copy data set.

System action: The utility job terminates.

User response: Specify a field selection list that does not include any XML, BLOB, CLOB, or DBCLOB column from the indicated table.

Problem determination: When the source object of the UNLOAD utility is an image copy, selection of a LOB or XML column is not supported.

Severity: 8 (error)

DSNU1221I   *csect-name* **- UNLDDN IS NOT SPECIFIED OR SPECIFIED TEMPLATE** *template-name* **IS INVALID FOR MULTIPLE TABLE SPACES**

Explanation: For the UNLOAD utility, the output data set specification is invalid when the source objects consist of multiple table spaces.

System action: The utility job terminates.

User response: Include an appropriate TEMPLATE statement in the unload job step that contains the table space name as a variable. If the table space is selected from different databases, the database name must also be included as a variable. Specify the template name to the UNLDDN option.

Problem determination: When the source objects of the UNLOAD utility consist of multiple table spaces, an output data set corresponding to each table space is required.

Severity: 8 (error)

DSNU1222I   *csect-name* **- TEMPLATE** *template-name* **SPECIFIED FOR PUNCHDDN IS INVALID FOR MULTIPLE TABLE SPACES**

Explanation: For the UNLOAD utility, the specification of output data set to store the generated LOAD statement is invalid when the source objects consist of multiple table spaces.

System action: The utility job terminates.

User response: Include an appropriate TEMPLATE statement in the unload job step that contains the table space name as a variable. If the table space is selected from different databases, the database name must also be included as a variable. Specify the template name to the PUNCHDDN option. The TEMPLATE should be different from the one used for UNLDDN.

Problem determination: When the source objects of the UNLOAD utility consist of multiple table spaces, an output data set corresponding to each table space is required.

Severity: 8 (error)

DSNU1223I   *csect-name* **- TABLESPACE SPECIFICATION IS MISSING FOR FROMCOPY OR FROMCOPYDDN**

Explanation: For the UNLOAD utility, the FROMCOPY or FROMCOPYDDN option is used but the TEMPLATE specification is missing.

System action: The utility job terminates.

User response: Add the TABLESPACE option in the UNLOAD statement to specify the table space name of the image copy.

Problem determination: When the source objects of the UNLOAD utility is an image copy, the TABLESPACE option is required to specify the table space name of the image copy.

Severity: 8 (error)

DSNU1224I   *csect-name* **- TABLE** *table-name* **IS NOT IN THE TABLE SPACE** *table-space-name*

Explanation: The indicated *table-name* given in a FROM TABLE option is not in the table space specified by the TABLESPACE option.

System action: The utility job terminates.

User response: Correct either the TABLESPACE or the FROM TABLE specification to select a table in the specified table space. If the table was dropped, remove the FROM TABLE option for the dropped table.

Problem determination: There is an unmatch between the TABLESPACE and FROM TABLE specifications or the specified table was dropped. Dropped tables are not supported by the UNLOAD utility.

Severity: 8 (error)

DSNU1225I   *csect-name* **- TABLESPACE KEYWORD IS NOT SPECIFIED. AT LEAST ONE FROM TABLE SPECIFICATION IS REQUIRED.**

Explanation: There is no TABLESPACE nor FROM TABLE specification in a UNLOAD statement. The source object for unloading can not be identified.

System action: The utility job terminates.

User response: If the TABLESPACE specification is omitted, specify at least one FROM TABLE option to identify the source object. Instead of a TABLESPACE specification, the LIST option can be used, if a LISTDEF statement defines a list of objects. When the LIST option is used, the FROM TABLE option can not be specified.

Severity: 8 (error)

DSNU1226I   *csect-name* **- THE NUMBER OF OBJECTS IS TOO LARGE**

Explanation: The number of source objects for the UNLOAD utility is too large. Overflow in some of the internal data structures occurred.

System action: The utility job terminates.

User response: Divide the UNLOAD job step into two or more job steps and, in each job step, use a LISTDEF that expands to less number of objects.

Problem determination: This problem will most likely occur when the LIST option is used and the corresponding LISTDEF definition is expanded to a large number of objects.

Severity: 8 (error)

DSNU1227I   *csect-name* **- THE CONTENT OF THE IMAGE COPY DATA SET IS INCONSISTENT WITH TABLE SPACE** *table-space*

Explanation: For the UNLOAD utility, the data set that is specified by the FROMCOPY or the FROMCOPYDDN option references a data set that does not match the definition of the table space in the catalog.

csect-name
>   The name of the control section that issued the message.

table-space
>   Name of the table space.

System action: The utility job terminates.

User response: Perform one of the following actions:

- Specify a correct image copy data set name for the table space
- Specify the correct table space name for the image copy

In the case where the image copy was taken before a REORG where dropped columns were materialized, UNLOAD is not permitted.

Severity: 8 (error)

DSNU1228I *csect-name* **- A ROWID COLUMN** *column-name*, **TABLE** *table-name* **IS SELECTED FOR UNLOAD BUT THE PAGE SET HEADER PAGE IS MISSING IN THE SPECIFIED IMAGE COPY DATA SET**

Explanation: For the UNLOAD utility, a ROWID column is selected for unloading from an image copy, but the required information to form the ROWID can not be retrieved because the page set header page is not included in the specified image copy data set.

System action: The utility job terminates.

User response: Use the FROMCOPYDDN, instead of FROMCOPY, to specify a DD name where the first piece of copy that contain the page set header page is concatenated to the source image copy data set.

Problem determination: The problem will most likely occur when a piece of copy data set, not the first one, is specified to the FROMCOPY option.

Severity: 8 (error)

DSNU1229I *csect-name* **- ERROR IN LOADING DICTIONARY FOR TABLE SPACE** *table-space*

Explanation: For the UNLOAD utility, an error was detected in loading the dictionary for decompression while reading an image copy data set.

System action: The utility job terminates.

User response: If the right image copy data set is specified to the FROMCOPY or FROMCOPYDDN, it will be required to take a new image copy for the table space by running the COPY utility or using other alternative methods.

Problem determination: Possibly the image copy data set is corrupted.

Severity: 8 (error)

DSNU1230I *csect-name* **- INCONSISTENT STRIP CHAR LENGTH FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: An invalid strip character was specified for the indicated field of the table.

System action: The utility job terminates.

User response: Specify the correct strip character applicable to the field.

Problem determination: For an SBCS or MIXED character field, the strip character, if specified, must be an SBCS character; for a DBCS field, it must be a DBCS character.

Severity: 8 (error)

DSNU1231I *csect-name* **- LENGTH OF OUTPUT RECORD IS TOO LONG FOR TABLE** *table-name*

Explanation: An output record for the indicated table exceeded the physical limit.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until the total number of records in error reaches the value specified by the MAXERR option (the default is 1).

If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

User response: Consider the following options to reduce the size of the output records:
- Unload only the necessary fields using the field specification options.
- Specify the LENGTH option with TRUNCATE and/or STRIP options for the applicable fields.

If the above options can not be applied, you will have to use an application to unload the data from the table.

Problem determination: This message is issued only when the NOPAD option is specified in the UNLOAD statement and most likely when BLOB, CLOB, or DBCLOB columns are selected for unload from the table. It is also possible that, if the expansion of output strings occurs due to the encoding scheme specified for the output, the total length of some output records becomes longer than expected.

Severity: 4 (warning)

DSNU1232I   *csect-name* **- COMPRESSED ROW IS IGNORED BECAUSE THE DICTIONARY IS NOT AVAILABLE FOR TABLE** *table-name*

Explanation: The UNLOAD utility encountered a compressed row when unloading the data from an image copy data set, but the dictionary for decompression has not been loaded for the table space.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

This message is issued when:

* A copy of pieces (not the first piece) of a compressed table space is specified in the FROMCOPY or FROMCOPYDDN option of the UNLOAD utility.
* The image copy data set in the FROMCOPY or FROMCOPYDDN option of the UNLOAD utility was built either by the COPY utility with the SYSTEMPAGES NO option or by the DSN1COPY utility, and the original compressed table space was filled by INSERT statements and never reorganized.

System action: The record in error is not unloaded to the output data set. The UNLOAD utility continues to unload records until the total number of records in error reaches the value specified by the MAXERR option. (The default value of this option is 1.) If the total number of records reaches the MAXERR value, the UNLOAD utility issues message DSNU1219I and terminates with return code 8.

User response: To unload compressed rows from the copy of pieces, use the FROMCOPYDDN option to specify a DD name under which the first copy of pieces that contains the compression dictionary is concatenated.

If the problem persists, it is likely that the image copy data set was built in such a way that the data in the image copy data set cannot be unloaded. (Either the COPY utility was used with the SYSTEMPAGES NO option or the DSN1COPY utility was used, and the original compressed table space was filled by INSERT statements and never reorganized.)

This copy is not suitable for use with the UNLOAD utility. Create a new copy with the SYSTEMPAGES YES option and unload data from the new copy.

Severity: 4 (warning)

Related reference:

➥ UNLOAD (DB2 Utilities)

Related information:
"DSNU1219I" on page 866

DSNU1233I   *csect-name* **- DATA IS TOO LONG FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: The space for the field in an output record is not enough to unload the data from the corresponding column of a row in the indicated table.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until the total number of records in error reaches the value specified by the MAXERR option (the default is 1).

If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

User response: Apply appropriate values to the LENGTH and/or POSITION parameters in the field specification list to accommodate the data in the output records. Alternatively, specify TRUNCATE option to the field, if applicable. Specifying the STRIP option, if applicable, can also reduce the length of data if the data contains leading or trailing blanks.

Problem determination: This message is issued when the space of a field in an output record is limited by the LENGTH specification for the field or the POSITION specification for the next field.

Severity: 4 (warning)

DSNU1234I *csect-name* **- NUMERIC CONVERSION ERROR OCCURRED FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: The UNLOAD utility encountered an error when it tried to convert a data of one numeric type to another.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until the total number of records in error reaches the value specified by the MAXERR option (the default is 1).

If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

User response: To eliminate any numeric overflow error, specify the same numeric data type for the output as the base type of the columns in the table. Convert numeric types only when it is known that the range of the source data fits in the allowable range of the specified numeric type for the output.

Problem determination: This message is issued most likely when an overflow occurs in converting a data of one numeric type to another. For example, an attempt to convert an INTEGER 32768 to a SMALL INT causes an overflow.

Severity: 4 (warning)

DSNU1235I *csect-name* **- CCSID SUBSTITUTION IS ATTEMPTED WITH NOSUBS OPTION FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: While the UNLOAD utility is running with the NOSUBS option, a CCSID substitution is attempted in the CCSID conversion process for the indicated field.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until the total number of records in error reaches the value specified by the MAXERR option (the default is 1). If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

User response: If CCSID substitution should not occur, specify an appropriate CCSID that has the corresponding entries to the original CCSID, if available. To allow the default CCSID substitution, remove the NOSUBS option.

Severity: 4 (warning)

DSNU1236I *csect-name* **- POSITION SPECIFICATION CAUSES FIELD OVERLAP FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: For the UNLOAD utility, the space for an output field was overlapped by the indicated field to which the POSITION parameter is specified.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until the total number of records in error reaches the value specified by the MAXERR option (the default is 1). If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

User response: Consider the following options to eliminate this situation:
- Remove the POSITION specifications from the field specification list.
- Adjust the POSITION parameter for the field to preserve enough space for the previous fields.
- Adjust the LENGTH, possibly with the TRUNCATE option, of the previous fields, if applicable.

Problem determination: This message will be issued only when the NOPAD option is specified in the UNLOAD statement. The POSITION specification for the field caused a conflict with the positioning and/or the length of a previous field.

Severity: 4 (warning)

DSNU1237I *csect-name* **- ERROR IN CCSID CONVERSION FOR FIELD** *field-name*, **TABLE** *table-name*

Explanation: The CCSID conversion process detected a code point or form exception while the UNLOAD utility was running.

System action: The record in error will not be unloaded to the output data set. The UNLOAD utility continues until

the total number of records in error reaches the value specified by the MAXERR option (the default is 1). If the total number of records reaches the MAXERR value, the UNLOAD utility issues the DSNU1219I message and terminates with the return code 8.

System programmer response: Verify the contents of SYSSTRINGS. If necessary, insert a row into SYSSTRINGS for the conversion. For an SBCS translation, the row must contain a translate table or the name of a user-provided translate procedure. For a DBCS translation, the row must contain the name of a user-provided translate procedure.

User response: If the specified CCSID for output is incorrect, correct the CCSID specification. If a valid CCSID is specified, notify the system programmer.

Severity: 4 (warning)

DSNU1238I **TABLE** *table2* **IS NOT IN SAME TABLESPACE AS TABLE** *table1*

Explanation: All source tables for a single UNLOAD utility statement must be in the same table space.

System action: Utility processing terminates.

User response: Remove the FROM TABLE control statement that specifies *table2* from the current UNLOAD request. Create another UNLOAD utility statement that contains the FROM TABLE control statement for *table2* so that all source tables in a single UNLOAD request are from the same table space. Resubmit the job.

Severity: 8 (error)

DSNU1239I *csect-name* **- VERSION** *version-number* **OBDREC WAS NOT FOUND IN OBID** *object-id*

Explanation: The version information for a record that is being unloaded from an image copy is not available in the OBD or the system pages in the image copy.

csect-name
> The name of the control section that issued the message.

version-number
> The version number of the record in error.

object-id
> The object ID of the record in error.

Unloading data from an image copy requires that the image copy data set was taken with the SYSTEMPAGES YES option on the COPY utility statement. Without the system pages, the UNLOAD utility cannot unload versioned data.

Possible causes for this error include:
* The system pages are not available in the image copy.
* The version 0 OBD information has been deleted by the MODIFY RECOVERY utility.

System action: The record that is in error is not unloaded to the output data set. The UNLOAD utility continues to process records. If the total number of records in error reaches the value that is specified by the MAXERR option, processing stops and message DSNU1219I is issued.

The default value for MAXERR is 1.

User response: If the DDL of the table, including alterations up to the time of the image copy, is available, you can rebuild the OBD information and unload the data. To rebuild OBD information:

1. Re-create the table to the state that it was in when the image copy was taken.
2. If necessary, use DSN1COPY with the IBDIXLAT option to update the DBID, PSID, and OBID information in the image copy so that it matches the new table.
3. Unload the data from the updated image copy.

Severity: 4 (warning)

Related reference:

⇨ UNLOAD (DB2 Utilities)

⇨ MODIFY RECOVERY (DB2 Utilities)

Related information:

"DSNU1219I" on page 866

DSNU1241I   *csect-name* **-***utility-name* **UTILITY CANNOT PROCEED BECAUSE TABLE** *table-name* **IS SPECIFIED IN MORE THAN ONE FROM TABLE SPECIFICATION**

Explanation: The table *table-name* cannot be specified in multiple FROM TABLE specifications.

System action: Utility processing terminates.

User response: Remove the FROM TABLE specifications for the same table. Group all WHEN control statements for the same table into a single WHEN control statement by using the OR keyword. Resubmit the job.

Severity: 8 (error)

DSNU1242I   *csect-name* **ALL KEYS OF A NON-PARTITIONED SECONDARY INDEX WILL BE SORTED.**

Explanation: During a part-level REORG, all keys of a non-partitioned secondary index are sorted before rebuilding the index. This action is controlled by the SORTNPSI keyword and the REORG_PART_SORT_NPSI subsystem parameter. When you specify SORTNPSI as YES or AUTO, all keys are sorted. When you set REORG_PART_SORT_NPSI to YES or AUTO and you do not specify SORTNPSI, all keys are sorted.

System action: Processing continues.

Severity: 0

DSNU1250I   *csect-name* **- ERROR IN CCSID CONVERSION FOR** *object-type*. **SOURCE CCSID=***ccsid*, **TARGET CCSID=***ccsid*

Explanation: The CCSID conversion process failed for *object-type*. The values of *object-type* and the cause of the error for each type are:
- SBCS PAD CHAR
- DBCS PAD CHAR
- VERIFICATION
- HEADER CONST
- *field-name*

System action: The utility terminates.

System programmer response: Check the SYSSTRINGS catalog table to determine whether a row exists for the CCSID conversion. If a row does not exist, and OS/390 V2R9 is installed, refer to *z/OS C/C++ Programming Guide* for additional conversions that are supported. If no conversions are available, contact your IBM service representative to request support for the conversion.

User response: If the CCSID that is specified in the utility control statement is incorrect, correct the CCSID specification. If a valid CCSID is specified, notify the system programmer.

Severity: 8 (error)

DSNU1251I   *csect-name* **- ERROR IN CCSID CONVERSION FOR** *object-type* **IN FIELD** *field-name* **OF TABLE** *table-name*. **SOURCE CCSID=***ccsid*, **TARGET CCSID=***ccsid*

Explanation: The CCSID conversion process failed for *object-type* . The values of *object-type* and the cause of the error for each type are:
- STRIP CHAR

System action: The utility terminates.

System programmer response: Check the SYSSTRINGS catalog table to determine whether a row exists for the

CCSID conversion. If a row does not exist, and OS/390 V2R9 is installed, refer to *z/OS C/C++ Programming Guide* for additional conversions that are supported. If no conversions are available, contact your IBM service representative to request support for the conversion.

User response: If the CCSID that is specified in the utility control statement is incorrect, correct the CCSID specification. If a valid CCSID is specified, notify the system programmer.

Severity: 8 (error)

DSNU1252I *csect-name* **- PARTITION PARALLELISM IS NOT ACTIVATED FOR TABLESPACE** *table-space-name*. **THE PARTITION VARIABLE IN THE TEMPLATE DSN WAS SPECIFIED BUT NO PARTITION IS ACTIVATED**

Explanation: A utility control statement for the UNLOAD utility contains a template name in the UNLDDN option. The template contains a variable that represents a partition number. However, the source object is either a non-partitioned table space or an image copy that does not support partition parallelism.

System action: UNLOAD processing continues. For the indicated table space, a single output data set is allocated, and the unloaded data is placed in that data set.

Severity: 4 (warning)

DSNU1253I *csect-name* **- USER** *user-id* **DOES NOT HAVE THE SELECT PRIVILEGE ON TABLE** *table-name* **IN TABLE SPACE** *table-space-name*

Explanation: User *user-id* attempted to run a utility that requires the SELECT privilege on table *table-name*. The user does not have that privilege.

System action: The utility terminates.

System programmer response: Grante the user the SELECT privilege on table *table-name* in table space *table-space-name*.

Severity: 8 (error)

DSNU1254I *csect-name* **FORMAT DELIMITED REQUIRES ONE AND ONLY ONE FROM TABLE**

Explanation: UNLOAD FORMAT DELIMITED supports unloading only from a single table. FORMAT DELIMITED was specified either with:

- Multiple FROM TABLE specifications, or
- No FROM TABLE specifications, with a table space that contains more than one table.

System action: Processing terminates.

User response: Correct the UNLOAD statement and resubmit the utility.

Severity: 8 (error)

DSNU1255I *csect-name* **UNLOAD IS NOT ALLOWED BECAUSE THE SOURCE IS MISSING THE HEADER PAGE**

Explanation: An attempt was made to unload data from a sequential copy data set that does not contain either the header page of the table space or the header page of the partition.

System action: The UNLOAD utility terminates.

Programmer response: Use an image copy data set that contains the header page.

Severity: 8 (error)

DSNU1256I *csect-name* **- VIRTUAL STORAGE REQUIRED FOR SPANNED RECORDS EXCEEDS DEFINED LIMIT.**

Explanation: The amount of virtual storage required to process spanned records has exceeded the limit defined on the system.

csect-name
>   The name of the control section that issued the message.

To calculate the storage limit, multiply the number of LOB columns by the value of the LOBVALA subsystem parameter, and the number of XML columns by the value of the XMLVALA subsystem parameter, and sum the two products:

x = number of LOB columns × LOBVALA value
y = number of XML columns × XMLVALA value
limit = $x + y$

System action: The utility job terminates.

User response: Either modify the utility statement to reduce the size of the records or increase the amount of virtual storage allowed for spanned records processing. To increase storage limit, adjust the LOBVALA or XMLVALA values as appropriate for your environment.

Severity: 8 (error)

Related reference:

⮕ USER LOB VALUE STG field (LOBVALA subsystem parameter) (DB2 Installation and Migration)

⮕ USER XML VALUE STG field (XMLVALA subsystem parameter) (DB2 Installation and Migration)

DSNU1257I   *csect-name* **- REFERENCE TO LOB OR XML DATA IN A WHEN CLAUSE IS NOT ALLOWED IF THE LOAD STATEMENT SPECIFIES FORMAT SPANNED YES.**

Explanation: A WHEN clause of a LOAD statement references data that is not in the first segment of a spanned record.

csect-name
>   The name of the control section that issued the message.

System action: The utility job terminates.

User response: Modify the WHEN clause to remove references to data beyond the first segment of a spanned record.

Severity: 8 (error)

Related reference:

⮕ LOAD (DB2 Utilities)

DSNU1258I   *csect-name* **- ORDER OF FIELDS IS NOT VALID FOR KEYWORD SPANNED.**

Explanation: LOB and XML fields were found in an invalid location. These fields must be at the end of the field specification list for spanned records.

csect-name
>   The name of the control section that issued the message.

System action: The utility job terminates.

User response: Modify the field specification list.

Severity: 8 (error)

DSNU1259I   *csect-name* **- THE INPUT DATA SET DOES NOT HAVE THE ATTRIBUTE SPANNED.**

Explanation: The utility specified SPANNED YES, but the data set is not spanned.

csect-name
>   The name of the control section that issued the message.

System action: The utility job terminates.

User response: Modify the record format of the input data set.

Severity: 8 (error)

DSNU1271I *csect-name* **REQUIRED KEYWORD NUMRECS MISSING FOR TABLE** *table-name*

Explanation: An attempt was made to use the LOAD utility to load data into multiple tables. If any table that is to be loaded specifies the NUMRECS keyword, then all tables in the LOAD operation must specify the NUMRECS keyword.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table that is missing the NUMRECS keyword.

System action: Utility operation is terminated with return code 8. The tables are not loaded.

User response: Add the required keyword NUMRECS to all INTO TABLE clauses of the LOAD utility control statement, and resubmit the job.

Severity: 8 (error)

Related reference:

↪ LOAD (DB2 Utilities)

DSNU1300I *csect-name* **MODIFY STATISTICS COMPLETED SUCCESSFULLY**

Explanation: This message reports the successful completion of the MODIFY STATISTICS utility operation.

Severity: 0 (informational)

DSNU1301I *csect-name number-deleted* **SYSIBM.SYSCOLDIST_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSCOLDIST_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1302I *csect-name number-deleted* **SYSIBM.SYSCOLUMNS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSCOLUMNS_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1303I *csect-name number-deleted* **SYSIBM.SYSINDEXES_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSINDEXES_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1304I *csect-name number-deleted* **SYSIBM.SYSINDEXPART_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSINDEXPART_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1305I *csect-name number-deleted* **SYSIBM.SYSINDEXSTATS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSINDEXSTATS_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1306I *csect-name number-deleted* **SYSIBM.SYSLOBSTATS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSLOBSTATS_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
> The name of the control section that issued the message.

number-deleted
> The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1307I   *csect-name number-deleted* **SYSIBM.SYSTABLEPARTS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSTABLEPARTS_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
>   The name of the control section that issued the message.

number-deleted
>   The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1308I   *csect-name number-deleted* **SYSIBM.SYSTABLES_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSTABLES_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
>   The name of the control section that issued the message.

number-deleted
>   The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1309I   *csect-name number-deleted* **SYSIBM.SYSTABSTATS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSTABSTATS_HIST rows that were deleted by the MODIFY STATISTICS utility.

csect-name
>   The name of the control section that issued the message.

number-deleted
>   The number of rows that were deleted.

The rows were deleted based on the AGE or DATE criteria that the user specified.

User response: No action is required.

Severity: 0

DSNU1310I   **THE IMAGE COPY FOR** *obj-type dbname.tsname* **DSNUM** *dsnum* **RESIDES ON A TAPE VOLUME BUT IS NOT PREALLOCATED VIA JCL AND TAPEUNITS IS 0**

Explanation: The specification of TAPEUNITS=0 restricts the utility from using tape units that are not preallocated via JCL.

System action: Processing stops for this object.

User response: Either change the value on the TAPEUNITS keyword and resubmit the job, or preallocate the image copy and resubmit the job.

Severity: 8 (error)

DSNU1311I *csect-name* **OBJECT SORTING DETECTED AN ERROR DURING PROCESSING RC =** *return-code*, **RSN =** *reason-code*

Explanation: An internal call to the queueing service failed.

System action: Processing halts with return code 8.

System programmer response: If the problem persists, contact IBM Software Support.

DSNU1312I *csect-name* **PROCESSING DEFERRED FOR OBJECT** *obj-type obj-qualifier.obj-name*, **RC =** *return-code*

Explanation: During parallel processing, the object *obj-qualifier.obj-name* could not be processed.

System action: The system defers processing of the object until parallel processing is complete.

User response: Review the messages that precede DSNU1312I in the job output. Because processing is parallel, messages for other objects that are being recovered might be mixed in with messages for this object. Look for messages that refer to the same *obj-qualifier.obj-name*.

DSNU1313I *csect-name* **PROCESSING DEFERRED FOR OBJECT** *obj-type obj-qualifier.obj-name* **BECAUSE THE NUMBER OF FULL AND INCREMENTAL COPIES EXCEEDS WHAT WAS SPECIFIED FOR TAPEUNITS**

Explanation: During parallel processing, the object *obj-qualifier.obj-name* could not be processed because the number of image copies exceeds the number of tape drives available. The number of tape units available is controlled by the TAPEUNITS parameter of the RECOVER utility.

System action: The system defers processing of the object until parallel processing is complete.

System programmer response: No action is required for this job. For future recovery work, ensure that the value specified for the TAPEUNITS parameter is set to make efficient use of available hardware.

DSNU1314I *csect-name number-deleted* **SYSIBM.SYSKEYTARGETS_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSKEYTARGETS_HIST rows that were deleted by the MODIFY STATISTICS utility. The rows were deleted because AGE or DATE criteria were specified.

csect-name
        The name of the control section that issued the message.

number-deleted
        The number of rows that were deleted.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1315I *csect-name number-deleted* **SYSIBM.SYSKEYTGTDIST_HIST ROWS WERE DELETED**

Explanation: This message displays the number of SYSIBM.SYSKEYTGTDIST_HIST rows that were deleted by the MODIFY STATISTICS utility. The rows were deleted because AGE or DATE criteria were specified.

csect-name
        The name of the control section that issued the message.

number-deleted
        The number of rows that were deleted.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1316I *csect-name* **- THE FOLLOWING TABLESPACES ARE MISSING FROM THE RECOVERY LIST**

Explanation: The RECOVER utility has determined that a table space set (a base table space or a partition and its dependent LOB and XML table spaces) is incomplete.

csect-name
> The name of the control section that issued the message.

The message header is followed by a list of all base, LOB, and XML table spaces that are not in the specified table space set.

System action: The utility job is terminated.

User response: Run the RECOVER utility with the complete list of table spaces, including the dependent LOB and XML table spaces.

Severity: 8 (error)

Related reference:

⮕ RECOVER (DB2 Utilities)

DSNU1317I *csect-name* **- POINT-IN-TIME RECOVERY IS NOT ALLOWED WHEN PENDING DEFINITION CHANGES EXIST ON TABLESPACE** *tablespace-name*

Explanation: Point-in-time recovery was attempted on the specified table space, which has pending definition changes. The pending definition changes must be materialized before point-in-time recovery can run.

tablespace-name
> The name of the table space.

System action: Utility processing is not initiated.

User response: Before running the RECOVER job, take one of the following actions:
- Execute ALTER TABLESPACE with DROP PENDING CHANGES to drop the pending definition changes on the table space.
- Run REORG TABLESPACE with SHRLEVEL REFERENCE or SHRLEVEL CHANGE to materialize the pending definition changes on the table space.

Severity: 8 (error)

Related reference:

⮕ ALTER TABLESPACE (DB2 SQL)

⮕ RECOVER (DB2 Utilities)

⮕ REORG TABLESPACE (DB2 Utilities)

DSNU1318I *csect-name* **- POINT-IN-TIME RECOVERY WITH AN IMAGE COPY THAT WAS TAKEN WITH THE SHRLEVEL CHANGE OPTION IS NOT ALLOWED ON** *object-type object-qualifier.object-name*

Explanation: RECOVER with the TOCOPY, TOLASTCOPY, or TOLASTFULLCOPY option was run. The source for recovery was an image copy with the following characteristics:
- The image copy was taken with the SHRLEVEL CHANGE option.
- The image copy was completed prior to materialization of pending definition changes.

An image copy with this combination of characteristics is not valid as the source for recovery.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

System action: Utility processing is not initiated.

User response: Instead of running RECOVER with an image copy, run RECOVER with the TOLOGPOINT or

TORBA option. Specify a log point that is after the point at which the image copy was completed. To determine the RBA or LRSN value when the image copy was completed:

1. Run REPORT RECOVERY on the table space.

2. Find the PIT LRSN value in the SYSCOPY record for the image copy that was used in the RECOVER job. The PIT LRSN value is the RBA or LRSN when the image copy was completed.

Severity: 8 (error)

Related reference:

➥ REORG TABLESPACE (DB2 Utilities)

DSNU1319I *csect-name* **- POINT-IN-TIME RECOVERY IS NOT ALLOWED BECAUSE TABLE SPACE** *database-name.tablespace-name* **CONTAINS A CLONE TABLE**

Explanation: RECOVER was run to recover a table space with a clone table to a point in time before materialization of pending definition changes. Point-in-time recovery is not valid under these circumstances.

database-name.*tablespace-name*
    The name of the database and table space.

System action: Utility processing is not initiated.

User response: Before running the RECOVER job, execute ALTER TABLE with DROP CLONE to drop the clone table.

Severity: 8 (error)

Related reference:

➥ REORG TABLESPACE (DB2 Utilities)

DSNU1320I *csect-name* **- ALL PARTITIONS FOR** *object-type object-qualifier.object-name* **MUST BE RECOVERED TOGETHER**

Explanation: RECOVER with DSNUM *integer* was issued to recover a single partition to a point in time before materialization of pending definition changes. Recovery to a point in time before materialization of pending definition changes must be performed on all partitions of an object together.

object-type
    The type of object.

object-qualifier.*object-name*
    The name of the object, as a qualified name.

System action: Utility processing is not initiated.

User response: Take one of the following actions to recover all partitions together:

- If table-space-level image copies exist, run RECOVER without the DSNUM parameter.

- If a partition-level image copy exists for each partition, run a single RECOVER control statement for all partitions of the table space. You can do that by running a LISTDEF control statement that contains an INCLUDE list option with the PARTLEVEL parameter, to include all partitions in the table space. Then run RECOVER with that LISTDEF statement.

Severity: 8 (error)

Related reference:

➥ REORG TABLESPACE (DB2 Utilities)

DSNU1321I *csect-name* **OPERAND FOR KEYWORD** *utility-keyword* **IS INVALID. REASON CODE** *reason-code*

Explanation: In a utility control statement, the operand that is specified for the indicated keyword is invalid.

csect-name
    The name of the control section that issued the message.

reason-code
The following numeric value to indicate the reason that the keyword is invalid:

1   The specified LRSN for a data-sharing environment is too low. Six-byte operands must be greater than X'00FFFFFFFFFF'. Ten-byte operands must be greater than X'0000FFFFFFFFFFFFFFFF'.

System action: Utility processing terminates.

User response: Correct the invalid operand and resubmit the utility job.

Severity: 8 (error)

Related reference:

Syntax and options of the RECOVER control statement (DB2 Utilities)

DSNU1350I   *csect-name* **SYSKEYTARGETSTATS CATALOG STATISTICS FOR** *index-name* **KEYSEQ** *keyseq-num* **PARTITION** *part-num* **KEYCARDF =** *cardinality* **HIGHKEY =** *highkey* **HIGH2KEY =** *high2key* **LOWKEY=***lowkey* **LOW2KEY=***low2key*

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the target key part of the partition *part-num* of an index.

cardinality
The number of distinct values of the key-target

highkey   The highest value of the key-target

high2key
The second highest value of the key-target

lowkey   The lowest value of the key-target

low2key   The second lowest value of the key-target

If the values for *highkey*, *high2key*, *low2key*, and *lowkey* are null, then the invoker of RUNSTATS is not authorized to read the table.

The message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSKEYTARGETSTATS table (DB2 SQL)

DSNU1351I   *csect-name* **SYSKEYTARGETS CATALOG STATISTICS FOR** *index-name* **KEYSEQ** *keyseq-num* **KEYCARDF =** *cardinality* **HIGH2KEY =** *high2key* **LOW2KEY=** *low2key*

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the key-target of an index.

cardinality
The number of distinct values of the key-target

high2key
The second highest value of the key-target

low2key   The second lowest value of the key-target

If the values for *high2key* and *low2key* are null, then the invoker of RUNSTATS is not authorized to read the table.

The message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSKEYTARGETS table (DB2 SQL)

DSNU1352I *csect-name* **SYSKEYTGTDISTSTATS CATALOG STATISTICS FOR** *index-name* **KEYSEQ** *keyseq-num*, *keyseq-num*, **...PARTITION** *part-num* **CARDINALITY =** *cardinality* **FREQUENCY KEYVALUE =** *frequency* **X'**hex_string**'** *frequency* **X'**hex_string**'** **... ...** *frequency* **X'**hex_string**'**

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the key column *keyseq-num* partition *part-num*. If more than one *keyseq-num* are listed, then the statistics collected and the expression values reported represent the concatenation of those key columns.

The value of CARDINALITY is the number of distinct values for the listed *keyseq-num*. This statistic appears only if the KEYCARD option was specified.

The values under FREQUENCY are the same as those that are inserted into the DB2 catalog. Consider each frequency as a four-digit number with an implied decimal point two places to the left of the final digit. The resulting number is the percentage of total rows in the table that contain a particular value for the named index key columns.

Frequency is associated with the occurrence of a single value which is listed under KEYVALUE.

Frequencies for the MOST frequent values are collected by default or if the FREQVAL ... MOST options are specified and given a COUNT value greater than zero.

Frequencies for the LEAST frequent values are collected, if the FREQVAL ... LEAST options are specified and given a COUNT value greater than zero.

Frequencies for the MOST and LEAST frequent values are collected, if the FREQVAL ... BOTH options are specified and given a COUNT value greater than zero.

If the default is taken, then the ten most frequently occurring values, plus the lowest and highest values in the partition (if not already included in the ten greatest frequencies), are reported in descending order of frequency.

If fewer than ten values appear, then there are fewer than ten distinct values in the index.

If no values appear, then the values are uniformly distributed.

If frequencies appear but key column values do not, then the invoker of RUNSTATS is not authorized to read the table. Message DSNU619I is issued to identify the table.

The message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSKEYTGTDISTSTATS table (DB2 SQL)

Related information:
"DSNU619I" on page 756

DSNU1353I *csect-name* **SYSKEYTGTDIST CATALOG STATISTICS FOR** *index-name* **KEYSEQ** *keyseq-num*,
keyseq-num,**...PARTITION** *part-num* **CARDINALITY= cardinality FREQUENCY KEYVALUE** *frequency*
X'*hex_string***'** *frequency* **X'***hex_string***'** **... ...** *frequency* **X'***hex_string***'**

Explanation: The RUNSTATS utility issues this message to report the statistics collected for the indexed key column
keyseq-num. If more than one *keyseq-num* are listed, then the statistics collected and the expression values reported
represent the concatenation of those key columns.

The value of CARDINALITY is the number of distinct values for the listed key columns. This statistic appears only if
the KEYCARD option was specified.

The values under FREQUENCY are the same as those that are inserted into the DB2 catalog. Consider each *frequency*
as a four-digit number with an implied decimal point that is two places to the left of the final digit. The resulting
number is the percentage of total rows in the table that contain a particular value for the named index key columns.

Frequency is associated with the occurrence of a single value which is listed under KEYVALUE.

When FREQVAL...MOST and a COUNT value greater than zero are specified, or if frequent values are collected by
default, the frequencies for the most frequently occurring values are collected.

When FREQVAL...LEAST and a COUNT value greater than zero are specified, the frequencies for the least frequently
occurring values are collected.

When FREQVAL...BOTH and a COUNT value greater than zero are specified, the frequencies for the most and least
frequently occurring values are collected.

If the default is taken, then the ten most frequently occurring values are reported in the descending order of
frequency.

If fewer than ten values appear, then there are fewer than ten distinct values in the index.

If no values appear, then the values are uniformly distributed.

If frequencies appear, but key columns do not, then the invoker of RUNSTATS is not authorized to read the table.
Message DSNU619I is issued to identify the table.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

▶ Statistics used for access path selection (DB2 Performance)

▶ RUNSTATS (DB2 Utilities)

▶ SYSIBM.SYSKEYTGTDIST table (DB2 SQL)

DSNU1354I **KEYWORD** *kkkkkkkk* **IS NOT SUPPORTED FOR XML OBJECTS. THE KEYWORD IS IGNORED**

Explanation: The identified keyword *kkkkkkkk* does not apply to XML objects. The use of KEYWORD *kkkkkkkk* is
ignored when specified for XML objects.

System action: The keyword is ignored and the utility processing continues.

Severity: 0 (informational)

DSNU1355I *csect-name* **SYSKEYTGTDISTSTATS CATALOG HISTOGRAM STATISTICS FOR** *index-name*
KEYSEQ *keyseq-num*, *keyseq-num*,**...** **PARTITION** *part-num* **QUANTILE** *quan-num* **LOWVALUE=**
X'*hex_string***'** **HIGHVALUE= X'***hex_string***'** **CARDF =** *cardinality* **FREQUENCYF=** *frequency*

Explanation: The RUNSTATS utility issues this message to report the histogram statistics collected for the indexed
key column for each quantile, as indicated by *quan-num* and each partition, as indicated by *part-num*. If more than
one key column is listed, then the statistics collected represent the concatenation of those key columns.

When a value of 1 is specified for NUMCOLS or for default histogram processing, LOWVALUE and HIGHVALUE

represent the first key expression value low bound and high bound for each quantile range. If a value greater than 1 is specified for NUMCOLS, then the LOWVALUE and HIGHVALUE represent the concatenation of the key expressions.

cardinality represents the number of distinctive values in each quantile. If more than one key column is listed, then *cardinality* represents the number of distinctive combinations of that KEYSEQ group in each quantile.

frequency represents the percentage of rows in each quantile.

If the total number of distinctive values for that KEYSEQ, or KEYSEQ group, is less than the specified number of quantiles, then the quantiles collected are up to the number of distinctive values.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSKEYTGTDISTSTATS table (DB2 SQL)

DSNU1356I   *csect-name* **SYSKEYTGTDIST CATALOG HISTOGRAM STATISTICS FOR** *index-name* **KEYSEQ** *keyseq-num*, *keyseq-num*,...**QUANTILE** *quan-num* **LOWVALUE=X'***hex_string*' **HIGHVALUE= X'***hex_string*' **CARDF =** *cardinality* **FREQUENCYF=** *frequency*

Explanation: The RUNSTATS utility issues this message to report the histogram statistics collected for the indexed key column for each quantile, as indicated by *quan-num*. If more than one key column is listed, then the statistics collected represent the concatenation of those key columns.

For single indexed KEYSEQ, LOWVALUE, and HIGHVALUE columns, each column represents the key column value low bound and high bound for each quantile range. If more than one key column is listed, then the LOWVALUE and HIGHVALUE represent the concatenation of those KEYSEQ values.

cardinality represents the number of distinctive values in each quantile. If more than one key column is listed, then *cardinality* represents the number of distinctive combinations of that KEYSEQ group in each quantile.

frequency represents the percentage of rows in each quantile.

If the total number of distinctive values for that KEYSEQ is less than the specified number of quantiles, then the quantiles collected are up to the number of distinctive values.

This message is routed to SYSPRINT.

System action: Processing continues.

User response: No action is required.

Severity: 0 (informational)

Related reference:

Statistics used for access path selection (DB2 Performance)

RUNSTATS (DB2 Utilities)

SYSIBM.SYSKEYTGTDIST table (DB2 SQL)

DSNU1357I   **THE STATS PROFILE FOR TABLE table-name HAS BEEN SET IN SYSTABLES_PROFILES.**

Explanation: The statistics profile for the table has been set in the SYSTABLES_PROFILE catalog table.

table-name
      The name of the table.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1358I   THE STATS PROFILE FOR TABLE *table-name* **HAS BEEN UPDATED IN SYSTABLES_PROFILES.**

Explanation: The profile for the table has been updated in the SYSTABLES_PROFILE catalog table.

table-name
>   The name of the table.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1360I   THE STATS PROFILE FOR TABLE *table-name* **HAS BEEN DELETED IN SYSTABLES_PROFILES.**

Explanation: The profile for the table has been deleted from the SYSTABLES_PROFILE catalog table.

table-name
>   The name of the table.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1361I   THE STATS PROFILE WITH STATSTIME = *timestamp* **FOR TABLE** *table-name* **HAS BEEN USED.**

Explanation: The profile that is identified by the indicated STATSTIME timestamp has been used for the table.

timestamp
>   The STATSTIME timestamp.

table-name
>   The name of the table.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1362I   THE OBJECT *schema-name*.*name* **IN THE STATS PROFILE DOES NOT EXIST.**

Explanation: RUNSTATS was executed with USE PROFILE, but the schema does not exist in the profile. The object might have been dropped.

schema-name.*name*
>   The name of the object that does not exist.

System action: Execution of the job step stops.

User response: You can reset the profile in SYSTABLES_PROFILE for the available objects.

If the statistics profile has references to indexes that have been deleted, replace the profile:

1. Execute RUNSTATS with DELETE PROFILE to delete the invalid profile.
2. Execute RUNSTATS with SET PROFILE set a new valid profile in the SYSTABLES_PROFILES catalog table.

Severity: 8 (error)

DSNU1363I   **THE STATS PROFILE FOR TABLE** *table-name* **NOT FOUND.**

Explanation: RUNSTATS was executed with the option to use, update, or delete a profile, but no profile was found for the table.

The statistics profile for the specified table might not exist because the table was dropped.

table-name
>   The name of the table.

System action: Execution of the job step stops.

User response: Verify that the table exists. If it has been dropped, remove it from the RUNSTATS utility statement.

Execute RUNSTATS TABLESPACE TABLE SET PROFILE to set the profile for the table indicated by *table-name*.

If the table was dropped, remove the table specification for the dropped table from the RUNSTATS utility control statement.

Severity: 8 (error)

Related concepts:

➡ RUNSTATS profiles (DB2 Utilities)

Related reference:

➡ RUNSTATS (DB2 Utilities)

DSNU1365I   **THE STATS PROFILE FOR TABLE** *table-name* **IS INVALID.**

Explanation: RUNSTATS TABLESPACE TABLE *table-name* USE PROFILE executed, but the profile specified for the table is invalid. No statistics were collected for the table.

table-name
>   The name of the table.

System action: Execution of the job step stops.

User response: Execute RUNSTATS TABLESPACE TABLE SET PROFILE to set the profile for the table indicated by *table-name*.

Severity: 8 (error)

DSNU1368I   **PARSING STATS PROFILE FOR TABLE** *table-name*

Explanation: The parsing of STATS PROFILE for the indicated table has started.

table-name
>   The name of the table.

This message is issued during the execution of RUNSTATS TABLESPACE TABLE USE PROFILE.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1369I   **PARSING STATS FOR TABLE** *table-name* **COMPLETED.**

Explanation: The parsing of STATS PROFILE for the indicated table is complete.

table-name
>   The name of the table.

This message is issued during the execution of RUNSTATS TABLESPACE TABLE USE PROFILE.

System action: Execution of the job step continues.

User response: No action is required.

DSNU1370I • DSNU1374I

Severity: 0

DSNU1370I   NO EXISTING STATISTICS AVAILABLE FOR TABLE *table-name***.**

Explanation: No statistics were found in the catalog tables for the specified table.

table-name
     The name of the table.

System action: Execution of the utility job set terminates.

User response: No action is required.

Severity: 8 (error)

DSNU1371I   INDEX KEYWORD CANNOT BE SPECIFIED WITH USE PROFILE.

Explanation: The INDEX keyword cannot be used with RUNSTATS TABLESPACE TABLE USE PROFILE.

System action: Execution of the job step stops.

User response: Remove either the INDEX keyword or the USE PROFILE keywords and resubmit the RUNSTATS job.

Severity: 8 (error)

DSNU1373I   PAGE SAMPLING NOT DONE FOR TABLE SPACE *space-name***.**

Explanation: The TABLESAMPLE keyword cannot be used with RUNSTATS TABLESPACE TABLE if the table space contains multiple tables.

space-name
     The name of the table space.

When RUNSTATS TABLESPACE TABLE is executed on a multi-table table space or a table space that is segmented and not partitioned, the TABLESAMPLE keyword is not used. Instead, all pages are scanned to collect statistics.

System action: Execution of the job step continues.

User response: No action is required.

Severity: 0

DSNU1374I   SAMPLE SIZE USED : ROWS = *rows***, PAGES=***pages***, SAMPLE RATE USED =** *sample-rate***,**
            REPEATABLE = *seed-value***.**

Explanation: This message provides information about the sample used by the RUNSTATS utility in the execution of RUNSTATS TABLESPACE TABLE TABLESAMPLE SYSTEM.

rows   The number of table rows used in the sample.

pages   The number of pages used in the sample.

sample-rate
     The sampling rate, which is a percentage of pages used compared to the number of pages in the table.

seed-value
     An integer value that is used to determine which pages to include in the sample.

System action: Execution of the job continues.

User response: No action is required.

Severity: 0

DSNU1375I   NO ROWS SAMPLED. SAMPLING RATE TOO SMALL OR TABLE SPACE IS EMPTY.

Explanation: Either the sampling rate specified for RUNSTATS TABLESPACE TABLE TABLESAMPLE SYSTEM was too small or the table space or partition was mass-deleted.

System action: Execution of the job step stops.

User response: Analyze the RUNSTATS report for problem determination. Update the RUNSTATS utility job statement to use a larger sampling rate and resubmit the job.

Severity: 8 (error)

DSNU1376I   SYSTABLES_PROFILES VALUE FOR TABLE *creator*.*table* **IS** *profile-value*

Explanation: Preview processing has retrieved the profile from SYSTABLES_PROFILES for the indicated table. The profile is output to SYSPRINT.

creator.*table*
        The table for which the preview was generated.

profile-value
        The profile that was retrieved.

System action: Processing continues.

User response: Examine the preview output and correct any errors. To correct the profile for a table:
1. Execute RUNSTATS with DELETE PROFILE to delete the profile
2. Execute RUNSTATS with SET PROFILE and specify the corrected options to set the profile.

Severity: 0

DSNU1378I   INDEX NAME *index-name* **IN PROFILE DOES NOT SPECIFY THE CREATOR NAME**

Explanation: The profile text from SYSTABLES_PROFILES contains an index name without the creator name. All index names in profile text must be fully qualified with the creator name.

index-name
        The name of the index.

PROFILE functions cannot be executed until syntax errors in the profile have been corrected.

System action: Utility processing is terminated.

User response: Either add the creator name for the index or delete the profile.

To add the creator name, use either:
• RUNSTATS UPDATE PROFILE
• SQL UPDATE

To delete the profile, use either:
• RUNSTATS DELETE PROFILE
• SQL DELETE

Severity: 8 (error)

Related reference:

⇥ RUNSTATS (DB2 Utilities)

⇥ DELETE (DB2 SQL)

⇥ UPDATE (DB2 SQL)

DSNU1379I   INVALID KEYWORD *keyword* **IN PROFILE**

Explanation: A profile contains a keyword that is not allowed.

keyword   The invalid keyword.

System action: The executing utility job step is terminated. Message DSNU1368I is issued.

User response: Remove all invalid keywords from the profile and rerun the utility. Message DSNU1368I identifies the profile that contains the error.

Severity: 8 (error)

Related information:

"DSNU1368I" on page 887

DSNU1380I   *csect-name table-name* **CATALOG ACCESSPATH STATISTICS RESET FOR** *object-name* **SUCCESSFUL**

Explanation: The access path statistics for the specified object were reset successfully in the specified catalog table.

csect-name
    The name of the control section that issued the message.

table-name
    The name of the catalog table that contained the statistics that were reset.

object-name
    The name of the object for which access path statistics were reset.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related tasks:

➡ Resetting access path statistics (DB2 Utilities)

➡ Maintaining statistics in the catalog (DB2 Performance)

Related reference:

➡ DB2 catalog tables (DB2 SQL)

➡ Statistics used for access path selection (DB2 Performance)

DSNU1381I   *csect-name* **MATCHING SYSSTATFEEDBACK ROWS MIGHT NOT BE DELETED, REASON:**
    reason-code

Explanation: The RUNSTATS utility removes SYSIBM.SYSSTATFEEDBACK table rows that match the statistics that it collects. Some applicable rows might not have been deleted from the SYSSTATFEEDBACK table for the specified reason. The most likely cause is contention on the SYSSTATFEEDBACK table.

reason-code
    A numeric value that indicates the reason for the failure.

System action: Processing continues. Some rows might remain in SYSIBM.SYSSTATFEEDBACK for statistics that were captured by this execution of the RUNSTATS utility.

User response: You can leave the rows not deleted by RUNSTATS in SYSIBM.SYSSTATFEEDBACK. These rows are deleted by the next successful execution of RUNSTATS that collects those statistics. Rows can also be manually deleted from SYSIBM.SYSSTATFEEDBACK.

If this message occurs repeatedly, see the specified *reason-code* to determine the cause. If indicated by the *reason-code*, check for contention on catalog table SYSIBM.SYSSTATFEEDBACK.

DSNU1382I   THE STATS PROFILE FOR TABLE *table-name* **DOES NOT EXIST. DEFAULT PROFILE STATS COLLECTED.**

Explanation: The RUNSTATS utility was run with the USE PROFILE option, but no profile exists for the indicated table. Default statistics are been collected for the table.

table-name
       The name of the table.

System action: Processing continues.

User response: No action is required if no additional statistics are required. If more statistics are needed, create a RUNSTATS profile for the table that is indicated by *table-name*, and run the RUNSTATS with the USE PROFILE option again.

Severity: 4 (warning)

Related concepts:

➥ RUNSTATS profiles (DB2 Utilities)

Related tasks:

➥ Creating RUNSTATS profiles (DB2 Utilities)

➥ Using RUNSTATS profiles (DB2 Utilities)

DSNU1400I   *csect-name* **- COPYTOCOPY UTILITY NOT SUPPORTED FOR** *object-type object-qualifier.object-name* **DSNUM** *n*

Explanation: The COPYTOCOPY utility cannot make copies of any of the following objects:

- Table spaces DSNDB01.SYSUTILX, DSNDB01.DBD01, DSNDB06.SYSCOPY, or DSNDB01.SYSDBDXA, or any of their indexes.
- An image copy that was taken by using the COPY utility with the CONCURRENT option.

csect-name
       The name of the control section that issued the message.

object-type
       The type of object.

object-qualifier.object-name
       The qualified object name.

n   The number of the partition or data set.

System action: Processing stops for this object. If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, COPYTOCOPY processes the next item in the list.

User response: If the unsupported object is a catalog or directory object, use another method to make all of the image copies for this object.

If the unsupported object is an image copy that was made with the CONCURRENT option, make a primary local image copy by running the COPY utility without the CONCURRENT option. Then, rerun COPYTOCOPY to make more image copies.

Severity: 8 (error)

Related concepts:

➥ Multiple image copies (DB2 Utilities)

➥ Page set and data set copies (DB2 Administration Guide)

Related reference:

➥ COPYTOCOPY (DB2 Utilities)

➥ COPY (DB2 Utilities)

DSNU1401I *csect-name* **- UNABLE TO LOCATE IMAGE COPY FOR** *obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: One of the following conditions prompted this error code:

- The COPYTOCOPY utility attempted to make additional image copies from the referenced object. However, SYSIBM.SYSCOPY does not contain a record of the referenced object.

- UNLOAD utility could not find a record in SYSIBM.SYSCOPY for the image copy data set that was requested.

System action: Processing halts for this object.

User response: Make a primary local image copy by running the COPY utility, then run COPYTOCOPY to make additional image copies.

Severity: 8 (error)

DSNU1402I *csect-name* **-** *copy-type* **FOR** *obj-type obj-qual.obj-name* **DSNUM** *n* **WITH START_RBA** *rba* **ALREADY EXISTS**

Explanation: COPYTOCOPY attempted to make an image copy that already exists.

System action: The utility does not make the image copy for the specified object type. COPYTOCOPY makes the remaining types of image copies. If there is a list of objects to be copied, COPYTOCOPY processes the next object in the list.

User response: Run the REPORT utility to determine the image copy types that you need. Modify your COPYTOCOPY job, and rerun it as needed.

Severity: 4 (warning) or 8 (error)

Related concepts:

➥ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU1403I *csect-name* **-** *copy-type* **DATA SET** *dsn* **WITH START_RBA** *rba* **IS IN USE BY COPYTOCOPY FOR** *obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: The COPYTOCOPY utility issues this message before it opens an image copy data set and begins the copying process.

System action: COPYTOCOPY processing continues.

Severity: 0 (informational)

Related concepts:

➥ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU1404I *csect-name* **- COPYTOCOPY PROCESSING COMPLETED FOR** *obj-type obj-qual.obj-name* **DSNUM** *n* **ELAPSED TIME =** *elapsed time* **NUMBER OF PAGES COPIED =** *n*

Explanation: COPYTOCOPY issues this message at the normal completion of an object. The time that elapsed since the beginning of processing the object to its completion was *hh* hours, *mm* minutes, and *ss* seconds.

System action: The COPYTOCOPY utility completed normally for this object. Processing continues.

Severity: 0 (informational)

DSNU1405I *csect-name* **-** *site-type* **SITE PRIMARY IMAGE COPY DOES NOT EXIST FOR** *obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: The COPYTOCOPY utility attempted to make a backup image copy for the site, but the primary image copy does not exist.

System action: The utility cannot make the site backup copy. If the utility job includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, COPYTOCOPY processes the next item on the list.

User response: Modify your COPYTOCOPY job, COPYDDN or RECOVERYDDN option, to include the ddname for the primary copy.

Severity: 8 (error)

Related concepts:

➦ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU1406I *csect-name* **- COPYTOCOPY COMPLETED. ELAPSED TIME =** *elapsed_time*

Explanation: DB2 issues this message at the normal completion of the COPYTOCOPY utility. The time that elapsed since the beginning of the COPYTOCOPY utility to its completion was *hh* hours, *mm* minutes, and *ss* seconds.

System action: Processing continues to the next utility statement in the utility job step, if any. If not, the utility job step halts, and the UTILID of the utility job step becomes available for reuse.

Severity: 0 (informational)

DSNU1407I *csect-name* **- MISMATCH BETWEEN THE NUMBER OF PAGES READ=**count **AND THE NUMBER OF PAGES ORIGINALLY COPIED=**count **FOR INPUT DATA SET** *dsname* **OBJECT NAME** *obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: The COPYTOCOPY or RECOVER utility issues this message when it detects that the number of pages read from the input image copy does not match the original number of pages copied by the COPY utility. The utility retrieves the number of pages originally copied from the COPYPAGESF column of the SYSIBM.SYSCOPY record for the input image copy data set.

System action: The COPYTOCOPY utility terminates processing of this object. The RECOVER utility attempts to continue by falling back to the previous full image copy, if any. If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, COPYTOCOPY processes the next item in the list.

User response: Report to IBM Center. Provide output of COPYTOCOPY job, the data set referenced in the message, and run REPORT utility on the object and provide its output.

Severity: 8 (error)

DSNU1408I *csect-name* **- SOURCE DATA SET** *dsname* **DOES NOT MATCH** *obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: The COPYTOCOPY or RECOVER utility issues this message when the DBID or PSID of the source data set does not match the DBID or PSID of the table space or index space for which the copy is made. This error can occur if the data set was reused.

The UNLOAD utility displays this message when no records were UNLOADED from an image copy data set because the OBID for the tables did not match the information in the DBD. This can occur if the tables in the table space were dropped and re-created after taking the image copy.

The COPY utility issues this message when the DBID or PSID of the header page of the target object does not match the information in the DBD. In this case, the source data set dsname will be omitted from the message.

System action: RECOVER attempts to continue processing by falling back to the previous full image copy, if any.

COPYTOCOPY searches for the next available image copy data set with the same START_RBA value as the START_RBA value registered in SYSIBM.SYSCOPY for dsname. If COPYTOCOPY finds a data set, it uses that data set. Otherwise, COPYTOCOPY issues message DSNU1401I.

The UNLOAD utility completes with reason-code four.

If FROMCOPY is specified, processing of this object halts. COPY fails with RC8 and an image copy is not created.

User response: Determine why the data set was reused for the image copy data set that is registered in SYSIBM.SYSCOPY.

If FROMCOPY is specified, correct the COPYTOCOPY utility control statement. Specify a source data set that is an image copy of the table space or index space for which you want additional image copies.

If this message was received for COPY, try to determine how the header page of the object became corrupted, and fix it using the REPAIR utility, if possible. Otherwise, recover the object using a good copy.

If this message was received for UNLOAD, specify a source data set that is an image copy of the table space taken with the current table definitions.

DSNU1409I • DSNU1420I

Severity: 4 (warning) or 8 (error)

DSNU1409I *csect-name* **-** *ttttt* **ERRORS ON DATA SET NAME=***dsname obj-type obj-qual.obj-name* **DSNUM** *n*

Explanation: DB2 detected I/O errors on the named object. *ttttt* is either 'READ' or 'WRITE'.

System action: If FROMCOPY is specified, processing is terminated. Otherwise, processing will continue. The utility will attempt to use the next available image copy data set that matches COPYTOCOPY job specification with the same START_ RBA as the data set named in error. If it fails to find one, processing of this object is terminated.

User response: An IOS (I/O subsystem) message might be included on the MVS console SYSLOG indicating the type of I/O error that occurred. If an IOS message was issued, follow the Problem Determination directions for the message.

Severity: 4 (warning)

DSNU1410I *csect-name* **- INCREMENTAL IMAGE COPY DISALLOWED FOR** *obj-type obj-qual.obj-name* **DSNUM** *n*. **FULL IMAGE COPY WILL BE USED**

Explanation: An incremental image copy is disallowed for the specified object type. The utility ignores the request, and creates a full image copy if available.

System action: The system treats the request to copy an incremental image copy as a request to copy a full image copy.

User response: The capacity of the target image copy data set may not be sufficient if its size was established assuming an incremental image copy. If this condition occurs, halt the utility and resubmit the job step with the appropriate size.

Severity: 4 (warning)

DSNU1411I *csect-name* **-** *ut_name* **DOES NOT SUPPORT RESTART(PHASE). PHASE CHANGED TO CURRENT.**

Explanation: This utility does not support the RESTART(PHASE).

System action: The utility overrides RESTART(PHASE) with RESTART(CURRENT).

Severity: 0 (informational)

DSNU1420I *csect-name* **SET** *status* **OPERATION SUCCESSFUL**

Explanation: This message reports the successful completion of the REPAIR SET TABLESPACE utility operation.

csect-name
>   The name of the control section that issued the message.

status **PRO**
>   REPAIR SET TABLESPACE PRO was specified, and the Persistent Read Only (PRO) restricted status has been turned on for the specified table space partition.

status **NOPRO**
>   REPAIR SET TABLESPACE NOPRO was specified, and the Persistent Read Only (PRO) restricted status has been turned off for the specified table space partition.

Note: Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

System action: The REPAIR utility completed normally.

User response: No action is required.

Severity: 0

DSNU1421I *csect-name* **TABLESPACE** *object-type object-qualifier.object-name* **PARTITION** *part* **IS IN PRO RESTRICTED STATE**

Explanation: This message is issued when:

- The REPAIR utility set the Persistent Read Only (PRO) restricted status.
- Read-only operations by a utility occurred on a partition in PRO restricted status.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

part　The partition number for a partitioned table space.

System action: Utility processing on the object completed normally.

User response: No action is required. Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

Severity: 4 (warning)

DSNU1422I *csect-name* **DELETE CRITERIA IS IGNORED AND RETAIN LAST(2) IS IN EFFECT DUE TO PRO RESTRICTED STATE ON** *object-type object-qualifier.object-name* **PARTITION** *part*

Explanation: The MODIFY RECOVERY utility was executed on a table space partition that is in Persistent Read Only (PRO) restricted status. The two most recent full image copies of the partition will be retained along with the SYSIBM.SYSLGRNX records for the partition.

This condition occurs from using the REPAIR utility to set the PRO restricted status for the purpose of prohibiting updates.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

part　The partition number for a partitioned table space.

System action: Utility processing on the object is terminated.

User response: No action is required. Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

Severity: 4 (warning)

DSNU1423I *csect-name utility-name* **UTILITY WILL NOT PROCESS**object-type *object-qualifier.object-name* **PARTITION** *part* **IN PRO RESTRICTED STATE**

Explanation: The COPY utility was executed on a table space partition that is in Persistent Read Only (PRO) restricted status. An image copy will not be created for this partition due to the PRO restricted status.

This condition occurs from using the REPAIR utility to set the PRO restricted status for the purpose of prohibiting updates.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility.

object-type
>   The type of object.

object-qualifier.object-name
>   The name of the object, as a qualified name.

part   The partition number for a partitioned table space.

System action: The table space is not copied.

User response: No action is required. Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

Severity: 0

DSNU1424I   *csect-name utility-name* **UTILITY MUST BE EXECUTED AT THE PARTITION LEVEL ON** *object-type object-qualifier.object-name* **DUE TO PRO RESTRICTED STATE**

Explanation: An attempt was made to execute the COPY utility or MODIFY RECOVERY utility at the table space level (DSNUM ALL option) on a table space with one or more partitions in Persistent Read Only (PRO) restricted status.

This condition occurs from using the REPAIR utility to set the PRO restricted status for the purpose of prohibiting updates.

csect-name
>   The name of the control section that issued the message.

utility-name
>   The name of the utility.

object-type
>   The type of object.

object-qualifier.object-name
>   The name of the object, as a qualified name.

System action: Utility processing on the object is terminated.

User response: Execute the COPY utility or MODIFY RECOVERY utility at the partition level by using the DSNUM *n* option or by using a LISTDEF list with the PARTLEVEL option.

Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

Severity: 8 (error)

DSNU1425I   *csect-name* **PRO RESTRICTED STATE ON** *object-type object-qualifier.object-name* **PROHIBITS PROCESSING**

Explanation: An attempt was made to execute a utility on a table space partition that is in Persistent Read Only (PRO) restricted status, and updates are not allowed.

This condition occurs from using the REPAIR utility to set the PRO restricted status for the purpose of prohibiting updates.

csect-name
>   The name of the control section that issued the message.

object-type
>   The type of object.

object-qualifier.object-name
>   The name of the object, as a qualified name.

System action: Utility processing on the object is terminated.

User response: Check with the DB2 system administrator or database administrator. Do not remove the PRO restricted status without the consent of the DB2 system administrator or database administrator, because updates to the partition might cause data loss.

Severity: 8 (error)

DSNU1450I *csect-name* **- CHARACTER CONVERSION FROM CCSID** *from-ccsid* **TO CCSID** *to-ccsid* **FAILED WITH ERROR CODE 'error-code' FOR TABLE** *dbid.obid* **COLUMN 'column-number'.**

Explanation: A conversion error occurred during the conversion of a string to a different coded character set. The error occurred during the processing of the utility statement.

The *from-ccsid* identifies the coded character set of to be converted. and the *to-ccsid* identifies the coded character set to which it must be converted.

The 'error-code'indicates the type of error as defined as follows: 8 - Length exception (for example, expansion required for PC MIXED data exceeds the maximum length of the string). 12 - Invalid code point 16 - Form exception (for example, invalid MIXED data). 20 - Translate procedure error (for example, an error during Unicode/390 stage 2 conversions or an exit set the length control field of the string to an invalid value). 24 - the conversion services have not been setup correctly or the conversion is not supported.

The *dbid.obid*:dd.gives the database id and the table obid of the object with the conversion error. This could be either the target or the source of the conversion.

The 'column-number' gives the column number of the failing object. This could be either the target or the source of the conversion.

System action: The utility job is terminated.

User response: Refer to Part 2 of DB2 Diagnosis Guide and Reference for information on identifying and reporting the problem. The The user may generate a dump by adding the DIAGNOSE ABEND statement in the utility job and re-running the utility. If the error can be fixed, correct the error and resubmit the utility, else issue the TERM utility command to terminate the utility.

Severity: 8 (error)

DSNU1455I *csect-name obj-type obj-qual.obj-name* **NOT CLONED**

Explanation: You specified an object-name along with the CLONE keyword, but the object is not involved with cloning. This message will also be issued if you specified an invalid clone object-name.

System action: Utility processing is not initiated.

User response: Correct the object-name to specify a clone object and resubmit the job.

Severity: 8 (error)

DSNU1457I *csect-name util-name* **UTILITY MAY NOT BE RUN AGAINST CLONE OBJECT** *qualifier.object-name*

Explanation: An attempt was made to execute utility *util-name* against a clone object *qualifier.object-name*. This is not allowed. The named object cannot be processed by the named DB2 utility.

System action: The utility job is terminated.

User response: Remove the invalid utility control statements, or refer to the base object, and resubmit the job.

Severity: 8 (error)

DSNU1458I *csect-name* **KEYWORD OR OPERAND** *oooooo* **INVALID WITH CLONE OBJECT** *qualifier.object-name* **FOR THE** *uuuuuu* **UTILITY**

Explanation: The use of KEYWORD or OPERAND *oooooo* of the *uuuuuu* utility is invalid when specified for a clone object. The object you specified cannot be processed by the named DB2 utility.

System action: The utility job is terminated.

User response: Remove the invalid utility control statements, or refer to a base object, and resubmit the job.

Severity: 8 (error)

DSNU1459I *csect-name* **KEYWORD OR OPERAND** *oooooo* **INVALID WITH AN OBJECT INVOLVED IN CLONING** *qualifier.object-name* **FOR THE** *uuuuuu* **UTILITY**

Explanation: The use of KEYWORD or OPERAND *oooooo* of the *uuuuuu* utility is invalid when specified for an object involved in cloning.

System action: The utility job is terminated.

User response: Remove the invalid utility control statements and resubmit the job.

Severity: 8 (error)

DSNU1502I *csect-name* **- RECOVER OF** *obj-type obj-qual.objname dsnum n* **WILL PROCEED TO LOG POINT X'***rba/lrsn***'**

Explanation: DB2 issues this message if the RECOVER utility statement includes the TOLASTCOPY or TOLASTFULLCOPY keyword. The message provides information on the point in time that the object was recovered to. In a data sharing environment, the log point identified is an LRSN. Otherwise, the log point identified is an RBA.

System action: The RECOVER utility statement continues processing.

Severity: 0 (informational)

DSNU1504I *csect-name* **RECOVERY OF NOT LOGGED** *obj-type obj-qual.obj-name* **CANNOT PROCEED BECAUSE THE TOLOGPOINT OR TOLOGRBA SPECIFIED IS NOT A RECOVERABLE POINT**

Explanation: The object has the NOT LOGGED logging attribute. For not logged objects, the recovery point must match a recoverable point. A recoverable point is established when one of the following circumstances are true:

- When a table space is altered from LOGGED to NOT LOGGED, which alters all of the index spaces for indexes on all of the tables in the table space
- When an image copy is taken against a not logged table space or a not logged index space

System action: Utility processing halts for this object. The utility job completes with a return code of 8. If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Resubmit the RECOVER job with TOLASTCOPY or TOLOGPOINT that matches a recoverable point.

Severity: 8 (error)

DSNU1505I *csect-name* **RECOVERY OF THE NOT LOGGED** *obj-type obj-qual.obj-name* **WAS TO THE LAST RECOVERABLE POINT: RBA/LRSN** *rba/lrsn*. **THE OBJECT HAS BEEN CHANGED SINCE THAT POINT**

Explanation: A full recovery was requested, but because the object is not a logged object, it was recovered to its last recoverable point, which means that any changes made to the object after the recoverable point are lost. This message warns that the object has been modified since the last recoverable point.

A recoverable point is established when one of the following circumstances is true:

- When a table space is altered from LOGGED to NOT LOGGED, which alters all of the index spaces for indexes on all of the tables in the table space
- When an image copy is taken against a not logged table space or a not logged index space

System action: Processing continues. DB2 cannot recover the modifications that were made to the object since the last recoverable point.

Severity: 4 (warning)

DSNU1506I *csect-name***: RECOVER** *recover-type* **OF** *object-type object-qualifier***.***object-name* **IS NOT ALLOWED BECAUSE THE LOGGING ATTRIBUTE OF THE OBJECT IS NOT LOGGED**

Explanation: The object to be recovered has a NOT LOGGED logging attribute, so no log records are available for recovery.

csect-name
> The name of the control section that issued the message.

recover-type
> The type of recovery: LOGONLY or BACKOUT.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

System action: Utility processing halts for this object. The utility job completes with a return code of 8.

If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Recover the object to a previous point in time that is a recoverable point. A recoverable point is:

- When a table space is altered from LOGGED to NOT LOGGED, thus altering all indexes on all tables in the table space.
- When an image copy is taken against a not logged table space or a not logged index.

Run the REPORT utility with the RECOVERY option and analyze the output to determine which recoverable point you want to use. Recover the not logged object to the recoverable point by using the TOLOGPOINT, TORBA,TOCOPY, TOLASTCOPY, or TOLASTFULLCOPY recovery options.

Severity: 8 (error)

Related reference:

➡ REPORT (DB2 Utilities)

➡ RECOVER (DB2 Utilities)

DSNU1507I   *csect-name*: **THE IMAGE COPY FOR** *obj-type obj-qual.obj-name* **IS NOT USABLE. RECOVERY BY FALLBACK TO THE PREVIOUS IMAGE COPY IS NOT POSSIBLE BECAUSE REQUIRED LOG RECORDS WERE NOT WRITTEN**

Explanation: DB2 cannot use the identified image copy. DB2 cannot recover by falling back to the previous image copy and applying log records. Some required log records for *object-qualifier.object-name* were not written because *object-qualifier.object-name* had the NOT LOGGED attribute.

System action: Utility processing halts for this object. The utility job completes with a return code of 8.

If the utility job step includes an OPTIONS control statement with the EVENT(ITEMERROR,SKIP) option, the utility will process the next item in the list.

User response: Check the utility job output for prior messages indicating an allocation, open, or read error on the image copy data set identified by the DSNU515I message. Correct the situation if possible. If the problem cannot be corrected, then run the REPORT RECOVERY utility for the object to determine the options that are available for recovery.

Severity: 8 (error)

DSNU1508I   *csect-name* **RECOVERY OF** *obj-type obj-qual.obj-name* **IS NOT ALLOWED TO A NOT LOGGED POINT**

Explanation: The object has the LOGGED attribute specified, but the TORBA, TOLOGPOINT, TOCOPY, TOLASTCOPY, TOLASTFULLCOPY option identifies a point when the object had the NOT LOGGED attribute. This is not allowed because the current logging attribute of the object must match the logging attribute of the object at the time of the chose recovery point.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) has been specified for the job step, then processing continues for the other objects in the list. Otherwise, recovery processing terminates.

User response: Alter the logging attribute of the object to NOT LOGGED, then resubmit the RECOVER utility job.

Severity: 8 (error)

DSNU1510I   *csect-name* **- LOG APPLY PHASE COMPLETE, ELAPSED TIME:** *HH.MM.SS*

Explanation: This message is issued at the normal completion of the LOG APPLY phase of the RECOVER utility. The time that has elapsed between the beginning and completion of this phase was:

hh   hours

mm   minutes

ss   seconds

System action: Processing continues.

Severity: 0 (informational)

DSNU1511I   *csect-name* **- FAST LOG APPLY WAS NOT USED FOR** *utility-name*

Explanation: FAST LOG APPLY was not used during utility processing.

csect-name
        The name of the control section that issued the message.

utility-name
        The name of the utility: RECOVER or FLASHCOPY WITH CONSISTENCY.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNU1512I   *csect-name* **RECOVERY OF A SINGLE PIECE OF** *obj-type obj-qual.obj-name* **IS NOT ALLOWED TO A DIFFERENT ROW FORMAT**

Explanation: You attempted to recover a single piece of a multi-piece linear table space to a point in time in which the rows are in a different format than the current format. For example, this error can occur when a table space is in reordered row format and you attempted to recover a piece of the table space to a point in time when the table space was in basic row format.

System action: The RECOVER utility terminates.

Programmer response: To recover the data, you can do one of the following actions:

• Recover the entire table space rather than a single piece.

• Recover the piece to a point-in-time when the table space is in the same row format.

Severity: 8 (error)

DSNU1513I   *csect-name* **RECOVER OF** *obj-type obj-qual.obj-name* **WITH A VALIDPROC IS NOT ALLOWED TO A DIFFERENT ROW FORMAT**

Explanation: You cannot recover an object that requires the recovery of part or all of a table with varying-length columns and a VALIDPROC when the RECOVER is one of the following types:

• From a point in time when the table space was in reordered row format to a point in time when the table space was in basic row format

• From a point in time when the table space was in basic row format to a point in time when the table space was in reordered row format.

System action: The RECOVER utility terminates.

Programmer response: To recover the object, you can execute the following steps:

1. ALTER the VALIDPROC to null on all tables that are part of the object to be recovered.

2. RECOVER the object.

3. Make any necessary changes to the VALIDPROCs so that the VALIDPROCs are in sync with the current format of the row.

4. ALTER ADD the VALIDPROCs to the tables.

Severity: 8 (error)

DSNU1519I *csect-name* **INVALID SPECIFICATION OF RESTOREBEFORE X'***xxxxxxxxxxxx***'**

Explanation: This message is issued for one of the following reasons:

- The LRSN or RBA value that was specified for the RESTOREBEFORE parameter is greater than the LRSN or RBA value that was specified for the TOLOGPOINT or TORBA parameter. The RESTOREBEFORE value determines the ending point of the search for image copies to restore. Therefore, the RESTOREBEFORE value must be less than the LRSN or RBA value that was specified for the TOLOGPOINT option. The RESTOREBEFORE value must also be less than the RBA value that was specified for the TORBA option.
- RECOVER did not find an image copy, concurrent copy, or system-level backup with a START RBA, START LRSN, or DATA COMPLETE LRSN value that is lower than the value that was specified for the RESTOREBEFORE parameter.

X'*xxxxxxxxxxxx***'**
    The LRSN or RBA value that was specified for the RESTOREBEFORE parameter.

System action: The RECOVER utility terminates processing.

User response: Take one of the following actions:

- Examine the RECOVER control statement. Modify the RESTOREBEFORE parameter value to be less than the TOLOGPOINT or TORBA parameter value. Resubmit the job.
- Run REPORT RECOVERY. In the output, examine image copy, concurrent copy, and system-level backup information for the objects that are being recovered. Find START RBA, START LRSN, or DATA COMPLETE LRSN in the output. These values indicate the first RBA or LRSN value after each copy was taken. Change the RESTOREBEFORE value in the RECOVER control statement to be higher than the smallest START RBA, START LRSN, or DATA COMPLETE LRSN value that you find in the REPORT RECOVERY output. Resubmit the job.

Severity: 8 (error)

DSNU1520I **THE RECOVERY BASE FOR** *obj-type obj-qual.obj-name dsnum n* **IS THE SYSTEM-LEVEL BACKUP WITH DATE = yyyymmdd, TIME = hhmmss, AND TOKEN = X** *byte-string*

Explanation: This message is issued by the RECOVER Utility in the RESTORE phase when a system-level backup has been chosen as the recovery base for the identified object.

System programmer response: Recovery processing continues.

Severity: 0 (informational)

DSNU1521I *csect-name* **A LOGONLY RECOVERY CANNOT BE DONE BECAUSE THE HEADER PAGE CANNOT BE ACCESSED FOR** *obj-type obj-qual.obj-name dsnum n*

Explanation: The header page could not be read. DB2 stops the RECOVER utility from proceeding with a LOGONLY recovery because the starting point is unavailable.

System programmer response: REMOVE the LOGONLY keyword from the RECOVER utility statement and recover the object by using an image copy along with a log.

Severity: 8 (error)

DSNU1522I **THE DFSMSHSM CALL TO RESTORE** *obj-type obj-qual.obj-name dsnum n* **FAILED WITH RC = X'hsm-rc' and REASON CODE = X'hsm-rsc'. SEE THE JOB LOG FOR DFSMSHSM MESSAGES INDICATING THE CAUSE OF THE ERROR**

Explanation: This message is issued by the RECOVER utility when the call to DFSMShsm to restore the data sets for the object from a system-level backup failed with the return code and reason code indicated in the message. The HSM job log will contain error messages issued by DFSMShsm that indicate the cause of the error.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) was specified, the recovery of this object will be skipped and the recovery of the other objects in the list will continue. Otherwise, the RECOVER utility will terminate.

System programmer response: Review the RECOVER utility job output and, in particular, the DFSMShsm messages in the HSM job log to determine the cause of the failure. If the restore of the data sets failed due to target volume

connectivity problems or hardware failures, the system storage administrator should be contacted to resolve the problem. After the problem is resolved, you can run the RECOVER utility job again.

If the problem with the system-level backup cannot be resolved, you can consider these recovery options to bypass the problem:

- RECOVER with the RESTOREBEFORE option to direct the recovery to find a recovery base prior to the system-level backup that has the problem. More log apply processing will be needed during the recovery of the object.
- RECOVER with the FROMDUMP option (or specify YES for the RESTORE/RECOVER FROM DUMP installation option on installation panel DSNTIP6) to direct the recovery to use a copy of the system-level backup that had been dumped to tape.
- Specify SYSTEM-LEVEL BACKUPS as NO on installation panel DSNTIP6 to avoid choosing system-level backups as a recovery base.

Problem determination: The DFSMShsm message(s) appearing in the HSM job log will indicate the cause of the failure.

Severity: 8 (error)

DSNU1523I   THE DFSMSHSM ARCXTRCT MACRO INVOCATION ISSUED BY THE RECOVER UTILITY FAILED WITH RC = X'*hsm-rc*'

Explanation: The RECOVER utility attempted to call the DFSMShsm ARCXTRCT macro, but the invocation failed.

hsm-rc   The return code from DFSMShsm.

RECOVER calls the ARCXTRCT macro to retrieve information for system-level backups for the database copy pool when the SYSTEM_LEVEL_BACKUPS subsystem parameter is set to YES.

System action: The RECOVER utility terminates.

System programmer response: Ensure that the HSM address space is active.

If you need help determining why the ARCXTRCT macro failed, call IBM Software Support for help with DFSMShsm.

Severity: 8 (error)

Related reference:

➡ RECOVER (DB2 Utilities)

➡ SYSTEM-LEVEL BACKUPS field (SYSTEM_LEVEL_BACKUPS subsystem parameter) (DB2 Installation and Migration)

Related information:

➡ ARCXTRCT: Extracting data from DFSMShsm (DFSMShsm Managing Your Own Data)

➡ Return codes from user macros (DFSMShsm) (DFSMShsm Managing Your Own Data)

DSNU1524I   *obj-type obj-qual.obj-name dsnum n* **CANNOT BE RESTORED FROM THE SYSTEM LEVEL BACKUP WITH DATE = yyyymmdd, TIME = hhmmss, AND TOKEN = X 'byte-string' BECAUSE THE SPECIFIED DUMPCLASS** *hsm-dumpclass-nm* **IS NOT KNOWN TO DFSMSHSM**

Explanation: This message is issued by the RECOVER Utility when the DFSMShsm dumpclass does not exist for the system-level backup that has been chosen as the recovery base for the identified object. The DFSMShsm dump class name can be specified on the RECOVER statement via the FROMDUMP DUMPCLASS(*hsm-dumpclass-nm*) options, or on installation panel DSNTIP6 by specifying RESTORE/RECOVER From dump => YES and DUMP CLASS NAME => *hsm-dumpclass-nm*.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) was specified, then the recovery of this object will be skipped and the recovery of the other objects in the list will continue. Otherwise, the RECOVER Utility will terminate.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Analyze the output from the LIST COPYPOOL ALLVOLS command to see which dump

classes exist for the system-level backup identified by the token. Correct the dump class specification on the RECOVER statement or on installation panel DSNTIP6, and rerun the RECOVER.

Severity: 8 (error)

DSNU1525I *obj-type obj-qual.obj-name dsnum n* **CANNOT BE RESTORED VIA FLASHCOPY FROM THE SYSTEM-LEVEL BACKUP WITH DATE = yyyymmdd, TIME = hhmmss, AND TOKEN = X** *byte-string*

Explanation: The RECOVER Utility has determined that the system-level backup chosen as the recovery base for the identified object either has not completed the background copy or does not reside on DASD, so the identified object cannot be restored via FlashCopy.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) was specified, then the recovery of this object will be skipped and the recovery of the other objects in the list will continue, otherwise the RECOVER Utility will terminate.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Analyze the output from the LIST COPYPOOL command to see whether the system-level backup identified by the token is recoverable. If the system-level backup no longer resides on DASD and has been dumped to tape, then rerun the RECOVER specifying the FROMDUMP option.

Severity: 8 (error)

DSNU1526I *obj-type obj-qual.obj-name dsnum n* **CANNOT BE RESTORED FROM THE SYSTEM-LEVEL BACKUP WITH TOKEN = X** *byte-string* **BECAUSE DFSMSHSM HAS NO RECORD OF THE SYSTEM-LEVEL BACKUP**

Explanation: The RECOVER Utility has chosen the identified system-level backup as the recovery base for this object, but DFSMShsm has no record of a system-level backup with this particular token.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) was specified, then the recovery of this object will be skipped and the recovery of the other objects in the list will continue, otherwise the RECOVER Utility will terminate.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Run the Print Log Map Utility (DSNJU004) to print the system-level backup information. If data sharing, ensure that the system-level backup information from all members is gathered. Locate the systemlevel backup information in the DSNJU004 output and compare it with the output from the LIST COPYPOOL command. DFSMShsm cannot be used as a recovery base because it has no record of the system-level backup.

The RESTOREBEFORE option may be specified on the RECOVER statement to direct the recovery to choose a recovery base that is older than the identified system-level backup, or SYSTEM-LEVEL BACKUPS => NO can be specified on installationation panel DSNTIP6 to avoid using system-level backups as a recovery base.

Severity: 8 (error)

DSNU1527I *obj-type obj-qual.obj-name dsnum n* **WAS SUCCESSFULLY RESTORED FROM A** *slb-type***, ELAPSED TIME = hh:mm:ss**

Explanation: The object was successfully restored from a system-level backup via FlashCopy or from a system-level backup that was dumped to tape. slb-type will be FLASHCOPY for the former and DUMP COPY for the latter.

System action: Recovery processing continues.

Severity: 0 (informational)

DSNU1528I *obj-type obj-qual.obj-name dsnum n* **CANNOT USE THE SYSTEM-LEVEL BACKUP WITH DATE = yyyymmdd, TIME = hhmmss AND TOKEN = X** *byte-string* **AS A RECOVERY BASE**

Explanation: This message is issued by the RECOVER utility for a point-in-time recovery when a system-level backup was chosen as the recovery base for an object. However, it cannot be used for the restore because one of the following utilities was executed on the object after the creation of the system-level backup:

- REORG TABLESPACE
- REORG INDEX
- REBUILD INDEX

- LOAD REPLACE
- RECOVER (from an image copy)

This restriction exists because the underlying data sets for the object might have been moved to different DASD volumes during the execution of these utilities. DFSMShsm requires that the data sets for an object must remain on the same DASD volume(s) to be correctly restored from the system-level backup.

System action: If OPTIONS EVENT(ITEMERROR,SKIP) was specified, the recovery of this object will be skipped and the recovery of the other objects in the list will continue. Otherwise, the RECOVER utility will terminate.

System programmer response: Refer to the *DB2 Utilities Guide and Reference* for a discussion of RECOVER utility constraints.

Severity: 8 (error)

DSNU1530I *csect-name* **- OBSOLETE OR NFM DB2 CATALOG OR DIRECTORY OBJECT** *object-type* *object-qualifier*.*object-name* **WILL NOT BE PROCESSED**

Explanation: The specified DB2 catalog or directory object is not available either because the object is no longer used by DB2 or because it will not exist until DB2 prepares to enter new-function mode.

csect-name
    The name of the control section that issued the message.

object-type
    The type of object.

object-qualifier.*object-name*
    The name of the object, as a qualified name.

This message is expected for: DB2 catalog and directory objects that will be created at the beginning of enabling-new-function mode and for obsolete DB2 catalog and directory objects that have been converted into new objects by installation job DSNTIJEN.

System action: Processing continues.

The utility continues to process other specified objects, if any exist, but the COPY, RECOVER, and REBUILD INDEX utilities skip the processing of the object that is named in this message.

For image copies that are allocated in JCL using DD statements, the COPY utility will open and close the output image copy data set if the data set resides on tape. This action happens even if this message is issued, in order to facilitate successful tape stacking of subsequent image copy data sets.

System programmer response: No action is required if DB2 is in either conversion mode or enabling-new-function mode.

If DB2 is in new-function mode, edit the COPY, RECOVER, and REBUILD INDEX jobs to remove the obsolete DB2 catalog and directory objects.

Severity: 0

Related concepts:

Job DSNTIJEN (DB2 Installation and Migration)

DSNU1540I *csect-name* **- CONSISTENCY PROCESSING FOR FLASHCOPY IS STARTED.**

Explanation: FlashCopy consistency processing has begun.

csect-name
    The name of the control section that issued the message.

System action: Processing continues.

Message DSNU15401I is issued when consistency processing finishes.

User response: Look for other messages that might indicate problems that require action.

Severity: 0

DSNU1541I *csect-name* **- CONSISTENCY PROCESSING FOR FLASHCOPY IS FINISHED, ELAPSED TIME=**_elapsed-time_

Explanation: FlashCopy consistency processing has completed.

csect-name
> The name of the control section that issued the message.

elapsed-time
> The time elapsed since the beginning of FlashCopy consistency processing.

System action: Processing continues.

User response: Look for other messages that might indicate problems that require action. The beginning of FlashCopy consistency processing is indicated by message DSNU1540I.

Severity: 0

DSNU1542I *csect-name* **- RECOVER BACKOUT ANALYSIS PROCEEDING FROM CURRENT LOGPOINT OF X'**_current-logpoint_**'**

Explanation: This message indicates the starting point in the DB2 log for the analysis of RECOVER BACKOUT processing for all objects in the list.

csect-name
> The name of the control section that issued the message.

current-logpoint
> The RBA or LRSN of the current point in the log.

System action: RECOVER processing continues.

User response: No action is required.

Severity: 0

Related reference:

↪ RECOVER (DB2 Utilities)

DSNU1543I *csect-name* **- FCCOPYDDN TEMPLATE IS INVALID BECAUSE IT DOES NOT CONTAIN PART OR DSNUM**

Explanation: The template the was used for a FlashCopy did not include &part(&pa) or &dsnum(&ds). One of these variables is required in the template.

csect-name
> The name of the control section that issued the message.

System action: The utility stops processing.

User response: Correct the template, which can be specified in the utility control statement or as a subsystem parameter.

Severity: 8 (error)

DSNU1544I *csect-name* **RECOVER BACKOUT YES CANNOT PROCEED ON** *object-type object-qualifier.object-name* **DUE TO SQL MASS DELETE OR OTHER TABLE ACTIVITY**

Explanation: The BACKOUT YES option was specified for the RECOVER utility, but the utility cannot proceed because of one the following operations on the object:
- SQL mass DELETE
- TRUNCATE TABLE
- DROP TABLE

The deleted data for these operations is not logged and therefore cannot be backed out. A BACKOUT YES recovery is prohibited when one of these operations has occurred between the recovery point (specified by TOLOGPOINT or TORBA) and the current time.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

System action: The RECOVER job fails.

User response: Remove the BACKOUT YES option from the RECOVER utility statement and resubmit the job. This revised statement indicates that the utility is to recover the list of objects by restoring a backup and then applying the logs forward to the recovery point.

Severity: 8 (error)

Related concepts:

[▶] Point-in-time recovery (DB2 Utilities)

Related reference:

[▶] Syntax and options of the RECOVER control statement (DB2 Utilities)

DSNU1545I *csect-name* **RECOVER WITH BACKOUT YES CANNOT PROCEED. THE RECOVERY POINT IS OLDER THAN THE EARLIEST CHECKPOINT IN THE BSDS.**

Explanation: The recovery point that is identified by TOLOGPOINT or TORBA is not contained within the most recent DB2 system checkpoints that are recorded in the BSDS for each member. RECOVER with the BACKOUT YES option cannot proceed, because the checkpoint RBA that is before the recovery point for each member is required for backout processing.

csect-name
> The name of the control section that issued the message.

System action: The RECOVER utility ends with return code 8 and does not process any of the objects. The objects are left in the state that they were in before the RECOVER operation.

System programmer response: Remove the BACKOUT YES option and resubmit the RECOVER job. This revised job recovers the objects to the same point in time. However, the data is recovered by using a recovery base and applying the logs rather than using backout processing.

Recommendation: Use the output from the REPORT utility with the RECOVERY option for all of the objects to ensure that a valid recovery base exists before you run RECOVER.

Severity: 8 (error)

Related concepts:

[▶] Recovery information that REPORT provides (DB2 Utilities)

Related reference:

[▶] Syntax and options of the RECOVER control statement (DB2 Utilities)

DSNU1550I *csect-name* **LOGCSR IS STARTED FOR MEMBER** *member-name***; PRIOR CHECKPOINT RBA=** *rba-value*

Explanation: This message is issued at the beginning of the log analysis process by the RECOVER utility for the specified member. This message is issued for recover to point-in-time with consistency, only when the objects at the recovery point on the specified member have active units of recovery. This message identifies the RBA of the checkpoint from which the RECOVER utility will process the log records during the LOGCSR phase for this DB2 member.

In a data sharing environment, *member-name* identifies the DB2 member. In a non-data sharing environment, *member-name* is blank.

System action: The RECOVER utility starts the log analysis process for the specified DB2 member.

Severity: 0 (informational)

Related concepts:

How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU1551I   *csect-name* **LOGCSR IS FINISHED FOR MEMBER** *member-name*; **ELAPSED TIME =** *hh:mm:ss*

Explanation: This message is issued at the completion of the log analysis process by the RECOVER utility for the specified member. This messages also shows the elapsed time for the LOGCSR phase on the specified DB2 member. The format of the elapsed time is 'hh:mm:ss' (hours, minutes, and seconds).

In a data sharing environment, *member-name* identifies the DB2 member. In a non-data sharing environment, *member-name* is blank.

System action: If there are other DB2 members that require log analysis, the analysis continues with the next member. Otherwise, the DSNU1552i message is issued to mark the end of the LOGCSR phase for this RECOVER utility.

Severity: 0 (informational)

DSNU1552I   *csect-name* **LOGCSR PHASE COMPLETE; ELAPSED TIME=** *hh:mm:ss*

Explanation: This message is issued at the completion of LOGCSR phase of the RECOVER utility. This message is issued for recover to point-in-time with consistency, only when the objects at the recovery point on any of the DB2 members have active units of recovery. This message shows the total elapsed time for the LOGCSR phase. The format of the elapsed time is 'hh:mm:ss' (hours, minutes, and seconds).

System action: The RECOVER utility continues to the next phase for processing.

Severity: 0 (informational)

DSNU1553I   *csect-name* *utility-name* **DETECTS THE FOLLOWING ACTIVE URS: INFLIGHT=** *inflight-count*, **INABORT=** *inabort-count*, **INDOUBT=** *indoubt-count*, **POSTPONED ABORT=** *postponed-count* **COMMITTED=** *committed-count* **ABORTED=** *aborted-count*

Explanation: The LOGCSR phase of the utility is complete or was skipped because no active units of recovery (URs) existed at the recovery point. The counts of units of recovery in the message indicate the number of units of work that have execution states that require rollback action to be performed during this recovery process in order to ensure recovery consistency.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the utility: RECOVER or FLASHCOPY WITH CONSISTENCY.

inflight-count
> The number of inflight units of recovery. These units of recovery neither completed phase one of the commit process nor began the process of backing out during the recovery point. The utility will roll back the changes during the LOGUNDO phase.

inabort-count
> The number of in-abort units of recovery. These units of recovery were in the process of being backed out during recovery point. The utility will roll back the changes during the LOGUNDO phase.

indoubt-count
> The number of indoubt units of recovery. These units of recovery were interrupted between phase one and phase two of the commit process. They are treated as in-abort units of recovery, and the utility will roll back the changes during the LOGUNDO phase.

DSNU1553I

postponed-count
> The number of postponed in-abort units of recovery. The backout of these units of recovery was postponed in an earlier restart process, and they were not attended to prior to the recovery point. The utility will roll back the changes during the LOGUNDO phase.

committed-count
> The number of committed units of recovery. These units of recovery will be backed out during RECOVER backout log apply processing.

aborted-count
> The number of units of recovery that have completed abort processing. These units of recovery will be backed out during RECOVER backout log apply processing.

If an active unit of recovery exists during the recovery point, the message text is followed by a table that shows the status of units of recovery that were active during the recovery to point-in-time on the objects being recovered. This table helps to identify the following conditions:

- Units of recovery that have changes on the recovered objects which need to be rolled back during the LOGUNDO phase.
- The scope of log records that require processing in order to roll back the changes.

For each active unit of recovery in the table, the table contains one line that shows information about the unit of recovery:

```
MEM T CONID CORRID AUTHORIZATION ID PLAN S
URID DAY TIME
```

MEM   The name of theDB2 member to which this unit of recovery belongs. In a data sharing environment, this field identifies theDB2 member. In a non-data sharing environment, this field is blank.

T   The connection type:

    B   Batch: A batch program or QMF user.

    I   Internal: A system service being performed on a user's behalf.

    S   Subsystem: Work that originated from an attached subsystem, either CICS or IMS/VS.

CONID
> The connection identifier for related units of recovery. Batch connections are not related to any other connection. Subsystem connections with the same identifier indicate units of recovery that originated from the same subsystem.

CORRID
> The correlation identifier, which is unique within a subsystem connection. It relates a unit of recovery to a particular CICS or IMS/VS transaction.

AUTHORIZATION ID
> Authorization identifier of the user for whom this unit of recovery was created.

PLAN   Name of the plan that is allocated to the unit of recovery.

S   Status of the unit of recovery during the recovery point:

    D   Indoubt: The unit of recovery had completed the first phase of commit, but DB2 had not received the second phase instruction. Its change on the objects being recovered must be rolled back.

    E   Committed: The unit of recovery has been committed.

    F   Inflight: The unit of recovery had not completed the first phase of commit. Its change on the objects being recovered must be rolled back.

    P   Postponed abort: The unit of recovery has not yet completed backout afterDB2 was last restarted. Its change on the objects being recovered must be rolled back.

URID   The unit of recovery identifier, which is the log RBA of the beginning of this unit of work.

DAY   The day that the unit of recovery was created, in the format *ddd*. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

TIME The time the unit of recovery was created, in the format *hh:mm:ss*. It is approximately the time of the first SQL statement of the application or the first SQL statement following a commit point.

This line is followed by one or more lines that show information about the objects that are included in the job and are modified by this unit of recovery. If the object is partitioned, one line represents each partition that is included in the job and is modified by this unit of recovery:

```
DBNAME SPACENAME DBID/PSID PART RBA
```

DBNAME
The name of the database to which the table space or index space belongs.

SPACENAME
The table space or index space that is changed by this unit of recovery.

DBID/PSID
The database and page set object identifiers (OBIDs).

PART The partition number, if the table space or index space is partitioned.

RBA In a non data-sharing environment, the RBA of the first update made by this unit of recovery on this object. In a data-sharing environment, the LRSN of the first update made by this unit of recovery on this object.

System action: The LOGUNDO phase begins.

User response: No action is required.

Severity: 0

DSNU1554I *csect-name* **LOGUNDO IS STARTED FOR MEMBER** *member-name*

Explanation: This message signals the beginning of the LOGUNDO phase by the RECOVER Utility or FlashCopy with consistency. This message is issued in the following situations:

- A point-in-time recovery with consistency when the objects at the recovery point on any DB2 member have active units of recovery (URs). The changes made by these URs on the objects that are being recovered need to be rolled back in the LOGUNDO phase.
- A RECOVER utility job with the BACKOUT YES option
- A FlashCopy with consistency

csect-name
The name of the control section that issued the message.

member-name
The DB2 member. In the following situations, *member-name* is blank:

- In a non-data sharing environment
- During a FlashCopy with consistency
- During a RECOVER utility job with the BACKOUT YES option

For the last two situations (FlashCopy with consistency and RECOVER BACKOUT YES), DB2 performs a merged LOGUNDO phase instead of a LOGUNDO phase for each member.

System action: The RECOVER utility or FlashCopy with consistency starts the LOGUNDO phase for the DB2 member or members.

Operator response: No action is required.

Severity: 0

Related concepts:

➡ Point-in-time recovery (DB2 Utilities)

➡ FlashCopy image copies (DB2 Utilities)

Related reference:

➡ RECOVER (DB2 Utilities)

Related information:

"DSNU1555I"

"DSNU1556I"

DSNU1555I *csect-name* **RECOVER LOGUNDO STATUS: LOG RECORD AT RBA** *current-rba* **TO RBA** *last-rba* **ON MEMBER** *member-name*

Explanation: This message is produced periodically during the LOGUNDO phase of the RECOVER utility or FlashCopy with consistency. The RBAs or LRSNs in this message can be used to monitor the progress of the LOGUNDO phase. The RBA values apply to a non-data sharing environment. The LRSN values apply to a data-sharing environment.

csect-name
> The name of the control section that issued the message.

current-rba
> The RBA or LRSN of the log record that DB2 is currently processing.

last-rba The RBA or LRSN of the last log record that DB2 will read in this LOGUNDO phase.

member-name
> The DB2 member. In the following situations, *member-name* is blank:
> - In a non-data sharing environment
> - During a FlashCopy with consistency
> - During a RECOVER utility job with the BACKOUT YES option
>
> For the last two situations (FlashCopy with consistency and RECOVER BACKOUT YES), DB2 performs a merged LOGUNDO phase instead of a LOGUNDO phase for each member.

System action: The RECOVER utility or FlashCopy with consistency continues processing.

Operator response: No action is required.

Severity: 0

Related concepts:

➡ Point-in-time recovery (DB2 Utilities)

➡ FlashCopy image copies (DB2 Utilities)

➡ How RBA and LRSN values are displayed (DB2 Administration Guide)

Related reference:

➡ RECOVER (DB2 Utilities)

Related information:

"DSNU1554I" on page 909

"DSNU1556I"

DSNU1556I *csect-name* **LOGUNDO IS FINISHED FOR MEMBER** *member-name* **; ELAPSED TIME=** *elapsed-time*

Explanation: This message is issued at the completion of the LOGUNDO phase by the RECOVER utility or FlashCopy with consistency.

csect-name
> The name of the control section that issued the message.

member-name
> The DB2 member. In the following situations, *member-name* is blank:
> - In a non-data sharing environment
> - During a FlashCopy with consistency
> - During a RECOVER utility job with the BACKOUT YES option
>
> For the last two situations (FlashCopy with consistency and RECOVER BACKOUT YES), DB2 performs a merged LOGUNDO phase instead of a LOGUNDO phase for each member.

elapsed-time
> The elapsed time for this LOGUNDO phase. The format of the elapsed time is *hh:mm:ss* where:
>
> | *hh* | hours |
> | *mm* | minutes |
> | *ss* | seconds |

System action: If the RECOVER utility needs to perform rollback operations on other DB2 members, processing continues for the next DB2 member. Otherwise, the DSNU1557I message is issued to mark the end of the LOGUNDO phase.

Operator response: No action is required.

Severity: 0

Related concepts:

▶ Point-in-time recovery (DB2 Utilities)

▶ FlashCopy image copies (DB2 Utilities)

Related reference:

▶ RECOVER (DB2 Utilities)

Related information:
"DSNU1554I" on page 909
"DSNU1555I" on page 910
"DSNU1557I"

DSNU1557I *csect-name* **LOGUNDO PHASE COMPLETE; ELAPSED TIME=** *hh:mm:ss*

Explanation: This message is issued at the completion of the LOGUNDO phase of the RECOVER utility. This message shows the total elapsed time for the LOGUNDO phase. The format of the elapsed time is 'hh:mm:ss' (hours, minutes, and seconds).

System action: The RECOVER utility continues to the next phase of processing.

Severity: 0 (informational)

DSNU1558I *csect-name* **- PRELOGC IS STARTED FOR MEMBER** *member-name*, **PRIOR CHECKPOINT RBA=***rba-value*

Explanation: The RECOVER utility began the log analysis process for the specified member, and one or more of the objects has a recovery base that is a FlashCopy with consistency, and there were active units of recovery during the FlashCopy. The value of *rba-value* is the RBA value of the checkpoint from which the RECOVER utility will process the log records during the PRELOGC phase for this DB2 member. In a data sharing environment, *member-name* identifies the DB2 member. In a non-data sharing environment, *member-name* is blank.

csect-name
> The name of the control section that issued the message.

System action: The RECOVER utility starts the preliminary log analysis process for the specified member.

User response: No action is required.

Severity: 0

DSNU1559I • DSNU1561I

Related concepts:

How RBA and LRSN values are displayed (DB2 Administration Guide)

Related reference:

RECOVER (DB2 Utilities)

DSNU1559I *csect-name* **PRELOGC IS FINISHED FOR MEMBER** *member-name*, **ELAPSED TIME=***hh:mm:ss*

Explanation: The RECOVER utility completed the preliminary log analysis for the specified member.

csect-name
> The name of the control section that issued the message.

hh:mm:ss
> The elapsed time for the operation, in hours, minutes, and seconds.

System action: If other DB2 members require preliminary log analysis, the analysis continues withe the next member. Otherwise, DB2 issues a DSNU1560I message to indicate the end of the PRELOGC phase for this RECOVER utility.

User response: No action is required.

Severity: 0

Related reference:

RECOVER (DB2 Utilities)

DSNU1560I *csect-name* **PRELOGC PHASE COMPLETE, ELAPSED TIME=***hh:mm:ss*

Explanation: the RECOVER utility completed the preliminary log analysis phase, and one or more of the objects has a recovery base that is a FlashCopy with consistency and there were active units of recovery during the FlashCopy.

csect-name
> The name of the control section that issued the message.

hh:mm:ss
> The elapsed time for the operation, in hours, minutes, and seconds.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

RECOVER (DB2 Utilities)

DSNU1561I *csect-name* **- PRELOGA PHASE COMPLETE, ELAPSED TIME=***hh:mm:ss*

Explanation: The RECOVER utility completed the RELOGA phase, and one or more of the objects has a recovery base that is a FlashCopy with consistency and there were active units of recovery during the FlashCopy.

csect-name
> The name of the control section that issued the message.

hh:mm:ss
> The elapsed time for the operation, in hours, minutes, and seconds.

System action: Processing continues.

User response: No action is required.

Severity: 0

RECOVER (DB2 Utilities)

DSNU1562I *csect-name* **- FLASHCOPY** *dataset-name* **FOR** *object-type database-name***.***tablespace-name* **HAS BEEN MIGRATED BY HSM**

Explanation: A FlashCopy image copy was chosen by the RECOVER utility as a recovery base, but it has been migrated by HSM.

csect-name
> The name of the control section that issued the message.

dataset-name
> The name of the data set.

*database-name***.***tablespace-name*
> The name of the database and table space.

object-type
> The type of object.

database-name
> The name of the database that is associated with the recovery.

This message has two forms. The short form is shown above. The long form is:

```
csect-name - FLASHCOPY dataset-name FOR
          object-type
database-name.tablespace-name
              DSNUM part-number HAS BEEN
          MIGRATED BY HSM
```

The long form of this message is issued when the specified object is partitioned or has multiple data sets.

part-number
> The value that identifies the partition or data set.

System action: The RECOVER utility ignores the SYSCOPY record and looks for and chooses a subsequent image copy as a recover base. The RECOVER utility looks for a local primary or local backup before looking for an older image copy. Sometimes no previous image copies can be used for the recovery. In this case a DSNU510I message is issued with a return code of 8.

User response: Request HSM to recall the data set, and resubmit the RECOVER utility.

Severity: 0

Related reference:

RECOVER (DB2 Utilities)

Related information:
"DSNU510I" on page 719

DSNU1563I *csect-name* **- FLASHCOPY** *dataset-name* **FOR** *object-type database-name***.***tablespace-name* **DSNUM** *part-number* **HAS BEEN DELETED**

Explanation: A FlashCopy image copy chosen by the RECOVER utility as a recovery base has been deleted.

csect-name
> The name of the control section that issued the message.

dataset-name
> The name of the data set.

*database-name***.***tablespace-name*
> The name of the database and table space.

object-type
> The type of object.

database-name
> The name of the database that is associated with the recovery.

part-number
> The value that identifies the partition or data set.

System action: The RECOVER utility ignores the SYSCOPY record and looks for another recovery base.

User response: No action is required.

Severity: 0

Related reference:

RECOVER (DB2 Utilities)

DSNU1600I *csect-name* **- BACKUP SYSTEM UTILITY FOR** *object* **STARTING, COPY POOL =** *copy-pool-name*, **TOKEN = X'***token***'**

Explanation: Backup of the indicated object, either data or logs, to the indicated copy pool and with the indicated token is starting.

System action: Backup of the indicated object has started.

System programmer response: None.

Severity: 0 (informational)

DSNU1601I *csect-name* **- BACKUP SYSTEM UTILITY IS NOT RESTARTABLE. RESTART IGNORED. A NEW BACKUP WILL BE INITIATED**

Explanation: The BACKUP SYSTEM utility cannot be restarted. The restart is ignored and a new backup is done.

System action: The restart is ignored, and a new BACKUP SYSTEM request is initiated.

System programmer response: None.

Severity: 0 (informational)

DSNU1602I *csect-name* **- BACKUP SYSTEM UTILITY COMPLETED, ELAPSED TIME =** *hh:mm:ss*

Explanation: The BACKUP SYSTEM utility cannot be restarted. The restart is ignored, and a new backup is done.

System action: None.

System programmer response: None.

Severity: 0 (informational)

DSNU1603I *csect-name* **- RESTORE SYSTEM UTILITY RESTART FAILED BECAUSE THE MEMBER-ID,***member-id***, ON WHICH THE RESTART HAS BEEN ISSUED IS DIFFERENT FROM THE MEMBER-ID,** *member-id***, ON WHICH THE RESTORE WAS ORIGINALLY DONE**

Explanation: A group RESTORE in a data sharing environment is already in the log apply phase. The RESTORE was originally issued and owned by a member other than the one that issued the restart, which is not allowed.

System action: The restart fails.

System programmer response: Issue the restart job from the member that was indicated in the message.

Severity: 8 (error)

DSNU1604I *csect-name* **- RESTORE SYSTEM PHASE LOG APPLY STARTED AT LOG POINT =** *log-point-value*

Explanation: The RESTORE SYSTEM utility has begun applying logs beginning at the log-point-value in order to arrive at the point in time indicated by the log point at which the logs were truncated via the SYSPITR parameter on the conditional restart command, CRESTART CREATE.

System programmer response: None.

Severity: 0 (informational)

DSNU1606I *csect-name* **- RESTORE SYSTEM UTILITY STARTING, COPY POOL =** *copy-pool-name***, TOKEN = X'***token***'**

Explanation: Restore of the indicated copy pool with the indicated token is starting.

System action: None.

System programmer response: None.

Severity: 0 (informational)

DSNU1607I *csect-name* **- RESTORE SYSTEM UTILITY REQUIRES INSTALL SYSADM AUTHORITY TO RUN. THE CALLER HAS INSUFFICIENT AUTHORITY**

Explanation: The invoker of the RESTORE SYSTEM utility lacks the installation SYSADM authority required to run the utility.

System action: The utility is terminated.

System programmer response: Ensure that the RESTORE SYSTEM utility is run by an administrator with installation SYSADM authority.

Severity: 8 (error)

DSNU1608I *csect-name* **-** *system-utility* **SYSTEM UTILITY FAILED. REQUIRED DFSMSHSM SERVICES NOT AVAILABLE.**

Explanation: The system that attempted to run the BACKUP SYSTEM or RESTORE SYSTEM utility does not have the version of z/OS that contains the version of DFSMSHSM that is required by those utilities.

System action: The utility is terminated.

System programmer response: Ensure that the required version of the z/OS operating system is installed on the system on which the BACKUP SYSTEM and RESTORE SYSTEM utilities are to be run.

Severity: 8 (error)

DSNU1609I *csect-name* **- BACKUP SYSTEM UTILITY REQUIRES SYSADM OR SYSCTRL AUTHORITY TO RUN. THE CALLER HAS INSUFFICIENT AUTHORITY**

Explanation: The invoker of the BACKUP SYSTEM utility lacks the SYSADM or SYSCTRL authority required to run the utility.

System action: The utility is terminated.

System programmer response: Ensure that the BACKUP SYSTEM utility is run by an administrator with SYSADM or SYSCTRL authority.

Severity: 8 (error)

DSNU1610I *csect-name* **- AN INCOMPLETE BACKUP SYSTEM HISTORY ENTRY HAS BEEN FOUND IN THE BSDS, INDICATING THERE MAY BE AN INVALID DATA AND LOG VERSION IN COPYPOOL =** *copy-pool-name* **FOR TOKEN = X'***token***'. DFSMSHSM WILL BE CALLED TO DELETE THE VERSION**

Explanation: During Backup System processing, an incomplete backup history entry was found in the BSDS. For example, an incomplete entry can exist if DB2 or z/OS goes down before backup processing can complete. Such an entry indicates that a data and log version may exist in the indicated copy pool for the indicated token.

System action: Backup System will invoke DFSMShsm to delete the version of the data and logs before proceeding with the current backup.

System programmer response: None.

Severity: 0 (informational)

DSNU1611I *csect-name* **- A FAILURE HAS OCCURRED DURING BACKUP SYSTEM PROCESSING. DFSMSHSM IS BEING CALLED TO DELETE THE INCOMPLETE VERSION IN COPYPOOL =** *copy-pool-name* **FOR TOKEN = X'**token**'**

Explanation: A failure that was previously indicated during Backup System processing has created an incomplete version. The copy pool and token with the incomplete version are indicated in the message text. This incomplete version must be deleted.

System action: Backup System will invoke DFSMShsm to delete the incomplete version.

System programmer response: None.

Severity: 0 (informational)

DSNU1612I *csect-name* **- RESTORE SYSTEM UTILITY FAILED BECAUSE THE SYSTEM IS NOT IN SYSTEM RECOVER PENDING STATE.**

Explanation: The RESTORE SYSTEM utility cannot be run unless the proper preparations have been made. That preparation includes truncation of logs and performing a conditional restart of the appropriate systems to place them in the System RECOVER PENDING state.

System action: The utility is terminated.

System programmer response: Complete the required preparations and retry the RESTORE SYSTEM utility.

Severity: 8 (error)

Related concepts:

⬀ Recovery of data to a prior point in time (DB2 Administration Guide)

⬀ Point-in-time recovery (DB2 Utilities)

⬀ System-level point-in-time recovery (DB2 Data Sharing Planning and Administration)

Related tasks:

⬀ Preparing to recover to a prior point of consistency (DB2 Administration Guide)

Related reference:

⬀ RESTORE SYSTEM (DB2 Utilities)

DSNU1614I *csect-name* **- BACKUP SYSTEM UTILITY FOR** *object-name*object **COMPLETED SUCCESSFULLY, COPY POOL =** *copy-pool-name*, **TOKEN = X'**token-value**', DATA COMPLETE LRSN = X'**lrsn-value**', ELAPSED TIME =** *elapsed-time*

Explanation: The backup of the indicated object, data or logs, has completed successfully.

csect-name
> The name of the control section that issued the message.

object-name
> The identifier of the object that was backed up.

copy-pool-name
> The name of the copy pool.

token-value
> The TOKEN value that was specified in the utility control statement.

lrsn-value
> The RBA or LRSN of the last log record written to the active log data sets after the backup of the database

copy pool was complete. Note that the same data complete LRSN value is reported in the DSNU1614I messages for the data and logs. When you copy both the database and log copy pools use this data complete LRSN as the log truncation point for the conditional restart of DB2 for system recovery to a point in time if the log copy pool is restored, or for cloned systems created with a system-level backup.

elapsed-time
>   The elapsed time of the utility in hours, minutes, and seconds, shown in the following format: *hh:mm:ss*

System action: Processing continues.

System programmer response: When this message is issued for the database copy pool, capture the data complete LRSN. You can use the LRSN to determine the following information:

- Which system-level backup is used by RESTORE SYSTEM during system point in time recovery. The RESTORE SYSTEM utility chooses a system-level backup (for the restore of the database copy pool) when the data complete LRSN is less than the SYSPITR log point.
- The CRESTART ENDRBA, ENDLRSN, or SYSPITR log truncation point for conditional restart of DB2 when the log copy pool has been restored outside of DB2
- The CRESTART ENDRBA or ENDLRSN for the conditional restart of DB2 for cloned systems

You can use the Print Log Map Utility, DSNJU004, to print information for system-level backups, including the data complete LRSN. For a data sharing systems, invoke DSNJU004 on the BSDS for the member where the backup was invoked.

For a conditional restart of DB2, use the Change Log Inventory Utility, DSNJU003, to create the conditional restart record (CRESTART).

Severity: 0

Related concepts:

➦ How RBA and LRSN values are displayed (DB2 Administration Guide)

Related reference:

➦ BACKUP SYSTEM (DB2 Utilities)

➦ RESTORE SYSTEM (DB2 Utilities)

➦ DSNJU004 (print log map) (DB2 Utilities)

➦ DSNJU003 (change log inventory) (DB2 Utilities)

DSNU1615I   DUMPONLY WAS SPECIFIED, BUT NO SYSTEM-LEVEL BACKUP CORRESPONDING TO THE TOKEN EXISTS

Explanation: The request to dump a specific system-level backup to tape has failed because either:

- A system-level backup with the specified token does not exist, or
- The DASD version of the system-level backup is invalid

System action: BACKUP SYSTEM processing terminates without dumping the system-level backup to tape.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Run the Print Log Map Utility (DSNJU004) to print the system-level backup information. If data sharing, ensure that the system-level backup information from all members is gathered. Locate the system-level backup information in the DSNJU004 output and compare it with the output from the LIST COPYPOOL command.

Severity: 8 (error)

DSNU1616I   *csect-name* - THE CALL TO DFSMShsm TO RESTORE SYSTEM FAILED. PLEASE SEE MESSAGE ARC1806E IN THE OUTPUT FOR DETAILS. COPY POOL = *copy-pool-name*, TOKEN = X'*token*'. A RETRY WILL BE ATTEMPTED IF AN APPROPRIATE BACKUP VERSION CAN BE IDENTIFIED.

Explanation: The call to DFSMShsm to RESTORE SYSTEM with the indicated copy pool name and token failed during DFSMShsm processing. DFSMSHSM will have issued message ARC1806E indicating the cause of the error. This message can be found in the job log of the RESTORE SYSTEM utility.

If this is the first attempt by the RESTORE SYSTEM utility to do the RESTORE, a second attempt will be made if there is another appropriate Backup version recorded in the BACKUP SYSTEM history BSDS record. An appropriate Backup version is one for which data backup completed before the log truncation point indicated in the SYSPITR parameter of the conditional restart command, CRESTART CREATE. Messages indicating the progress of a subsequent attempt will be issued, just as for the first attempt.

If the second attempt fails, no further attempts will be made and the request will be failed.

System action: If this is the first attempt to do the RESTORE and there is another appropriate Backup available, a second attempt will be made. Otherwise the utility will terminate.

System programmer response: Correct the problem identified in message ARC1806E and rerun the utility.

Problem determination: The job log will contain the ARC1806E message that is issued by DFSMShsm. This message will contain information indicating the cause of the failure in DFSMShsm.

Severity: 8 (error)

DSNU1618I *csect-name* **- RESTORE SYSTEM UTILITY FAILED BECAUSE THERE ARE NO BACKUP SYSTEM HISTORY ENTRIES IN THE BSDS**

Explanation: No BACKUP SYSTEM history entries exist in the BSDS. Without these records, the RESTORE SYSTEM utility cannot restore the system. Backups of the target systems must be taken before any restore operations can be done.

csect-name
> The name of the control section that issued the message.

If you ran RESTORE SYSTEM with the FROMDUMP option, references in this message to BACKUP SYSTEM history entries refer only to history entries for backups that were dumped to tape.

System action: The utility terminates.

System programmer response: None.

Severity: 8 (error)

Related reference:

↪ RESTORE SYSTEM (DB2 Utilities)

DSNU1619I *csect-name* **- RESTORE SYSTEM UTILITY FAILED, BECAUSE THERE ARE NO BACKUP SYSTEM HISTORY ENTRIES IN THE BSDS WITH A DATA COMPLETION LRSN/RBA PRIOR TO THE LOG TRUNCATION POINT**

Explanation: The RESTORE SYSTEM utility failed, because the logs have been truncated to a point before which any backups are available.

csect-name
> The name of the control section that issued the message.

BACKUP SYSTEM history entries exist in the BSDS. However, none of them indicate a data backup completion LRSN or RBA with a value less than the log truncation point. The log truncation point is indicated in the SYSPITR parameter of the conditional restart command, CRESTART CREATE, that is used to truncate the logs.

If you ran RESTORE SYSTEM with the FROMDUMP option, references in this message to BACKUP SYSTEM history entries refer only to history entries for backups that were dumped to tape.

System action: The utility terminates.

Severity: 8 (error)

Related concepts:

Point-in-time recovery with system-level backups (DB2 Administration Guide)

A conditional restart control record (DB2 Utilities)

How RBA and LRSN values are displayed (DB2 Administration Guide)

Related tasks:

Managing the log and the bootstrap data set (DB2 Administration Guide)

Related reference:

BACKUP SYSTEM (DB2 Utilities)

RESTORE SYSTEM (DB2 Utilities)

DSNU1621I *csect-name* - *utility-name* **UTILITY FAILED BECAUSE THE SYSTEM IS IN SYSTEM RECOVER-PENDING STATE**

Explanation: When the system is in RECOVER-PENDING state, no utility other than RESTORE is allowed to run. An attempt was made to run a utility while the system was in RECOVER-PENDING state.

System action: The utility is terminated.

System programmer response: Review the need to run the utility and, if appropriate, resubmit your job after the system has been restored.

Severity: 8 (error)

DSNU1623I *csect-name* - **THE TERM UTILITY REQUEST IS REJECTED BECAUSE THE SYSTEM IS IN SYSTEM RECOVER PENDING STATE**

Explanation: The TERM request is rejected. The only utility that is allowed to run while the system is in the system RECOVER pending state is the RESTORE utility.

System action: The TERM request is ignored.

Severity: 4 (warning)

DSNU1624I *csect-name* - *system-utility* **SYSTEM UTILITY FAILED BECAUSE ONE OR MORE NON-DORMANT MEMBERS ARE IN A FAILED OR NOT NORMALLY QUIESCED STATE. THOSE MEMBERS ARE:** *member-id*

Explanation: The BACKUP or RESTORE of the members of a data sharing group requires that all members be active or normally quiesced. If any members are found not to be in either of these states, the BACKUP or RESTORE cannot continue.

System action: The utility is terminated.

System programmer response: Determine why these systems are in a problem state, and whether they can be moved out of this state. Once all members have been returned to either the active or normally quiesced state, resubmit the job.

Severity: 8 (error)

DSNU1626I *csect-name* - *system-utility* **SYSTEM UTILITY FAILED BECAUSE RESTORE SYSTEM IS ALREADY RUNNING OR ANOTHER TASK IS PROCESSING IN A MANNER THAT PREVENTS THIS TASK FROM SERIALIZING PROPERLY**

Explanation: The attempt to acquire the serialization lock failed. In the case of RESTORE SYSTEM, this can only occur if RESTORE SYSTEM is already running. In the case of BACKUP SYSTEM, this can occur if any of the following events are in process:

- System checkpoint
- Data set extensions
- Writing 32K pages

- Writing close page set control log records (PSCRs)
- Creating data sets (for table spaces, indexes, and so on)
- Deleting data sets (for dropping tables spaces, indexes, and so on)
- Renaming data sets (for online reorganizing of table spaces, indexes, and so forth during the switch phase)

System action: The utility is terminated.

System programmer response: If it is the RESTORE SYSTEM utility that is being invoked, the only other utility that can be running is RESTORE SYSTEM. If it is the BACKUP SYSTEM utility that is being invoked, any system activity that provokes any of the processes listed in the Explanation above could be the cause of the failure. If possible, determine which utility is responsible and resubmit this job once that utility completes. You may also resubmit your job at a later, less busy time.

Severity: 8 (error)

DSNU1627I *csect-name* **- RESTORE SYSTEM PRE-LOG APPLY COMPLETED SUCCESSFULLY, COPY POOL =** *copy-pool-name* **TOKEN = X'***token***' ELAPSED TIME =** *hh:mm:ss***.**

Explanation: Restoration of the data from the indicated copy pool with the indicated token has completed successfully.

System action: Continue with log apply phase.

System programmer response: None.

Severity: 0 (informational)

DSNU1628I *csect-name* **- RESTORE SYSTEM PHASE LOG APPLY COMPLETED, ELAPSED TIME =** *hh:mm:ss*

Explanation: The log apply phase of the RESTORE SYSTEM completed in the amount of time that was indicated.

Severity: 0 (informational)

DSNU1629I *csect-name* **- DB2 PUT ONE OR MORE OBJECTS INTO THE RECOVER-PENDING STATE, REBUILD-PENDING STATE, OR LOGICAL PAGE LIST DURING THE LOG APPLY PHASE.**

Explanation: During the log apply process, objects were put into one of the exception states that is indicated in the message. Before these objects can be used, they must be processed again after the Restore completes.

System action: None.

System programmer response: Process those objects in the RECOVER-PENDING or REBUILD-PENDING states with the REBUILD or RECOVER utilities as appropriate. See the Utility Guide and Reference for further details.

To recover pages from the LPL, use the -START DATABASE command with the SPACENAM option, the RECOVER utility, or the LOAD utility.

Problem determination: Use the DISPLAY DATABASE command to identify which objects are in the exception states indicated in the message.

Severity: 0 (informational)

DSNU1630I *csect-name* **BACKUP SYSTEM UTILITY FAILED, BECAUSE action FCINCREMENTAL WAS SPECIFIED BUT THE INCREMENTAL FLASHCOPY COULD NOT BE PROCESSED**

Explanation: This message is issued when all of the following conditions exist:

- The database copy pool has been defined with more than one DASD version.
- A persistent incremental FlashCopy relationship already exists for the database copy pool.
- The next version or backup of the database copy pool to be taken is for a full FlashCopy rather than an incremental FlashCopy.

If the action specified is ESTABLISH, then a persistent incremental FlashCopy relationship already exists for the database copy pool.

If the action specified is END, then the persistent incremental FlashCopy relationship cannot be ended because the next version of the database copy pool to be taken is for a full FlashCopy, rather than for the incremental FlashCopy.

System programmer response: Remove the ESTABLISH FCINCREMENTAL option or the END FCINCREMENTAL option and rerun the BACKUP SYSTEM utility job. This will create a full FlashCopy of the database copy pool. It will also create the log copy pool if the FULL option was specified.

If you specified ESTABLISH FCINCREMENTAL and it is important based on your schedule of backing up your system for this invocation of the BACKUP SYSTEM utility to create a system-level backup that's copy of the database copy pool is an incremental FlashCopy, then analyze the following:

- The job output from the last couple of invocations of the BACKUP SYSTEM utility. Look for message DSNU1639I.
- The output from the DFSMShsm LIST COPYPOOL command for the database copy pool.
- The output from the print log map utility (DSNJU004) for the system-level backup information. Look for system-level backups with TYPE=I.

If you specified END FCINCREMENTAL, then remove these keywords until the next backup version of the database copy pool to be taken is the version associated with the persistent incremental FlashCopy relationship. END FCINCREMENTAL can only be specified to take a last incremental FlashCopy and withdraw the persistent incremental FlashCopy relationship when the backup version to be taken is the incremental FlashCopy relationship.

Severity: 8 (error)

Related reference:

�️ BACKUP SYSTEM (DB2 Utilities)

▷ DSNJU004 (print log map) (DB2 Utilities)

Related information:

"DSNU1639I" on page 923

DSNU1631I *csect-name* **-** *utility-name* **SYSTEM UTILITY FAILED BECAUSE THE CALL TO DFSMSHSM FAILED WITH RC = X'** *return-code***', REASON = X'***reason-code***'. SEE THE HSM ACTIVITY LOG FOR HSM MESSAGES INDICATING THE CAUSE OF ERROR**

Explanation: The call to DFSMShsm to perform the backup or the restore of the copy pools failed with the return code and reason code indicated in the message.

csect-name
> The name of the control section that issued the message.

utility-name
> The name of the system utility that failed.

return-code
> The return code issued by DFSMShsm. This value indicates which message to look for in the HSM job log:

> **X'6'** Look for message ARC1806E.

> **X'66'** Look for message ARC1866I.

> **X'8' or higher**
> > Look for message ARC1808E.

reason-code
> A numeric value that indicates the reason that the utility failed.

System action: The utility is terminated.

System programmer response: Look up the return code and reason code in the DFSMShsm documentation, and take the appropriate action. Examine the HSM job log. In some cases, the HSM job log contains messages that indicate the cause of the error.

Problem determination: The HSM log does not have information for every error. If the HSM log does not have information for this error, use the reason code and return code to diagnose the problem.

Severity: 8 (error)

Related reference:

➟ BACKUP SYSTEM (DB2 Utilities)

➟ RESTORE SYSTEM (DB2 Utilities)

➟ ARC messages (MVS System Messages)

➟ ARC return codes and reason codes (MVS System Messages)

DSNU1632I *csect-name* **- DISPLAY REQUEST WITH MEMBER SPECIFICATION WAS MADE BUT THE SYSTEM IS IN SYSTEM RECOVER-PENDING STATE. THE MEMBER SPECIFICATION IS IGNORED AND ONLY THOSE UTILITIES RUNNING ON THE SYSTEM FROM WHICH THE DISPLAY COMMAND WAS ISSUED WILL BE DISPLAYED**

Explanation: When a system is in the system RECOVER-PENDING state it isn't possible to determine which utilities are executing on members in a data sharing group other than the member on which the DISPLAY command is running. The MEMBER specification is ignored and only those utilities executing on the system on which the command is running are displayed.

System action: Display any utilities running on the system on which the command is running.

Severity: 0 (informational)

DSNU1633I *csect-name* **- BACKUP SYSTEM UTILITY FAILED BECAUSE UNABLE TO WRITE BSDS RECORD DUE TO VSAM ERROR: RC = X'***return code***', RSN = X'***reason code***

Explanation: The attempt to update or insert the BACKUP SYSTEM history BSDS record failed due to an error during VSAM processing. The VSAM return and reason codes are indicated in the message.

System action: The utility is terminated.

System programmer response: The return and reason code in the message will identify the VSAM error that occurred when attempting to write out the BSDS record. Correct the problem and resubmit the job.

Severity: 8 (error)

DSNU1634I *csect-name* **- RESTORE SYSTEM UTILITY FAILED BECAUSE THE SYSTEM IS IN TRACKER MODE AND LOGONLY WAS NOT SPECIFIED**

Explanation: The RESTORE SYSTEM utility cannot be run on a tracker system unless the LOGONLY keyword is specified.

System programmer response: Change the RESTORE SYSTEM utility control statements to include the LOGONLY parameter.

Severity: 8 (error)

DSNU1635I *csect-name* **THE RBA RANGE FOR THE LAST CHECKPOINT ISSUED DURING THE LOGAPPLY PHASE OF THE RESTORE SYSTEM UTILITY IS START_RBA = X'***yyyyyyyyyyyy***' END_RBA = X'***zzzzzzzzzzzz***' FOR MEMBER** *mmmm*

Explanation: This message displays the RBA range of the last checkpoint issued by the RESTORE SYSTEM utility. You can use the RBA range as input to DSN1LOGP to print log records that show which objects were affected by the RESTORE SYSTEM utility. The subsystem name of the member is issued only for systems in data sharing mode.

System programmer response: If you suspect a problem, run the DSN1LOGP stand-alone utility to verify which objects were affected by the RESTORE SYSTEM utility.

Severity: 0 (informational)

Related concepts:

➟ How RBA and LRSN values are displayed (DB2 Administration Guide)

DSNU1636I   RESTORE SYSTEM UTILITY FAILED BECAUSE THE SPECIFIED DUMPCLASS IS NOT KNOWN TO DFSMSHSM

Explanation: The RESTORE SYSTEM Utility failed to restore the DB2 database copy pool because the DFSMShsm dump pclass specified does not exist for the system-level backup. Specify the DFSMShsm dump class via the RESTORE SYSTEM DUMPCLASS option or via installation panel DSNTIP6 by specifying RESTORE/RECOVER From dump => YES and DUMP CLASS NAME => hsm-dumpclass-nm.

System action: The RESTORE SYSTEM Utility terminates processing.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Run the Print Log Map Utility (DSNJU004) to print the system-level backup information. If data sharing, ensure that the system-level backup information from all members is gathered. Locate the systemlevel backup information in the DSNJU004 output and compare it with the output from the LIST COPYPOOL command. The LIST COPYPOOL command output shows the DFSMShsm dump classes associated with each system-level backup.

Severity: 8 (error)

DSNU1637I   RESTORE SYSTEM UTILITY FAILED BECAUSE NO FLASHCOPY IS AVAILABLE

Explanation: The RESTORE SYSTEM Utility could not restore the DB2 database copy pool because the DASD version of the system-level backup is unavailable.

System action: The RESTORE SYSTEM Utility terminates processing.

System programmer response: Issue the DFSMShsm LIST COPYPOOL command with the ALLVOLS option for the DB2 database copy pool. Run the Print Log Map Utility (DSNJU004) to print the system-level backup information. If data sharing, ensure that the system-level backup information from all members is gathered. Locate the system-level backup information in the DSNJU004 output and compare it with the output from the LIST COPYPOOL command.

If the system-level backup has been dumped to tape, then you can use either of these options to direct the utility to use the copy of the system-level backup on tape:

• The RESTORE SYSTEM Utility FROMDUMP option
• On installlationation panel DSNTIP6, specify RESTORE/RECOVER From dump => YES

Severity: 8 (error)

DSNU1638I   RESTORE SYSTEM UTILITY FAILED BECAUSE NO FLASHCOPY IS AVAILABLE

Explanation: The DFSMShsm LIST COPYPOOL command issued by the RESTORE SYSTEM Utility has failed. The information from the LIST COPYPOOL command is required to correctly restore system-level backups that have been dumped to tape.

System action: The RESTORE SYSTEM Utility terminates processing.

System programmer response: Refer to the DFSMShsm manuals using the return code and reason code to determine why the LIST COPYPOOL command failed and correct the problem. You can execute the RESTORE SYSTEM Utility again after correcting the problem.

Severity: 8 (error)

DSNU1639I   *csect-name* THE SYSTEM LEVEL BACKUP TAKEN IS AN INCREMENTAL FLASHCOPY OF THE DATABASE COPY POOL

Explanation: This message indicates that the system level backup that was created by the BACKUP SYSTEM utility is an incremental FlashCopy of the database copy pool. This indicates that, in most cases, the background copy will complete faster than the background copy for a full copy.

System programmer response: You might have to take action if an incremental copy was not expected. Run a LIST COPYPOOL command and run the print log map (DSNJU004) utility.

Severity: 0 (informational)

DSNU1640I *csect-name* **- Z/OS DFSORT LOAD MODULE NOT LOADED.**

Explanation: The z/OS DFSORT load module was not located in the system search order.

System action: Utility processing is terminated.

Operator response: Notify the system programmer.

System programmer response: Verify that the system is configured to allow DB2 to use DFSORT load modules. If DFSORT is not installed as the primaryz/OS sort product, make the DFSORT SORTLPA and SICELINK libraries available for use by DB2 utilities.

Severity: 8 (error)

Related reference:

➡ SORT LIBRARY field (DB2 Installation and Migration)

DSNU1645I **VALUE MISSING FROM THE SYSVALUEDDN DATA SET ASSOCIATED WITH** *vcat-specification*

Explanation: A row in the SYSVALUEDDN data set does not contain a complete specification of the integrated catalog facility (ICF) alias names (VCAT). Both the previous and current alias names must be specified.

vcat-specification
The alias names specified in a row of the SYSVALUEDDN data set.

The specification is a pair of integrated catalog facility (ICF) aliases separated with a comma.

System action: The utility terminates.

System programmer response: Correct the row in the SYSVALUEDDN data set indicated by *vcat-specification* and resubmit the job.

Severity: 8 (error)

Related reference:

➡ Data sets that RESTORE SYSTEM uses (DB2 Utilities)

DSNU1646I **AN INVALID CHARACTER** *char* **WAS SPECIFIED ON THE LINE CONTAINING** *vcat-specification* **IN THE SYSVALUEDDN DATA SET**

Explanation: A row in the SYSVALUEDDN data set contains an invalid character.

char The invalid character in the SYSVALUEDDN data set.

vcat-specification
The record in the SYSVALUEDDN data set that specifies the previous and current integrated catalog facility (ICF) alias names.

System action: The utility terminates.

System programmer response: Correct the character in the row of the SYSVALUEDDN data set indicated by *vcat-specification* and resubmit the job.

Severity: 8 (error)

Related reference:

➡ Data sets that RESTORE SYSTEM uses (DB2 Utilities)

DSNU1647I **A MATCH FOR** *vcat-name* **COULD NOT BE FOUND IN SYSVALUEDDN WHILE PROCESSING LOG RECORD FOR PAGE SET** *object-qualifier.object-name* **WITH DBID** *database-id* **AND PSID** *pageset-id*

Explanation: An integrated catalog facility (ICF) alias name in a log record does not match any found in the SYSVALUEDDN data set.

vcat-name
The name of an integrated catalog facility (ICF) alias specified in SYSVALUEDDN.

| *object-qualifier.object-name*
| The qualified name of the object referenced in the log record.

| *database-id*
| The DBID of the object referenced in the log record.

| *pageset-id*
| The PSID of the object referenced in the log record.

| **System action:** The utility abends with completion code X'04E' and reason code X'00E40106'.

| **System programmer response:** Add a row in the SYSVALUEDDN data set to specify the mapping for *vcat-name* and
| resubmit the job.

| **Related reference:**

| ➡ Data sets that RESTORE SYSTEM uses (DB2 Utilities)

| **DSNU1648I   RESTORE SYSTEM UTILITY FAILED. THE AVAILABLE BACKUP IS NOT USABLE BEYOND THE**
| **SIX BYTE LOGGING LIMIT AFTER THE BSDS HAS BEEN CONVERTED.**

| **Explanation:** The RESTORE SYSTEM utility was unable to use the available backup because it was taken before the
| BSDS was converted and the 6-byte logging limit was reached on this system.

| **System action:** Processing terminates.

| **User response:** This system can be restored only from this backup with a subsystem parameter setting of
| RESTORE_RECOVER_FROMDUMP = YES, which is typically used only at disaster recovery sites.

| **Severity:** 8 (error)

| **Related reference:**

| ➡ RESTORE SYSTEM (DB2 Utilities)

DSNU1702I   *csect-name* - LOB PAGE TYPE MISMATCH. ROWID *rowid* VERSION *version* PAGE *page-number*
EXPECTED TYPE *bit-value* ACTUAL TYPE *bit-value*

Explanation: A LOB page with page number *page-number* has an unexpected page type specification. In the expected
and actual page types, *bit-value* is a 3-bit value. The expected values can be B'100' for a LOB map page or B'110' for a
LOB data page. The LOB page in error is part of the LOB with the ROWID *rowid* and version *version*.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to
CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1703I   *csect-name* - LOB PAGE TYPE MISMATCH. PAGE *page-number* EXPECTED TYPE *bit-value* ACTUAL
TYPE *bit-value*

Explanation: A LOB page with page number *page-number* has an unexpected page type specification. In the expected
and actual page types, *bit-value* is a 3-bit value. The expected values can be B'001' for a LOB high-level space map, or
B'011' for a LOB low-level space map.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to
CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1704I *csect-name* **- UNEXPECTED PAGE TYPE. ROWID** *rowid* **VERSION** *version* **PAGE** *page-number*
ACTUAL PAGE TYPE *bit-value*

Explanation: A LOB page with page number *page-number* has an unexpected page type specification. In the actual page type, *bit-value* is a 3-bit value. At the time of processing, the CHECK LOB utility expected either B'100' for a LOB map page or B'110' for a LOB data page. The LOB page in error is part of the LOB with the ROWID *rowid* and version *version*.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1706I *csect-name* **- PAGE** *page-number* **MISSING IN LOB MAP. ROWID** *rowid* **VERSION** *version*

Explanation: A low-level space map, indicates that the LOB page with page number *page-number* is allocated to the LOB with the ROWID *rowid* and version *version*. The LOB map for the specified LOB does not include the specified page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1708I *csect-name* **- LOB MAP PAGE** *page-number* **MISSING. ROWID** *rowid* **VERSION** *version*

Explanation: A low-level space map indicates that the LOB map page with page number *page-number* is allocated to the LOB with the ROWID *rowid* and version *version*. The CHECK LOB utility was unable to verify the existence of this page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1710I *csect-name* **- LOB PAGE ALLOCATED TO A DIFFERENT LOB. PAGE** *page-number* **IS IN THE LOB**
MAP OF ROWID *rowid1* **VERSION** *version1*. **LOW-LEVEL SPACE MAP INDICATES THE PAGE**
BELONGS TO ROWID *rowid2* **VERSION** *version2*

Explanation: The LOB page with page number *page-number*, is in the LOB map of the LOB with ROWID *rowid1* and version *version1*. A low-level space map indicates that the page belongs to different LOB with ROWID *rowid2* and version *version2*. The page can be a data page or a LOB map page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1712I *csect-name* **- LOB DATA LENGTH INCORRECT. ROWID** *rowid* **VERSION** *version* **EXPECTED LENGTH** *hex-value1* **ACTUAL LENGTH** *hex-value2*

Explanation: The actual sum of the data lengths in the LOB data pages, *hex-value2*, does not match the expected value specified in the LOB map page, *hex-value1*. The LOB in error is identified by ROWID *rowid* and version *version*.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1714I *csect-name* **- LOB MAP PAGE COUNT INCORRECT. ROWID** *rowid* **VERSION** *version* **EXPECTED COUNT** *hex-value1* **ACTUAL COUNT** *hex-value2*

Explanation: The actual count of data pages, *hex-value1*, does not match the expected number of pages, *hex-value2*, that is specified in the LOB map page. The LOB in error is identified by ROWID *rowid* and version *version*.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1716I *csect-name* **- NEWER VERSION EXISTS FOR LOB. ROWID** *rowid* **VERSION** *version*

Explanation: The LOB table space contains a newer LOB version for the LOB that is identified by ROWID *rowid* and version *version*. The newer LOB has the same ROWID, *rowid*, but a higher VERSION number. The LOB that is identified by the message is reported as an orphan LOB by the CHECK DATA utility. Other than occupying space in the LOB table space, the presence of orphan LOBs does not hinder LOB processing.

System action: CHECK LOB utility processing continues. Utility returns completion code=0.

User response: The LOB can be deleted using REPAIR LOCATE ROWID VERSION DELETE.

Severity: 0

DSNU1720I *csect-name* **- PAGE** *page-number* **IN LOB MAP IS MISSING A LOW-LEVEL SPACE MAP ENTRY. ROWID** *rowid* **VERSION** *version*

Explanation: The LOB page that is specified by *page-number* is in the LOB map of the LOB with the ROWID *rowid* and version *version*, but is not allocated in any low-level space map entry.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1722I *csect-name* **- LOB PAGE CHECK BYTE MISMATCH. ROWID** *rowid* **VERSION** *version* **PAGE** *page-number* **EXPECTED BYTE** *hex-value1* **ACTUAL BYTE** *hex-value2*

Explanation: The actual check byte from the specified LOB data page, *hex-value2*, does not match the expected check byte, *hex-value1*, from the LOB map page of LOB with the ROWID *rowid* and version *version*.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1724I *csect-name* **-** *page-type* **PAGE NUMBER MISMATCH. ROWID** *rowid* **VERSION** *version* **EXPECTED PAGE NUMBER** *expected-number* **ACTUAL PAGE NUMBER** *actual-number*

Explanation: A LOB page had a page number, *actual-number*, other than the expected number, *expected-number*. The page belongs to the LOB with the ROWID *rowid* and version *version*. *page-type* specifies the type of page that was expected: 'LOB MAP' for an expected LOB map, or 'LOB DATA/MAP' for a page that could be a LOB map or a data page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1725I *csect-name* **-** *page-type* **PAGE NUMBER MISMATCH. EXPECTED PAGE NUMBER** *expected-number* **ACTUAL PAGE NUMBER** *actual-number*

Explanation: A LOB page had a page number, *actual-number*, other than the expected number, *expected-number*. *page-type* specifies the type of page that is expected: 'HIGH-LEVEL SPACE MAP' or 'LOW-LEVEL SPACE MAP'.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1726I *csect-name* **ZERO PAGE ENCOUNTERED** *page-type* **PAGE** *page-number* **ROWID** *rowid* **VERSION** *version*

Explanation: Page numbered *page-number* is a zero page. The page belongs to the LOB with the ROWID *rowid* and version *version*. *page-type* specifies the type of page that is expected: 'LOB MAP' for a LOB map, or 'LOB DATA/MAP' for a LOB map or a data page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1727I *csect-name* **ZERO PAGE ENCOUNTERED** *page-type* **PAGE** *page-number*

Explanation: Page numbered *page-number* is a zero page. While only the page header information was found to be zero, generally, the entire page will be zero. *page-type* specifies the type of page expected: 'HIGH-LEVEL SPACE MAP' or 'LOW-LEVEL SPACE MAP'.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1728I   *csect-name* **-** *page-type* **PAGE FROM LOB MAP IS MISSING. ROWID** *rowid* **VERSION** *version*

Explanation: A page is in the LOB map of the LOB with the ROWID *rowid* and version *version*, but is not present in the LOB table space. *page-type* specifies the type of missing page, 'LOB MAP' for an addition LOB map or 'LOB DATA/MAP' for a page that could be a LOB map or a data page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1730I   *csect-name* **- LOB MAP PAGE** *page-number* **CONTAINS ROWID** *rowid1* **VERSION** *version1* **BUT LOW-LEVEL SPACE MAP CONTAINS ROWID** *rowid2* **VERSION** *version2*

Explanation: The LOB map page of a LOB contains the ROWID *rowid1* and version *version1* but the low-level space map indicates that the LOB map page contains the ROWID *rowid2* and version *version2*. The ROWID and version must match.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1732I   *csect-name* **- PAGE** *page-number* **IN LOB MAP HAS AN UNALLOCATED OR DEALLOCATED LOW-LEVEL SPACE MAP ENTRY. ROWID** *rowid* **VERSION** *version*

Explanation: The LOB map of the LOB with the ROWID *rowid* and version *version*, indicates that page *page-number* is part of the LOB. The low-level space map entry for the page indicates that the page is unallocated or deallocated.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1734I   *csect-name* **- UNABLE TO READ** *page-type* **PAGE** *page-number***. ERROR CODE** *hex-value1* **REASON CODE** *hex-value2* **ROWID** *rowid* **VERSION** *version*

Explanation: An attempt to read page number *page-number* failed with error code *hex-value1* and reason code of *hex-value2*. The page belongs to the LOB with the ROWID *rowid* and version *version*. *page-type* specifies the type of page that is expected: 'LOB MAP' for an expected LOB map, or 'LOB DATA/MAP' for a page that could be a LOB map or a data page.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1735I *csect-name* **- UNABLE TO READ** *page-type* **PAGE** *page-number*. **ERROR CODE** *hex-value* **REASON CODE** *hex-value*

Explanation: An attempt to read page number *page-number* failed with error code *hex-value1* and reason code of *hex-value2*. *page-type* specifies the type of page that is expected: 'HIGH-LEVEL SPACE MAP' or 'LOW-LEVEL SPACE MAP'.

System action: If SHRLEVEL REFERENCE was specified or defaulted, then the LOB table space status is set to CHECK Pending (CHKP). If SHRLEVEL CHANGE was specified, then the table space status remains unchanged.

System programmer response: Notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4

DSNU1801I *csect-name* **SYSTEM PARAMETER** *parameter-name* **HAS BEEN MODIFIED TO** *parameter-value*. **RESTART IS NOT ALLOWED.**

Explanation: The utility could not be restarted because a DB2 subsystem parameter was modified after the first invocation of this utility.

csect-name
>   The name of the control section that issued the message.

parameter-name
>   The name of the parameter.

parameter-value
>   The parameter value that was changed.

System action: The utility abends with ABEND04E RC00E40018 (error detected during a restarted utility).

System programmer response: If necessary, change the subsystem parameter value back to the value that was used when the utility was first run.

User response: If necessary, issue the TERM UTILITY command to terminate the utility. If you want to restart the utility, determine the current value of the indicated subsystem parameter and take the appropriate action:

• If the parameter value has already been corrected, restart the utility.

• If the parameter value is still incorrect, notify the system programmer, and restart the utility after the value has been changed.

Related reference:

➡ -TERM UTILITY (DB2) (DB2 Commands)

Related information:

➡ 00E40018 (DB2 Codes)

DSNU2700I *csect-name* **UNLOAD PHASE STATISTICS: NUMBER OF RECORDS=***total-records* **(TOTAL). NUMBER OF RECORDS=***overflow-records* **(OVERFLOW)**

Explanation: The REBUILD INDEX utility is in the UNLOAD phase.

csect-name
>   The name of the control section that issued the message.

total-records
>   The total number of records, which indicates the number of rows in the entire table space.

overflow-records
>   The number of rows that are in the hash overflow area, and which require an index entry in the hash overflow index.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

Related reference:

REBUILD INDEX (DB2 Utilities)

DSNU2801I *csect-name* **- RTS VALUES ARE NOT AVAILABLE FOR ESTIMATING HASH SPACE VALUE FOR** *table-name*

Explanation: The REORG TABLESPACE utility was specified with the AUTOESTSPACE(YES) option, but real-time statistics were not available to estimate the size of the hash space. The user-specified hash space value was used to build the hash space for the table.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

System action: Utilities continue to process.

User response: Run the REORG TABLESPACE utility with the AUTOESTSPACE(YES) option when real-time statistics values are available.

Severity: 4 (warning)

Related reference:

REORG TABLESPACE (DB2 Utilities)

DSNU2802I *csect-name* **THE HASH SPACE HAS BEEN PREALLOCATED AT** *x* **FOR TABLE** *table-name*

Explanation: The hash space has been created for the indicated table.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

The hash space value can either be specified by the user or be determined based on real-time statistics, through the REORG TABLESPACE AUTOESTSPACE(YES) option.

System action: Utilities continue to process.

User response: No action is required.

Severity: 0

Related reference:

REORG TABLESPACE (DB2 Utilities)

DSNU2803I *csect-name* **PARTITION LEVEL REORG CANNOT BE PERFORMED AFTER ALTER ADD ORGANIZATION OR ALTER DROP ORGANIZATION**

Explanation: A partition-level reorganization cannot be performed on a table space after either an ALTER ADD ORGANIZATION or ALTER DROP ORGANIZATION action.

csect-name
> The name of the control section that issued the message.

System action: Utility execution is terminated.

User response: Correct hte REORG utility statement to remove the PART specification.

Severity: 8 (error)

Related reference:

↪ REORG TABLESPACE (DB2 Utilities)

DSNU2900I *csect-name* **MAPPING TABLE IS SPECIFIED WITH A NON-EXPANDED LRSN COLUMN**

Explanation: A REORG utility statement has detected that the user provided a mapping table with a CHAR(6) LRSN column definition.

csect-name
> The name of the control section that issued the message.

System action: Before reaching Version 11 new-function mode, REORG processing continues with the user input mapping table. After reaching Version 11 new-function mode, REORG ignores the user-specified mapping table and implicitly creates a mapping table with a CHAR(10) LRSN column definition.

Operator response: No action is required.

Severity: 4 (warning)

DSNU2901I *csect-name* **MAPPING TABLE** *table-name* **and MAPPING INDEX** *index-name* **CREATED IN DATABASE**
　　　　　　 database-name.*tablespace-name*

Explanation: The REORG utility statement implicitly created the reported mapping table and mapping index in the specified database name and table space name.

csect-name
> The name of the control section that issued the message.

table-name
> The name of the table.

index-name
> The name of the index.

database-name
> The name of the database.

tablespace-name
> The name of the table space.

System action: Utility processing continues.

Operator response: No action is required.

Severity: 0

DSNU2902I *csect-name* **MAPPING DATABASE** *database-name* **IS INVALID**

Explanation: The REORG utility statement has detected that the database that is specified for the REORG TABLESPACE utility MAPPINGDATABASE keyword or the REORG_MAPPING_DATABASE subsystem parameter does not exist or cannot be used to implicitly create a mapping table.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

System action: Utility processing terminates.

Administrator response: Specify a valid database name for the REORG TABLESPACE utility MAPPINGDATABASE keyword or specify a valid value for the REORG_MAPPING_DATABASE subsystem parameter.

Severity: 8 (error)

DSNU2903I *csect-name* **- PARTITION LEVEL INLINE COPY DATASETS WILL BE ALLOCATED.**

Explanation: REORG is reorganizing a partitioned table space and detects a valid TEMPLATE is specified for the creation of part-level inline copy data sets.

System action: Processing continues.

Severity: 0

DSNU2904I *csect-name* **- DATA RECORDS WILL BE UNLOADED VIA** *unload-method***.**

Explanation: The REORG utility determines the data records that are unloaded via one of the following *unload-methods*:

CLUSTERING INDEX
> Data records are unloaded by the clustering index order.

TABLE SCAN
> Data records are unloaded by performing a table scan.

TABLE SPACE SCAN
> Data records are unloaded by performing a table space scan.

System action: Processing continues.

Severity: 0

DSNU2905I *csect-name* **- EMPTY DATA PARTITION** *lowest-partition* **TO** *highest-partition* **DELETED.**

Explanation: The REORG utility deleted the empty partitions beginning with *lowest-partition* number to the *highest-partition* number. Auxiliary tables and auxiliary indexes that are associated with these deleted base partitions are also dropped, as well as implicitly created LOB table spaces.

System action: Processing continues.

Severity: 0

DSNU2906I *csect-name* **- REBALANCE PARTITION SUMMARY ON** *database-name*.*tablespace-name*

Explanation: The header message for the partition summary section of the REORG TABLESPACE REBALANCE output on database *database-name* table space *tablespace-name*. This section lists the affected data partitions as a result of data rebalancing:

LPART
> The logical partition number.

PPART
> The physical partition number.

ROWCOUNT
> The number of records in this partition.

LIMITKEY
> The limit key value for this partition.

csect-name
> The name of the control section that issued the message.

database-name
> The name of the database.

tablespace-name
> The name of the table space.

System action: Utility processing continues.

Severity: 0

| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

| **DSNU2907I** *csect-name* **- MAPPING** *object-type* *operation-type* **FAILED**

| **Explanation:** The REORG utility encountered an SQL error during implicit creation or deletion of the specified
| mapping object.

| *csect-name*
| The name of the control section that issued the message.

| *object-type*
| The type of object: table space, table, or index.

| *operation-type*
| The type of operation: creation or deletion.

| **System action:** If *operation-type* is creation, utility processing terminates.

| If *operation-type* is deletion, utility processing completes with a warning.

| **User response:** Determine the cause of the SQL error by using the information from the accompanying DSNU11xx
| message.

| **Severity:** If *operation-type* is creation:

| 8 (error)

| If *operation-type* is deletion:

| 4 (warning)

| **DSNU2916I** *csect-name* **- PENDING ALTER LIMIT KEY VALUES ARE BEING MATERIALIZED**

| **Explanation:** The REORG TABLESPACE utility is reorganizing a partitioned table space and, as part of that
| reorganization, is materializing the pending alter limit key values on the affected data partitions.

| *csect-name*
| The name of the control section that issued the message.

| **System action:** Utility processing continues.

| **User response:** No action is required.

| **Severity:** 0
| **Related tasks**:

| ➡ Redistributing data in partitioned table spaces (DB2 Administration Guide)

| ➡ Materializing pending definition changes (DB2 Administration Guide)
| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

| **DSNU2917I** *csect-name* **-** *keyword* **IS NOT ALLOWED ON TABLE SPACE WITH PENDING ALTER LIMIT KEY**
| **CHANGES**

| **Explanation:** The REORG TABLESPACE utility is attempting to run on a table space or partitions with pending alter
| limit key changes. However, the specified keyword is not allowed because of the pending alter limit key changes.

| *csect-name*
| The name of the control section that issued the message.

| *keyword* The keyword in the REORG TABLESPACE control statement that is not allowed in this situation.

| **System action:** Utility processing terminates

| **User response:** Take one of the following actions:

| • Remove *keyword* from the REORG TABLESPACE statement and rerun the utility.

| • Drop the pending alters for the table space and then rerun REORG TABLESPACE.

| **Severity:** 8 (error)

| **Related reference**:

| ➡ Syntax and options of the REORG TABLESPACE control statement (DB2 Utilities)

| ➡ ALTER TABLESPACE (DB2 SQL)

| **DSNU2921I** *csect-name* **- OPTION SHRLEVEL REFERENCE IS REQUIRED ON TABLESPACE** *tablespace-name* **TO COMPLETE THE POINT-IN-TIME RECOVERY PROCESS**

| **Explanation:** REORG with SHRLEVEL NONE was specified when REORG was run on the specified table space to complete a point-in-time recovery process. Only SHRLEVEL REFERENCE is valid in this case.

| *tablespace-name*
| The name of the table space.

| **System action:** Utility processing is not initiated.

| **User response:** Run REORG with SHRLEVEL REFERENCE to finalize the point-in-time recovery process.

| **Severity:** 8 (error)

| **Related reference**:

| ➡ REORG TABLESPACE (DB2 Utilities)

DSNU3030I JOB *jobname* **CANCEL ACCEPTED, WAITING FOR DB2 THREADS TO TERMINATE**

Explanation: A CANCEL command was issued for a job while a DB2 utility was active. The job will terminate after all of its DB2 threads have terminated.

System action: All threads created by the utility will perform the rollback and deallocation processing that is required. When this is complete the job will terminate.

System programmer response: Rollback processing might take a significant amount of time, especially for a LOAD LOG YES utility. Under extreme conditions a FORCE command can be used to immediately terminate the job, but this is not recommended because it can cause the thread deallocation to abend.

Severity: 0

DSNU3330I *csect-name* **- The** *utility-name* **UTILITY IS PART OF A SEPARATELY LICENSED FEATURE WHICH WAS NOT FOUND ON THIS SYSTEM.**

Explanation: DB2 could not find the specified utility in the load library concatenation that was specified in the submitted JCL. The specified utility is part of an optional feature of DB2.

csect-name
The name of the control section that issued the message.

utility-name
The name of the utility.

System action: Utility processing does not initiate.

User response: Obtain and install the optional utility feature (FMID JDBBB1K).

Severity: 8 (error)

Related concepts:

➡ DB2 utilities packaging (DB2 Utilities)

DSNU3333I LOAD MODULE *module-name* **DID NOT LOAD CORRECTLY. THE** *utility-name* **UTILITY FUNCTION IS RESTRICTED**

Explanation: The identified load module was found in the product library, but could not be loaded into memory correctly.

System action: Utility processing continues but the identified utility has restricted function. It may only be executed against objects in the DB2 catalog, the DB2 directory or the DB2 sample program.

System programmer response: Make sure the identified load module is reenterable and reusable and that it is being loaded from an APF authorized library. This error is normally caused by incorrectly copying a load module or its alias. Use only SMP/E APPLY or the IEWL program to create load modules and their aliases. If SMP/E has produced a load module that is not reenterable and reusable notify IBM Software Support.

User response: Notify the system programmer.

Severity: 4 (warning)

DSNU3334I REGION SIZE IS TOO SMALL. UTILITY CANNOT CONTINUE. INCREASE REGION BY *additional-memory***K.**

Explanation: The REGION size that is specified on the utility job or job step allocates too little virtual memory for the utility to run.

additional-memory
    The minimum additional amount of memory that the utility estimates it requires to run.

System action: The utility terminates.

System programmer response: If the region size is limited by the IEFUSI exit routine, consider modifying the exit routine to allow a larger region.

User response: Increase the region size by at least the amount indicated by additional-memory in the message, and resubmit the job. For example, if the region was set to 2048 KB (REGION=2048K), and the message indicates "INCREASE REGION BY 240K", change the region size to at least REGION=2288K. If possible, allow an additional buffer amount: REGION=3M or REGION=4M.

Severity: 8 (error)

DSNU3340I UTILITY PERFORMS DYNAMIC ALLOCATION OF SORT DISK SPACE

Explanation: The utilities that invoke a sort will perform predictable sort disk space dynamic allocation.

System action: Processing of the phase continues.

Programmer response: No action is required.

Severity: 0 (informational)

DSNU3341I CANNOT ALLOCATE SORT DATA SETS WITHIN 255 LIMIT

Explanation: When the utility performs dynamic allocation of sort data sets, the total number of data sets per phase must be less than 255 data sets.

System action: Utility processing is terminated.

System programmer response: Provide more disk space for sort work data sets.

Programmer response: No action is required.

Severity: 8 (error)

DSNU3342I NUMBER OF OPTIMAL SORT TASKS = *nnn***, NUMBER OF ACTIVE SORT TASKS =** *nnn*

Explanation: The first number indicates the optimal number of sort tasks for the utility execution. The second number show the maximum number of sort tasks possible for the utility execution.

System action: Processing of the phase continues.

Programmer response: No action is required.

Severity: 0 (informational)

DSNU3343I *csect-name* **- REAL-TIME STATISTICS INFORMATION MISSING FOR** *object-type object-qualifier.object-name partition-information*

Explanation: The utility could not read necessary information from real-time statistics tables for the specified index or table space.

csect-name
> The name of the control section that issued the message.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

partition-information
> If the object is partitioned, the partition number for which information is missing.

This message can be issued during REBUILD INDEX on catalog and directory indexes when DB2 has been started with the ACCESS (MAINT) option, in which case the real-time statistics information is not queried, or when the real-time statistics objects are not available.

System action: The utility continues, but will use estimates based on RUNSTATS information for this object.

System programmer response: Make sure that real-time statistics are enabled on the system. Run the REORG INDEX or REBUILD INDEX utility on the specified object to set the base values in the real-time statistics tables.

Severity: 0

Related tasks:

↪ Setting up your system for real-time statistics (DB2 Performance)

Related reference:

↪ REORG INDEX (DB2 Utilities)

↪ REBUILD INDEX (DB2 Utilities)

DSNU3344I **REAL-TIME STATISTICS INFORMATION MISSING FOR** *object-type object-qualifier.object-name part-number*. **PROCESSING CANNOT PROCEED.**

Explanation: RUNSTATS could not read the necessary information from the real-time statistics tables for the object.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

part-number
> The value that identifies the partition or data set.

System action: RUNSTATS terminates.

System programmer response: Run the REORG INDEX or REBUILD INDEX utility on the specified object to set the base values in the real-time statistics tables. Then run RUNSTATS with TABLESAMPLE SYSTEM *n* again.

Severity: 8 (error)

Related reference:

↪ REORG INDEX (DB2 Utilities)

↪ REBUILD INDEX (DB2 Utilities)

DSNU3345I *csect-name* **MAXIMUM** *parallelism-type* **PARALLELISM IS** *number-of-tasks* **BASED ON** *condition*

Explanation: This message indicates the maximum amount of parallelism that can be achieved when the utility runs, if no constraints are applied.

Parallelism for a utility is typically implemented by processing partitions or indexes in parallel. The maximum degree of parallelism when a utility runs is usually determined by either the number of partitions or the number of indexes that are being processed. This message can serve as a baseline to see if the optimal degree of parallelism is being achieved. If constraints limit parallelism, parallelism can be increased to this value if the constraints are removed.

csect-name
> The name of the control section that issued the message.

parallelism-type
> The type of parallelism to which the limit applies. Possible values are:

> **INDEX** Index parallelism

> **PARTITION**
>> Partition parallelism

number-of-tasks
> The maximum number of parallel tasks that can be started for this utility.

condition
> The condition that controls the maximum amount of parallelism. Possible values are:

> **NUMBER OF INDEXES**
>> The number of indexes that the utility is processing.

> **NUMBER OF INDEXES AND STATISTICS**
>> The number of indexes and statistics that the utility is processing.

> **NUMBER OF INPUT FILES**
>> The number of input files that the utility is processing.

> **NUMBER OF OBJECTS**
>> The number of objects that the utility is processing.

> **NUMBER OF PARTITIONS**
>> The number of partitions that the utility is processing.

> **NUMBER OF PARTITIONS AND INDEXES**
>> The number of partitions and indexes that the utility is processing.

System action: Processing continues.

User response: No action is required.

DSNU3346I **REAL-TIME STATISTICS INFORMATION MISSING FOR** *object-type object-qualifier.object-name*. **SAMPLING RATE CHANGED TO 100.**

Explanation: RUNSTATS could not read the necessary information from the real-time statistics tables for the object.

object-type
> The type of object.

object-qualifier.object-name
> The name of the object, as a qualified name.

System action: RUNSTATS sets the sampling rate to 100, and continues to run.

System programmer response: If a sampling rate of 100 is appropriate, no action is required. Otherwise, run the REORG TABLESPACE utility on the specified object to set the base values in the real-time statistics tables. Then run RUNSTATS with TABLESAMPLE SYSTEM *n* again.

Severity: 4 (warning)

Related reference:

↪ REORG TABLESPACE (DB2 Utilities)

Chapter 17. DSNV messages

Message identifiers that begin with DSNV communicate information about the agent services manager.

DSNV029E   **LOAD MODULE** *module-name* **DOES NOT HAVE AMODE(31) ATTRIBUTE**

Explanation: During DB2 startup, Agent Services establishes the tasking structure for the control and resource manager address spaces. Three load modules, DSNVEUS3, DSNVEUS4, and DSNVRCT, are loaded and attached as the control programs for each subtask created. All three load modules must have been link-edited with the attributes AMODE(31) and RMODE(ANY). At least one of the modules did not have the 31-bit addressing attribute when it was loaded. The 'module-name' in the error message is the alias name of the first load module whose addressing mode was invalid.

System action: DB2 startup is terminated.

Operator response: Notify the system programmer of the failure.

System programmer response: Verify that all installation and maintenance activities against these modules were executed using SMP, that the JCLIN for SMP includes the correct AMODE and RMODE control statements, and that the Linkage Editor provided with the Data Facility Product (DFP) for the version of MVS being used was invoked by SMP for the link-edits. Correct the procedure or JCL which caused the error and rerun the necessary SMP jobs.

Problem determination: A DB2 dump was requested to a SYS1.DUMP data set.

Load module names and aliases are as follows:

Load module
> **Alias**

DSNVEUS3
> DSNVEU3I

DSNVEUS4
> DSNVEU4I

DSNVRCT
> DSNVRCTI

The MVS service aid AMBLIST provides Linkage Editor module attributes in the summary section of the LISTLOAD function output.

DSNV086E   **DB2 ABNORMAL TERMINATION REASON=***reason-code*

Explanation: The DB2 subsystem is being abended, because an error that cannot be corrected has occurred. This message, which is not automatically deleted from the operator console, is issued during DB2 abnormal termination. The *reason-code* value is the subsystem termination reason code. If DB2 abnormal termination is invoked multiple times, the subsystem termination reason code that accompanies this message is the reason associated with the first invocation.

System action: Abnormal termination processing continues.

Operator response: Notify the system programmer, and restart DB2.

System programmer response: This message is accompanied by one or more dumps. Obtain a copy of SYS1.LOGREC after DB2 completely terminates; also obtain a copy of the SVC dump(s). If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

DSNV401I

- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Related concepts:

➡ DB2 reason codes (DB2 Codes)

| **DSNV401I   DISPLAY THREAD REPORT FOLLOWS -**

| **Explanation:**

| This message marks the beginning of multiple lines of information in response to a DISPLAY THREAD command.
| Some lines in the output have their own message numbers or alphanumeric identifiers to assist with identification.

| The output from the command consists of one or more (but not all) of the following sections:

| - If SCOPE (GROUP) was specified, any thread-specific information is preceded by relevant information about the
| member: Member-specific information (DSNV473I, DSNV747I, DSNV475I, DSNV476I, DSNV503I)

| - The member-specific information, or the DSNV401I message line if SCOPE(GROUP) was not specified, is followed
| by an information block that reports on threads by type:

| – Information about active, inactive, stored procedure, and system threads (DSNV402I, DSNV424I, and DSNV497I)

| – Information about indoubt and postponed threads (DSNV406I and DSNV431I)

| If TYPE(*) was specified, the output contains multiple information blocks.

| - Depending on other command options or system conditions, some additional detail lines are present within the
| report block:

| – V429: Information about stored procedures

| – V436: Information about SQL statements

| – V437: Information about the user and application

| – V440: Information about threads that are managed by an XA transaction manager

| – V441: User accounting information

| – V442: Information about correlation tokens

| – V444: Information about distributed threads (for active, inactive, stored procedure, and system threads)

| – V445: Information about database access threads

| – V447 and V448: Information about conversations

| – V449: Information about IMS and CICS connections and RRS

| – V450: Information about downstream participants that might be indoubt

| – V451: Information about incomplete resolution of indoubt threads

| – V452: Information about heuristic commit decisions

| – V453: Information about heuristic abort decisions

| – V457: Information about resync protocol errors with participants

| – V458: Connection information for indoubt threads

| – V459: Information about syncpoint protocol errors with coordinators

| – V460: Information about syncpoint protocol errors with participants

| – V461: Information about heuristic damage

| – V462: Information about cold starts by participants

| – V463: Information about cold starts by coordinators

| – V464: Information about resync protocol errors with coordinators

| – V465: Information about threads that remain in the prepared state

| – V466: Information about how long a thread was indoubt

| – V467: Information about distributed threads (for indoubt and postponed threads)

| – V471: Information about available agents

| – V480 and V481: Information about the unit of recovery

| – V482: WLM information

| – V485: Information about trusted contexts

– V490: Information about suspended threads
– V492: Information about storage
– V501: Information about commands
– V502: Information about DB2 exit routines
– V504: Information about threads that are terminating
– V505: Information about threads that are waiting
– V507: Information about monitoring tasks
– V520: Information about autonomous procedures
• Information that indicates NOT FOUND situations
• Other information reported in the DISPLAY THREAD output
• End of report markers
• Error messages

Member-specific information (DSNV473I, DSNV747I, DSNV475I, DSNV476I, DSNV503I):

If you specify the SCOPE(GROUP) option, the output includes member-specific information before the thread-specific information for that member.

Depending on the DISPLAY THREAD options and the threads that are found, the output includes one or more of the following lines. Each line is followed by thread-specific information for the specified type of thread on that member:

```
DSNV473I ACTIVE THREADS FOUND FOR MEMBER: text
DSNV474I INDOUBT THREADS FOUND FOR MEMBER: text
DSNV475I INACTIVE THREADS FOUND FOR MEMBER: text
DSNV476I POSTPONED ABORT THREADS FOUND FOR MEMBER: text
DSNV503I SYSTEM THREADS FOUND FOR MEMBER: text
```

text   Member name for which thread information is being displayed.

Information about active, inactive, stored procedure, and system threads (DSNV402I, DSNV424I, and DSNV497I):

The DISPLAY THREAD command reports the same type of information about active, inactive, stored procedure, and system threads.

The TYPE(ACTIVE) and TYPE(PROC) options generate a DSNV402I message.

```
DSNV402I - ACTIVE THREADS -
NAME     ST A   REQ ID              AUTHID    PLAN      ASID TOKEN
text     t  t   num text            text      text      hex  num
DISPLAY ACTIVE REPORT COMPLETE
```

The TYPE(INACTIVE) option generates a DSNV424I message.

```
DSNV424I - INACTIVE THREADS -
NAME     ST A   REQ ID              AUTHID    PLAN      ASID TOKEN
text     t  t   num text            text      text      hex  num
DISPLAY INACTIVE REPORT COMPLETE
```

The TYPE(SYSTEM) option generates a DSNV497I message.

```
DSNV497I - SYSTEM THREADS -
DB2 db2-statestate-phase
NAME     ST A   REQ ID              AUTHID    PLAN      ASID TOKEN
text     t  t   num text            text      text      hex  num
DISPLAY SYSTEM THREAD REPORT COMPLETE
```

Each thread that meets the report criteria is listed on a separate line. Additional information might be provided for individual threads.

DB2 *db2-statestate-phase*

        The state of DB2 as follows:

db2-state
> The state of DB2. Possible values are STARTING, STOPPING, or ACTIVE.

state-phase
> The phase of restart or shutdown, if available. If *db2-state* is ACTIVE, no phase information is
> provided.

NAME *text*
> The connection name that was used to establish the thread.

> If the connection status value is "D", the connection name is either the connection name for the allied
> address space or the DB2 subsystem name.

> For distributed database access threads, this field contains the following information:

requester-value
> For threads that access the server from a DB2 for z/OS requester, this value is the connection name
> of the thread at the requesting location.

SERVER
> For threads that access the server from a non-DB2 for z/OS requester, no connection name is
> identified.

blank   For threads where the originating task is running on the same DB2 subsystem, this value is blank.

> When the connection name value is blank, the expected connection status value is "PT".

ST *t*   A one- or two-letter code that indicates connection status. Possible values are:

AC   A thread is executing in an accelerator server. This status is displayed until accelerator processing
> concludes and returns control to DB2.

AT   A thread that processes an autonomous procedure.

D   The thread is in the process of termination as a result of the termination of the associated allied
> task. If this thread is also the last (or only) DB2 thread for the address space, the associated allied
> task is placed in a wait state.

> For an active thread, this value can indicate possible problems with DB2. If the activity indicator
> value is *, use the information in message DSN3201I to identify and resolve any problems.

DA   The database access thread slot is currently not associated with a remote connection and is available
> to be assigned to a type 2 inactive thread.

DI   The thread is disconnected from an execution unit. No TCB is associated with the DB2 thread.

> This value is possible is only when the connection name is "RRSAF".

N   The thread is in either IDENTIFY or SIGNON status.

> This value is accompanied by a token value of "0".

ND   The thread is in either IDENTIFY or SIGNON status, and the thread is currently not associated with
> any TCB.

PT   A parallel task thread was established (plan allocated).

QD   The thread is queued for termination as a result of the termination of the associated allied task. If
> this thread is also the last (or only) DB2 thread for the address space, the associated allied task is
> placed in a wait state.

> For an active thread, this value can indicate possible problems with DB2. If the activity indicator
> value is *, use the information in message DSN3201I to identify and resolve any problems.

QT   The CREATE THREAD request was queued. The associated allied task is placed in a wait state.

R2   A distributed thread is accessing a remote site on behalf of a request from another location. The
> thread is currently an inactive connection (type 2 inactive thread) and is waiting for an agent to
> become available.

RA   A distributed thread is accessing a remote site on behalf of a request from another location.

| | **RK** | A distributed thread is accessing a remote site on behalf of a request from another location. The thread is performing an operation that invoked Kerberos services. This status is displayed until Kerberos services returns control to DB2. |

RN A distributed thread is accessing a remote site on behalf of a request from another location. The request accesses data at another DB2 location. Therefore, the thread was suspended until DB2 connects to the partner location (establishes DB2 system conversations with the partner).

When another DB2 site is being contacted for the first time using system-directed access, DB2 must establish DB2 system conversations with the partner location. A request is presented to a DB2 DDF service task that establishes a system conversation. The thread is suspended until the request is processed. If the thread remains in this status for an extended period of time, the DB2 service task that establishes the system conversations might be busy processing other requests. The DB2 DISPLAY LOCATION command (DISPLAY LOCATION(*) DETAIL) shows conversation activity for this DB2 system conversation (SYSCON-O) service task.

RQ A distributed thread is accessing a remote site on behalf of a request from another location.

The thread is suspended for one of the following reasons:

- The maximum number of active database access threads was reached. The number of threads is described by the MAX REMOTE ACTIVE value of the DSN6SYSP macro in the DB2 startup parameter, usually DSNZPARM.
- The system profile monitoring threshold was reached. The threshold is described in the DSN_PROFILE_ATTRIBUTES table.

Database access agents (DBAAs) are queued until other DBAAs deallocate or go inactive, providing an available slot. The DBAA resumes when a slot becomes available, and the DBAA is next in the queue.

Consider increasing the MAX REMOTE ACTIVE value.

RX The distributed thread is executing an XA transaction on behalf of a request from another location.

SP A thread is running within a stored procedure. This status value is displayed until the stored procedure terminates and returns control to DB2.

SW A thread is waiting for a stored procedure to be scheduled. This status value is displayed until the stored procedure begins to run.

T An allied, nondistributed thread was established (plan allocated).

TD An allied thread was established (plan allocated), and the thread is currently not associated with any TCB.

TN An allied thread was distributed to access data at another DB2 location, but it was suspended because DB2 was not connected to the partner location.

When another DB2 site is being contacted for the first time using system-directed access, DB2 must establish DB2 system conversations with the partner location. A request is presented to a DB2 DDF service task that establishes a system conversation. The thread is suspended until the request is processed. If the thread remains in this status for an extended period of time, the DB2 service task that establishes the system conversations might be busy processing other requests. The DB2 DISPLAY LOCATION command (DISPLAY LOCATION(*) DETAIL) shows conversation activity for this DB2 system conversation (SYSCON-O) service task.

TR An allied thread was distributed to access data at another location.

A *t* An indicator that the thread is active within DB2. *t* can have one of the following values:

* The thread is active.

blank The thread is inactive.

REQ *num*

A counter that shows the number of DB2 requests. This counter restarts at 1 when the count exceeds 32767 requests.

ID *text* The recovery correlation ID that is associated with the thread.

DSNV401I

For distributed database access threads, this field contains one of the following values:

028.DBAA *nn*
> This value indicates that the database access agent (DBAA) is performing connection processing, which consists of establishing the DBAA thread and validating and verifying the user's ID.
>
> RACF requests are serialized on one of the DB2 RACF service tasks. This serialization might increase the validation time when multiple DBAAs are being created concurrently.
>
> This value is possible only while the thread is being created.

requester-value
> For threads that use system-directed access or application-directed access from a DB2 requester, this value is the correlation ID of the thread at the requesting location.
>
> This value is possible only after the thread has been created.

external-name
> For threads that use application-directed access from a non-DB2 requester, this value is the first 12 characters in the DDM external name (EXTNAM) parameter of the DDM EXCSAT command that is received as part of the SQL CONNECT statement.
>
> This value is possible only after the thread has been created.

AUTHID *text*
> The authorization ID that is associated with a signed-on connection.

PLAN *text*
> The plan name that is associated with the thread.
>
> For distributed database access threads, this field contains one of the following values:

DISTSERV
> For threads that use application-directed access from a non-DB2 requester, no plan name is identified.

requester-value
> For threads that use system-directed access or application-directed access from a DB2 requester, this value is the plan name that is being executed at the requesting location.

> For RRSAF connected threads that did not specify a plan name at connect time, the value ?RRSAF is displayed.

> If a thread was not established, this field is blank.

> If this value is a system plan, see the information about system plans in message DSNT376I.

ASID *hex*
> The address space ID (ASID) of the home address space. This value is a hexadecimal number of up to four characters.

TOKEN *num*
> The thread token that is assigned to the thread.
>
> For threads with a connection status value of "N", this value is "0".
>
> If multiple threads have the same token, those threads are processing secondary connections in support of an application that uses package-based continuous block fetch.

The information about types of threads ends with a `REPORT COMPLETE` line.

Information about indoubt and postponed threads (DSNV406I and DSNV431I):

The DISPLAY THREAD command reports the same type of information about indoubt and postponed threads.

Information about indoubt threads begins with a DSNV406I message.

```
| DSNV406I - INDOUBT THREADS -
| COORDINATOR              STATUS    RESET URID       AUTHID
| text                     text      txt   hex        text
| DISPLAY INDOUBT REPORT COMPLETE
```

Information about postponed abort threads begins with a DSNV431I message.

```
| DSNV431I - POSTPONED ABORT THREADS -
| COORDINATOR              STATUS    RESET URID       AUTHID
| text                     text      txt   hex        text
| DISPLAY POSTPONED ABORT REPORT COMPLETE
```

Each thread that meets the report criteria is listed on a separate line. Additional information might be provided for individual threads.

COORDINATOR *text*

> The name of the two-phase commit coordinator. The value depends on the type of thread, as follows:

> **Allied threads**
>> The coordinator name is one of the following values:
>> - The IMS connection name
>> - The CICS connection name
>> - RRS (for Resource Recovery Services attachment facility connected threads)
>> - The location name of the local DB2 subsystem.
>>
>> Additional information is displayed for IMS and CICS connections and for RRS.

> **Database access threads**
>> The coordinator name is one of the following values:
>> - *location*:*luname*, where *location* is the relational database name for the coordinator and *luname* is the SNA LU name for the coordinator.
>> - *location*:*port*, where *location* is the relational database name for the coordinator and *port* is the TCP/IP port number for two-phase commit resynchronization for the coordinator.
>> - <*luname*>:*luname*, where *luname* is the SNA LU name for the coordinator.
>> - *nnn.nnn.nnn.nnn*:*port* , where *nnn.nnn.nnn.nnn* is the IP address for the coordinator and *port* is the TCP/IP port number for two-phase commit resynchronization for the coordinator.
>>
>> If the thread is distributed, additional information is displayed.

STATUS *text*

> The status of the thread.

> If the status of a thread for a logical unit of work at a participant is INDOUBT, COMMITTED-H, or ABORTED-H, and the coordinator is a DB2 location that has not cold started, the lack of any information about the thread in a DISPLAY THREAD(INDOUBT) report indicates that the decision at the coordinator was to abort the logical unit of work.

> This field can have one of the following values:

> **ABORT-P**
>> A postponed abort unit of recovery. Objects for which this unit of recovery has backout work pending are inaccessible (restart-pending status) until the abort is completed (for example, by means of the RECOVER POSTPONED command).

> **ABORT-PSTRT**
>> A postponed abort unit of recovery that is currently undergoing RECOVER POSTPONED processing or automatic DB2 backout processing (requested by restarting with system parameter LBACKOUT = AUTO).

> **ABORTED**
>> A coordinator status that indicates that DB2 has one or more downstream participants that are using the Presumed Nothing (PN) protocol and might be indoubt. The thread is included in the

DSNV401I

DISPLAY THREAD output until all downstream PN participants complete indoubt resolution. The detail line V450 lists the downstream participants that have pending resolution. Database locks that were held for the thread are released.

If a participant resolved the indoubt thread with a heuristic decision, and the decision was COMMIT, detail line V452 is displayed. This detail line contains the name of the participant and the heuristic decision that it made. Because the coordinator aborted and the participant committed, this situation indicates heuristic damage. Information about the thread is displayed until it is deleted with the RESET INDOUBT command. The database administrators at all involved locations need to know that heuristic damage occurred and at which location.

ABORTED-H

An indoubt thread that was heuristically resolved with the RECOVER INDOUBT command. Information about the thread is displayed is displayed until the coordinator is informed of the heuristic decision. Additional detail lines might be included in the report.

COMMITTED

A thread that is committed. This status is displayed when DB2 is the coordinator and has downstream participants that might be indoubt. Information about the thread is included in the DISPLAY THREAD output until all downstream participants complete indoubt resolution. Detail line V450 lists the downstream participants that have pending resolution. Database locks that were held for the thread are released.

If a participant resolved the indoubt thread with a heuristic decision, and the decision was ABORT (rollback), detail line V453 is displayed. This detail line contains the name of the participant and the heuristic decision that it made. Because the coordinator committed and the participant aborted, this situation indicates heuristic damage. Information about the thread is displayed until it is deleted with the RESET INDOUBT command. The database administrators at all involved locations need to know that heuristic damage occurred and at which location.

COMMITTED-H

An indoubt thread that was heuristically resolved with the RECOVER INDOUBT command. Information about the thread is displayed until the coordinator is informed of the heuristic decision, and all downstream participants, if any, complete indoubt resolution. Additional detail lines might be included in the report to identify downstream participants that have pending resolution.

INDOUBT

A thread that is indoubt. Information about the thread is displayed until all indoubt resolution responsibilities are complete. Additional detail lines might be included in the report.

RESET *txt*

If this column contains "YES", the thread must be removed from the indoubt display. If necessary, use the RESET INDOUBT command to purge a thread. If this column is empty, the thread does not need to be removed.

URID *hex*

An RBA of the DB2 recovery log. This point is the beginning of recovery logging for this thread.

AUTHID *text*

The primary authorization ID that is associated with the thread.

The information about types of threads ends with a `REPORT COMPLETE` line.

V429: Information about stored procedures:

If a thread is running a stored procedure, the output includes the following information about the stored procedure.

```
NAME     ST A  REQ ID              AUTHID   PLAN     ASID TOKEN
text     t  t  num text            text     text     hex  num
  V429 CALLING PROCEDURE=text, LOAD MODULE=text,
       PROC=text, ASID=hex, WLM_ENV=text
```

PROCEDURE *text*

The name of the stored procedure.

LOAD MODULE *text*

The MVS load module that is associated with the stored procedure. This value is empty until the load module name is determined.

PROC *text*

The name of the JCL PROC that is used to start the address space where the stored procedure is running. This field is empty until the stored procedure is assigned to a specific stored procedure address space.

ASID *hex*

The MVS address space ID (ASID) of the address space where the stored procedure is running. This field contains the value 0000 until the stored procedure is assigned to a specific stored procedure address space.

WLM_ENV *text*

The name of the WLM application environment where the stored procedure is running.

V436: Information about SQL statements:

If you specify a detail report and a thread is processing an SQL statement, the output includes the following information about the SQL statement and the program that contains the statement.

```
NAME     ST A   REQ ID              AUTHID   PLAN     ASID TOKEN
text     t  t   num text            text     text     hex  num
  V436-PGM=text, SEC=num, STMNT=num, THREAD-INFO=text
```

PGM *text*

The collection ID and associated package or DBRM member. This information is provided in the following format:

collection.package

If the SQL statement that is being executed is not associated with a package, *collection* is not applicable and is indicated by an asterisk (*).

SEC *num*

The SQL section number that is associated with the package or DBRM.

STMT *num*

The SQL statement number that is associated with the package or DBRM.

THREAD-INFO *text*

Information about the thread. The information is presented in a colon-delimited list that contains the following segments:

- The primary authorization ID that is associated with the thread.
- The name of the user's workstation.
- The ID of the user.
- The name of the application.
- The statement type for the currently executing statement: dynamic or static.
- The statement identifier for the currently executing statement, if available. The statement identifier can be used to identify the particular SQL statement.
  - For static statements, the statement identifier correlates to the STMT_ID column in the SYSIBM.SYSPACKSTMT table.
  - For dynamic statements, the statement identifier correlates to the STMT_ID column in the DSN_STATEMENT_CACHE_TABLE table.
- The name of the role that is associated with the thread.
- The correlation token that can be used to correlate work at the remote system with work that runs at the DB2 subsystem. The default correlation token, if available, is enclosed in '<' and '>' characters, and contains three components, which are separated by periods:
  - A 3 - 39 character IP address
  - A 1 - 8 character port address
  - A 12 character unique identifier

An asterisk (*) in any segment indicates that the information is not available.

V437: Information about the user and application:

If the client system is able to provide specific information about the user that is associated with this thread, this information is included in the output. If the information is not provided by the client system, the relevant fields contain an asterisk (*).

```
| NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
| text      t  t  num text          text      text
|   V437-WORKSTATION=text
|        USERID=text
|        APPLICATION NAME=text
```

WORKSTATION *text*
> The workstation name

USERID *text*
> The user ID

APPLICATION NAME *text*
> The name of the client application

V440: Information about threads that are managed by an XA transaction manager:

If the thread is related to a global transaction that is coordinated by an XID, the output includes XID information. The DISPLAY THREAD command must include the TYPE(ACTIVE) and DETAIL options, or the TYPE (INDOUBT) option for this information to be included.

Line V440 indicates that the thread is managed by an XA transaction manager, such as WebSphere® Application Server, which identifies the transaction with an *xid* value. The XID is provided to allow correlation with the XA transaction manager. DB2 uses both the logical unit of work identifier, *luwid*, and the XA transaction identifier, *xid*, to coordinate and recover transactions. DB2 is a participant in an XA transaction that is executing on behalf of an XA transaction manager.

```
| NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
| text      t  t  num text          text      text      hex  num
|   V440-XID=hex
```

XID *hex*
> The XID is provided in its hexadecimal representation and consists of the following elements:
>
> - A 4-byte format ID.
> - A 4-byte global transaction identifier length (the gtrid_length field).
> - A 4-byte branch qualifier length (bqual_length).
> - Variable-length data that is composed of at most two contiguous components: A global transaction identifier length (gtrid_length), and a branch qualifier length (bqual_length). The gtrid_length element specifies the number of bytes (1-64) that constitute the gtrid, starting at the first byte of the data element. The bqual-length element specifies the number of bytes (up to 64) that constitute bqual, starting at the first byte after gtrid.

V441: User accounting information:

If the client system is able to provide user accounting information for the thread, this information is included in the output. The DISPLAY THREAD command must include the DETAIL option and either the TYPE(ACTIVE), TYPE(INACTIVE), or TYPE(INDOUBT) option for this information to be included.

```
| NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
| text      t  t  num text          text      text      hex  num
|   V441-ACCOUNTING=text
```

ACCOUNTING *text*
> The client end-user accounting information. This information is provided if the appropriate information is provided by the client system, and if DB2 recognizes the format of the information.

V442: Information about correlation tokens:

If the client system is able to provide a correlation token that can be used to correlate work at the remote system with work that is performed at the DB2 subsystem, this information is included in the output.

```
| NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
| text      t  t  num text          text      text      hex  num
|   V442-CRTKN=text
```

CRTKN *text*
> The correlation token, which is made up of three components separated by periods:
>
> *ip-address.port-address.unique-id*

ip-address
> The IP address is 3 to 39 characters in length.

port-address
> The port address is 1 to 8 characters in length.

unique-id
> The ID is 12 characters in length.

V444: Information about distributed threads (for active, inactive, stored procedure, and system threads):

If any threads were distributed to other locations, the output includes the logical unit of work identifier for each distributed thread.

```
NAME     ST A  REQ ID          AUTHID   PLAN     ASID TOKEN
text     t  t  num text        text     text
         text
  V444-text=num ACCESSING DATA AT
   text-address..number
```

text=num **ACCESSING DATA AT**
> Information to identify the distributed thread as follows:

> *text*   The global logical unit of work ID (LUWID).

> *num*   The local token that identifies the thread. This token can be used in place of the LUWID in any DB2 command that accepts LUWID as input.

> If the thread becomes indoubt, a new token is assigned for the indoubt thread.

text-address..number
> The remote DBMS at which data is being accessed.

> *text*   The partner location name.

> *address* The partner LU name for an SNA connection, or the dotted decimal IP address for a TCP/IP connection (nnn.nnn.nnn.nnn).

> *number* An SQL port number. This token is included only if the second token is an IP address.

V445: Information about database access threads:

The output includes the following information about database access threads:

```
NAME     ST A  REQ ID          AUTHID   PLAN     ASID TOKEN
text     t  t  num text        text     text
         text
  V445-text=num ACCESSING DATA FOR
   text
```

text=num
> Information to identify the distributed thread as follows:

> *text*   The global logical unit of work ID (LUWID).

> *num*   The local token that identifies the thread. This token can be used in place of the LUWID in any DB2 command that accepts LUWID as input.

> If the thread becomes indoubt, a new token is assigned for the indoubt thread.

ACCESSING DATA FOR *text*
> The network address of the remote location. This value can be blank.

> - If the connection with the requester is through SNA, this field contains the relational database name of the requester or the VTAM LU name of the requester, a dash (-) delimiter, and the LU name of the requester.
> - If the connection with the requester is through TCP/IP, this field contains the IP address of the requester.

V447 and V448: Information about conversations:

The output contains conversation information for active and inactive threads. Conversation information is displayed for a distributed access thread, distributed inactive connection, or distributed allied thread when a detail report is

DSNV401I

specified. Information is displayed for each conversation that is associated with the thread or connection. The report lists detailed information regarding the connection.

```
NAME      ST A   REQ ID               AUTHID    PLAN      ASID TOKEN
text      t  t   num text             text      text
              text
  V447--INDEX SESSID        A ST TIME
  V448--(txt) text          t tx timestamp
```

INDEX (_text_**)**

> The index value from the location information in the V445 output line. This value provides correlation of the conversation detail information to the associated location.

SESSID _text_

> The session identifier. The session identifier is either a pair of TCP/IP port numbers or a VTAM-defined session instance identifier.

> **TCP/IP connections**
>> The local DB2 TCP/IP port number followed by the TCP/IP port number for the partner. The two port numbers are separated by a colon. For example: 5001:28191.

> **SNA partners**
>> The VTAM-defined session instance identifier of the session on which the conversation is running. The field contains zeros if the session identifier is not applicable.

>> VTAM might not supply the entire session ID (SID) to DB2. The first two digits of the SID might be incorrect, in which case the SID, as presented in this message, is not acceptable in VTAM commands. If this occurs, use the VTAM DISPLAY NET,ID=_db2-luname_,SCOPE=ACT command to obtain the full SID. The DISPLAY NET command lists all sessions for the DB2 logical unit. Review the DISPLAY NET report for the complete session ID.

A _text_ A single alphabetic character that indicates activity. This field can contain one of the following values:

> **N** The conversation is active in the network.

> **W** The conversation is suspended in DB2 while it waits for the network notification that the function is complete.

> **Blank** All other cases.

ST _text_ The status of the conversation. This information is presented as two characters and can contain the following values:

> **First character**

>> **A** Conversation is in allocation.

>> **C** Session limits are being negotiated with the partner (CNOS) prior to conversation allocation.

>> **D** Conversation is in deallocation.

>> **R** Receiving.
>>> • At the requesting site, this value indicates that the conversation is receiving a response from a request.
>>> • At the server site, this value indicates that the conversation is receiving or waiting for a request.

>> **S** Sending.
>>> • At the requesting site, this value indicates that the conversation can send requests to the server.
>>> • At the server site, this value indicates that the conversation is sending or preparing to send a response.

>> **X** Exchanging log name information with the partner before conversation allocation to determine if the partner supports protected conversations. If the thread remains in this status for an extended period of time, the DB2 service task that exchanges log names

| might be busy processing other requests. The DB2 DISPLAY LOCATION command (DISPLAY LOCATION(*) DETAIL) shows you conversation activity for this (RESYNC) service task.

Blank All other cases.

Second character

1 Unprotected conversation using system-directed access is active.

2 Unprotected conversation using application-directed access is active.

3 Protected conversation using system-directed access is active.

4 Protected conversation using application-directed access is active.

5 A distributed thread is processing a secondary connection in support of an application that uses package-based continuous block fetch.

Blank All other cases.

TIME *timestamp*
The timestamp (*yydddhhmmssth*), in local time, of the last message that was sent or received on the conversation. The timestamp consists of the following segments:

yy Two-digit indicator for the year.

ddd Three-digit indicator for the day of the year.

hhmmssth
Indicator for the time of the day.

V449: Information about IMS and CICS connections and RRS:

When the coordinator is an IMS or CICS connection or is RRS, the output includes the following additional information about the coordinator.

```
COORDINATOR              STATUS    RESET URID       AUTHID
text                     text      txt   hex        text
  V449-HAS NID=text AND ID=text
```

NID *text*
The network ID that is assigned by IMS or CICS, or, if the connection type is RRSAF, the RRS unit of recovery ID.

ID *text* The correlation ID that is assigned by IMS or CICS, or, if the connection type is RRSAF, the correlation ID that is assigned by the connected application.

V450: Information about downstream participants that might be indoubt:

If you specify TYPE(INDOUBT) in the DISPLAY THREAD command, the output also includes the following information about any downstream participants that might be indoubt. The indoubt participants that are listed in the output might be the result of an abnormal termination of this DB2 subsystem. The LUWID of the indoubt thread is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR              STATUS    RESET URID       AUTHID
text                     text      txt   hex        text
  V450-HAS PARTICIPANT INDOUBT AT
   text-address..number
```

HAS PARTICIPANT INDOUBT AT *text address number*
The name of a remote DBMS at which data is being accessed or where a thread is indoubt.

- *text* is the partner location name.
- *address* is the partner LU name for an SNA connection, or the dotted decimal IP address for a TCP/IP connection (nnn.nnn.nnn.nnn).
- *number* is a resynchronization port number. This token is included only if the second token is an IP address.

V451: Information about incomplete resolution of indoubt threads:

Line V451 indicates that an indoubt thread was manually resolved with the RECOVER INDOUBT command, but indoubt resolution with the coordinator is not complete. The thread remains indoubt. The LUWID of the indoubt

thread is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR            STATUS    RESET URID      AUTHID
text                   text      txt   hex       text
  V451-RESYNC WITH COORDINATOR STILL PENDING
```

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V452: Information about heuristic commit decisions:

Line V452 indicates that a decision was made at the participant to force the indoubt thread to commit without waiting for automatic resynchronization to be performed. The LUWID of the indoubt thread is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR            STATUS    RESET URID      AUTHID
text                   text      txt   hex       text
  V452-HEURISTIC COMMIT BY PARTICIPANT AT LOCATION text
```

HEURISTIC COMMIT BY PARTICIPANT AT LOCATION *text*
> The location (relational database name) where the HEURISTIC COMMIT was performed.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

This information is displayed until DB2 resolves the indoubt status with the coordinator. If the coordinator decision is the same as the heuristic decision of the partner, the detail line is no longer displayed.

V453: Information about heuristic abort decisions:

Line V453 indicates that a decision was made at the participant to force the indoubt thread to abort without waiting for automatic resynchronization to be performed. The LUWID of the indoubt thread that was forced to abort is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR            STATUS    RESET URID      AUTHID
text                   text      txt   hex       text
  V453-HEURISTIC ABORT BY PARTICIPANT AT LOCATION text
```

HEURISTIC ABORT BY PARTICIPANT AT LOCATION *text*
> The location (relational database name) where the HEURISTIC ABORT was performed.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

This information is displayed until DB2 resolves the indoubt status with the coordinator. If the coordinator decision is the same as the heuristic decision of the partner, this detail line is no longer displayed.

If the coordinator status is COMMITTED and the heuristic decision of the participant is ABORT, the message continues to be displayed until it is deleted with the RESET INDOUBT command. The database administrators at all involved locations need to know that heuristic damage occurred and at which location.

V457: Information about resync protocol errors with participants:

Line V457 indicates that a protocol error was detected during attempted automatic indoubt resolution with the participant. Manual resolution of an indoubt thread at the participant location might be required. The LUWID of the thread that might require manual resolution is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR            STATUS    RESET URID      AUTHID
text                   text      txt   hex       text
  V457-RESYNC PROTOCOL ERROR WITH PARTICIPANT text
```

RESYNC PROTOCOL ERROR WITH PARTICIPANT *text*
> The location where manual resolution might be needed. The location is either the relational database name or the logical unit name (LU name) of the participant.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V458: Connection information for indoubt threads:

If you specify TYPE(INDOUBT) in the DISPLAY THREAD command, the output contains the following connection information :

```
| COORDINATOR              STATUS    RESET URID        AUTHID
| text                     text      txt   hex         text
|   V458-CONNECTION-NAME text CORRELATION-ID text
```

CONNECTION-NAMEtext
> The connection name that is assigned by the coordinator if the coordinator is another DB2 subsystem. The field contains the constant SERVER if the coordinator is not a DB2 subsystem.

CORRELATION-ID text
> The correlation ID that is assigned by the coordinator if the coordinator is another DB2 subsystem. The field contains the first 12 characters of the EXTNAME parameter of the EXCSAT command that is sent by the coordinator if the coordinator is not a DB2 subsystem.

V459: Information about syncpoint protocol errors with coordinators:

Line V459 indicates that a protocol error was detected during sync point processing with the coordinator. Manual resolution of an indoubt thread at this location is required. The LUWID of the thread that requires manual resolution is displayed in the COORDINATOR field prior to this message.

```
| COORDINATOR              STATUS    RESET URID        AUTHID
| text                     text      txt   hex         text
|    V459-SYNCPOINT PROTOCOL ERROR WITH COORDINATOR
```

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V460: Information about syncpoint protocol errors with participants:

Line V460 indicates that a protocol error was detected during sync point processing with the participant. Manual resolution of an indoubt thread at the participant location might be required. The LUWID of the thread that might require manual resolution at the participant is displayed in the COORDINATOR field prior to this message.

```
| COORDINATOR              STATUS    RESET URID        AUTHID
| text                     text      txt   hex         text
|     V460-SYNCPOINT PROTOCOL ERROR WITH PARTICIPANT text
```

SYNCPOINT PROTOCOL ERROR WITH PARTICIPANT text
> The location where manual resolution might be needed. The location is either the relational database name or the logical unit name (LU name) of the participant.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V461: Information about heuristic damage:

Line V461 indicates that indoubt resolution with the coordinator completed. Heuristic damage was detected at this location because the heuristic decision that was made at this location is different than the decision that was made at the coordinator. The thread remains indoubt. The LUWID of the thread is displayed in the COORDINATOR field prior to this message. The STATUS field shows the heuristic decision that was made at this location.

This message is displayed until it is removed by a RESET INDOUBT command.

```
| COORDINATOR              STATUS    RESET URID        AUTHID
| text                     text      txt   hex         text
|   V461-RESYNC WITH COORDINATOR SHOWS HEURISTIC DAMAGE
```

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V462: Information about cold starts by participants:

Line V462 indicates that DB2 has coordinator responsibility for an indoubt thread at the participant, and the participant informed DB2 that it performed a cold start operation and lost all knowledge of indoubt threads. Heuristic damage might have occurred at the participant. The LUWID of the indoubt thread is displayed in the COORDINATOR field prior to this message.

Use the RESET INDOUBT command to purge this message from the indoubt report. The cold-starting participant might have been coordinating downstream participants that are now indoubt. The database administrator at these participants might need to know the status of the thread at the coordinator, so that correct heuristic decisions can be made.

```
| COORDINATOR              STATUS    RESET URID        AUTHID
| text                     text      txt   hex         text
|   462-COLD START BY PARTICIPANT AT LOCATION text
```

DSNV401I

COLD START BY PARTICIPANT AT LOCATION *text*
>   The location (relational database name) where the cold start was performed.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V463: Information about cold starts by coordinators:

Line V463 indicates that one of the following situations occurred:

- The coordinator for a thread that is indoubt at a DB2 subsystem was cold started and therefore lost all knowledge of indoubt threads.
- The knowledge of the local system about the coordinator was cold started, for example, by the RESET INDOUBT FORCE command.

Manual resolution of an indoubt thread at this location is required. The LUWID of the thread for which manual resolution is required is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR              STATUS     RESET URID      AUTHID
text                     text       txt   hex       text
   V463-COORDINATOR HAS COLD STARTED
```

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V464: Information about resync protocol errors with coordinators:

Line V464 indicates that a protocol error was detected during attempted automatic indoubt resolution with the coordinator. Manual resolution of an indoubt thread at this location is required. The LUWID of the thread that requires manual resolution is displayed in the COORDINATOR field prior to this message.

```
COORDINATOR              STATUS     RESET URID      AUTHID
text                     text       txt   hex       text
   V464-RESYNC PROTOCOL ERROR WITH COORDINATOR
```

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V465: Information about threads that remain in the prepared state:

Line V465 indicates that the thread was prepared for the specified period of time. DB2 is the participant in the logical unit of work. DB2 has completed the first phase of commit and is waiting for the commit or abort decision to be transmitted from the coordinator for the specified period of time. This message is not included in the output until the thread is in the prepared state for 60 seconds. The thread is identified by the line that precedes this message.

You must specify TYPE (ACTIVE) in the DISPLAY THREAD command for this information to be displayed.

```
NAME     ST A   REQ ID             AUTHID    PLAN      ASID TOKEN
text     t  t   num text           text      text      hex  num
   V465-THREAD HAS REMAINED IN A PREPARED STATE FOR timestamp
```

timestamp
>   The length of time that the thread was prepared, in the format *hh:mm:ss* (hours, minutes and seconds). The maximum number of reported hours is 65535, which is roughly 7.5 years. After this time period elapses, the count is reset to zero.

You can take the following actions to try to resolve this thread manually (that is, take heuristic action) if you need to release locks that are held by this thread:

1. Use the CANCEL DDF THREAD command to force the thread from the prepared state to the indoubt state. When the thread is indoubt, it is displayed in the DISPLAY THREAD TYPE(INDOUBT) report.
2. Determine the proper decision (commit or abort) by contacting the coordinator location, as shown in the display of indoubt threads.
3. Use the RECOVER INDOUBT command to heuristically commit or abort the thread.

V466: Information about how long a thread was indoubt:

Line V466 indicates that the thread entered the indoubt state at the specified time. DB2 is the participant in the logical unit of work, has completed the first phase of commit, and is waiting for the commit or abort decision to be transmitted from the coordinator. However, DB2 lost communications with the coordinator. The LUWID of the thread is displayed in the COORDINATOR field prior to this message.

```
| COORDINATOR                STATUS    RESET URID        AUTHID
| text                       text      txt   hex         text
|   V466-THREAD HAS BEEN INDOUBT FOR timestamp
```

THREAD HAS BEEN INDOUBT FOR *timestamp*

> The length of time that the thread was indoubt, in the format *hh:mm:ss* (hours, minutes and seconds). The maximum number of reported hours is 65535, which is roughly 7.5 years. After this time period elapses, the count is reset to zero.

If necessary, you can release locks that are held by this thread by performing the following actions:

1. Determine the proper decision (commit or abort) by contacting the coordinator location, as shown in the display of indoubt threads.
2. Use the RECOVER INDOUBT command to heuristically commit or abort the thread.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V467: Information about distributed threads (for indoubt and postponed threads):

For indoubt and postponed distributed threads, the output includes the following additional information.

```
| COORDINATOR                STATUS    RESET URID        AUTHID
| text                       text      txt   hex         text
|   V467-HAS LUWID text=num
```

LUWID *text num*

> The global logical unit of work ID and the local token that identifies the thread.

You must specify TYPE(INDOUBT) in the DISPLAY THREAD command for this information to be displayed.

V471: Information about available agents:

Line V471 indicates that the agent is currently not associated with a remote location. In the following situations, the agent can be assigned to a process:

- The completion of a queued receive request on a type 2 inactive thread.
- A request that a new connection be established, after MAX REMOTE ACTIVE has been reached.

```
| NAME     ST A   REQ ID           AUTHID     PLAN      ASID TOKEN
| text     t  t   num text         text       text      hex  num
|   V471-luwid=token
```

luwid=token

> The global logical unit of work ID and the local token that identifies the agent.

V480 and V481: Information about the unit of recovery:

When the DISPLAY THREAD command is issued with the RRSURID option, additional information is provided about the unit of recovery.

Information about the participant is provided in line V480 when the TYPE(ACTIVE) option is used.

```
| NAME     ST A   REQ ID           AUTHID     PLAN      ASID TOKEN
| text     t  t   num text         text       text      hex  num
|   V480-DB2 IS PARTICIPANT FOR RRS URID=hex
```

URID *hex*

> The Recoverable Resource Services (RRS) unit of recovery ID in which this thread participates.

Information about the coordinator is provided in line V481 when either the TYPE(ACTIVE) or the TYPE(INDOUBT) option is used.

```
| NAME     ST A   REQ ID           AUTHID     PLAN      ASID TOKEN
| text     t  t   num text         text       text      hex  num
|   V481-DB2 IS COORDINATOR FOR RRS URID=hex
```

URID *hex*

> The Recoverable Resource Services (RRS) unit of recovery ID for which this thread is coordinator.

DSNV401I

When the V481 message is included in an INDOUBT thread report, the preceding lines indicate the status of the DB2 unit of recovery that contains the RRS unit of recovery. If manual recovery of the RRS unit of recovery is required, the outcome of the DB2 unit of recovery can determine whether to commit or rollback the RRS unit of recovery.

V482: WLM information:

If you request a detail report and the thread is associated with a z/OS Workload Manager (WLM) enclave, the output includes an indication that WLM is currently applying its performance goals to the thread.

```
V482 WLM-INFO=text
```

text   Information about the WLM characteristics that are associated with the thread. This field includes the values of the following identifiers when known, separated by colons:

- The name of the service class.
- The service class period number.
- The importance level of the period.
- The performance index of the service class period, scaled by 100. A value of 100 corresponds to a value of 1.0 in the corresponding RMF™ report.

An asterisk (*) indicates that the value is not known or is not applicable.

V485: Information about trusted contexts:

If a thread is associated with a trusted context, the output includes the trusted context name, system authorization ID, and role.

This information is included when a non-detail report is specified.

```
NAME      ST A   REQ ID               AUTHID    PLAN      ASID TOKEN
text      t  t   num text             text      text      hex  num
  V485-TRUSTED CONTEXT=text,
       SYSTEM AUTHID=text,
       ROLE=text
```

TRUSTED CONTEXT *text*
    The trusted context name.

SYSTEM AUTHID *text*
    The system authorization ID that is associated with the user in the trusted context.

ROLE *text*
    The role that is associated with the user in the trusted context. If the role is not defined for the trusted context, this field contains an asterisk (*).

V490: Information about suspended threads:

Line V490 indicates that a thread in your system is currently suspended. This message is informational only.

```
NAME      ST A   REQ ID               AUTHID    PLAN      ASID TOKEN
text      t  t   num text             text      text      hex  num
   V490- SUSPENDED timestamp smodule soffset smodptf-level
```

timestamp

    The suspend date and time. The format is YYDDD-HH:MM:SS.HSEC.

smodule soffset **and** *smodptf-level*
    These fields are for the use of IBM Software Support.

You must specify TYPE(SYSTEM) in the DISPLAY THREAD command for this information to be displayed.

V492: Information about storage:

The V492 line is an informational message for use by IBM Software Support.

```
NAME      ST A   REQ ID               AUTHID    PLAN      ASID TOKEN
text      t  t   num text             text      text      hex  num
  V492-LONG numK 64VLONG numK 64LONG numK
```

V501: Information about commands:

Line V501 indicates that the system thread is currently executing the specified command. The thread is identified by the preceding lines of output.

```
NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
text      t  t  num text          text      text      hex  num
V501-COMMAND EXECUTING:  text
```

text   The command that the thread is currently running.

V502: Information about DB2 exit routines:

If a system thread is currently executing a DB2 exit routine that you specified, the output includes the following additional information about the thread. The thread is identified by the preceding lines of output.

```
NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
text      t  t  num text          text      text      hex  num
V502-PLOCK ENGINE text EXIT, RMID=text,FCODE=texttext
```

text   The name of the DB2 exit routine that the thread is running.

RMID=*text***, FCODE=***text text*

These fields are for the use of IBM Software Support.

RMID=*dec*

Resource manager identifier (RMID).

FCODE=*text*

Feature code.

text   A text description, if available.

V504: Information about threads that are terminating:

If a system thread is currently terminating another thread, the output includes the following additional information about the thread. The thread is identified by the preceding lines of output.

```
NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
text      t  t  num text          text      text      hex  num
V504-TERMINATING CORRID=text, CONNID=text, HASID=text
```

CORRID=*text*

The specified correlation ID of the thread that is being terminated.

CONNID=*text*

The connection ID of the thread that is being terminated.

HASID=*text*

The home address space ID of the thread that is being terminated.

V505: Information about threads that are waiting:

Line V505 indicates that a system thread is experiencing a long wait time for reading or writing data for the specified object. The thread is identified by the preceding lines of output.

```
NAME      ST A  REQ ID            AUTHID    PLAN      ASID TOKEN
text      t  t  num text          text      text      hex  num
V505-BM PROCESSING DBNAME=text, SPACE NAME= text, PART= num
```

DBNAME=*text*

The name of the database.

SPACE NAME= *text*

The name of the table space or index space.

PART= *num*

A number that identifies the partition. For non-partitioned data sets, no value is displayed.

More detailed information about the suspended thread is reported in line V490.

V507: Information about monitoring tasks:

If the thread is currently an internal DB2 monitoring task, the output includes the following additional information about the thread.

DSNV401I

```
V507-text MONITOR, INTERVALS=monintervals, STG=monstg, BOOSTS=monboosts, HEALTH=text
     REGION=num, AVAIL=num, CUSHION=num
```

text **MONITOR**

The type of monitoring task: active or inactive.

monintervals, *monstg*, *monboosts*, **and** *text*

These fields are for the use of IBM Software Support.

REGION *num*

The size of the region, as a number and unit of measurement.

AVAIL *num*

The amount of available storage, as a number and unit of measurement.

CUSHION *num*

The size of the storage cushion, as a number and unit of measurement. The storage cushion is a defined portion of total storage that is available for use, but whose use triggers alerts. The purpose of the alerts is to notify administrators of unexpected storage requirements and help them manage the condition.

You must specify TYPE(SYSTEM) in the DISPLAY THREAD command for this information to be displayed.

V520: Information about autonomous procedures:

When a thread is processing an autonomous procedure, the output includes the following additional information about the thread.

```
NAME     ST A   REQ ID           AUTHID   PLAN     ASID TOKEN
text     t  t   num text         text     text     hex  num
  V520-AUTONOMOUS PROCEDURE INVOKED BY THREAD
      WITH TOKEN=num
```

TOKEN *num*

The token of the invoking thread for the autonomous procedure.

Information that indicates NOT FOUND situations:

If no threads or connections of the requested type are found, one of the following messages is displayed.

NO CONNECTIONS FOUND

The DISPLAY THREAD TYPE(*), TYPE(ACTIVE) or TYPE(INACTIVE) command was unable to locate any connections within the subsystem.

NO CONNECTION FOUND FOR NAME=*text*

The DISPLAY THREAD TYPE(ACTIVE) or TYPE(INACTIVE) command was unable to locate any connections that are associated with the indicated connection name. This message is generated once for each connection name for which no connection was found.

NO THREADS FOUND FOR NAME=*text*

The DISPLAY THREAD TYPE(ACTIVE) or TYPE(INACTIVE) command was unable to locate any threads that are associated with the indicated connection name. This message is generated once for each connection name for which no threads were found.

NO SYSTEM THREADS FOUND

The DISPLAY THREAD command did not find any system threads that can be displayed.

NO SYSTEM THREADS FOUND FOR NAME= *connection-name*

The DISPLAY THREAD command did not find any system threads for the named connection.

NO INDOUBT THREADS FOUND

The DISPLAY THREAD command found no indoubt threads within the system. This message is generated when the command requests information about all (*) threads that are currently residing within an indoubt state (TYPE=INDOUBT) within the system, but no threads currently exist within this state.

csect **NO INDOUBT THREADS FOUND FOR CONNECTION NAME=***text*

The DISPLAY THREAD TYPE(INDOUBT) command was unable to locate any indoubt threads that are associated with the indicated connection name. This message is generated once for each connection name for which no indoubt threads were found. If you do not specify a connection name, the command uses a default connection name.

NO POSTPONED THREADS FOUND FOR NAME=*text*

The DISPLAY THREAD command found no postponed abort URs for the named connection. This message is generated when the command requests information about threads for the given connection name that are currently residing within a postponed abort state, but no threads currently exist within this state.

DSNV434I *csect* **NO POSTPONED ABORT THREADS FOUND**

The DISPLAY THREAD TYPE(POSTPONED) command found no postponed abort threads.

Other information reported in the DISPLAY THREAD output:

ARCHIVE LOG QUIESCE CURRENTLY ACTIVE

An ARCHIVE LOG MODE(QUIESCE) command is currently active. Updates against DB2 resources have been temporarily suspended, which might result in active threads being suspended until termination of the quiesce period.

SCOPE(GROUP) SUPPRESSED DURING RESTART AND SHUTDOWN

The DISPLAY THREAD command with the SCOPE(GROUP) option was issued during restart or shutdown. SCOPE(GROUP) is not supported during restart or shutdown. The command runs locally.

ONLY SYSTEM THREAD DATA AVAILABLE

The DISPLAY THREAD command was issued during restart or shutdown and specified a value for the TYPE option other than SYSTEM. This message indicates that only system thread information is available during restart and shutdown.

End of report markers:

The DISPLAY THREAD output normally ends with message DSN9022I.

If the command is not able to complete the report, the output ends with message DSN9023I and one of the following explanations:

DISPLAY THREAD MESSAGE POOL SIZE EXCEEDED

The amount of CSA or ECSA virtual storage that is needed to generate DISPLAY THREAD output exceeded the maximum size of the message buffer pool.

This situation can happen when many threads exist in DB2, and the pattern-matching character '*' is used in a DISPLAY THREAD command, as shown in the following example:

```
-DISPLAY THREAD(*)
```

In this case, processing is terminated.

To correct the error, reduce the amount of CSA or ECSA virtual storage that is required for the DISPLAY THREAD command by performing one of the following actions:

- Specify qualifiers, such as TYPE(INDOUBT), TYPE(ACTIVE), or TYPE(INACTIVE).
- Specify specific *connection-name*, *location-name*, or *luwid* values.

MESSAGE LIMIT EXCEEDED. DISPLAY IS TRUNCATED.

The number of DISPLAY THREAD command messages exceeds the maximum number of permitted messages. The maximum number of messages is specified by the LIMIT option or the default value for the LIMIT option.

DISPLAY THREAD OUTPUT TO BE CONTINUED

DISPLAY THREAD command output directed to an MVS console must be continued in another group of messages. The continuation notice contains a unique command ID that is also displayed in the first line of the continuation.

```
DISPLAY THREAD OUTPUT TO BE CONTINUED (CMDID = command-id)
```

Compare the command ID values to confirm that the continuation is from the same report.

```
DISPLAY THREAD OUTPUT CONTINUED (CMDID = command-id)
```

Error messages:

If the DISPLAY THREAD command fails, you receive one of the following messages:

csect-name **DISPLAY THREAD***pkwname* **TERMINATED, DEFAULT UNAVAILABLE**

DSNV401I

The DISPLAY THREAD command statement did not contain all required parameters. Command processing terminates.

csect-name
> The name of the control section that issued the message.

pkwname
> The command primary keyword as entered by the requester.

One of the following items was not specified: *connection-name*, LOCATION, or LUWID.

Reenter the command with one of these parameters specified.

LOCATION OR LUWID KEYWORD INVALID WHEN DDF NOT STARTED
> The DISPLAY THREAD TYPE(ACTIVE) command was issued with the keyword LOCATION or LUWID, but the distributed data facility (DDF) was not started. The LOCATION or LUWID keyword can be used only if DDF is in operation.

Command processing is complete.

INVALID PARAMETER WITH LUWID KEYWORD *luwid-parameter*

The DISPLAY THREAD TYPE(ACTIVE) or DISPLAY THREAD TYPE(INACTIVE) command was issued with the keyword LUWID and its required parameter list. However, one of the parameters was syntactically invalid as a logical unit of work identifier.

The command returns without processing.

Reenter the command with valid LUWID values.

SCOPE(GROUP) DISPLAY TERMINATED DUE TO IRLM TRUNCATION

The SCOPE(GROUP) keyword was specified, and IRLM truncated the responses gathered from other members. This error is a DB2 internal error.

The DISPLAY THREAD report is terminated and processing abnormally terminates.

Notify the system programmer and contact IBM Software Support to report the problem.

SCOPE(GROUP) DATA UNAVAILABLE

The SCOPE(GROUP) keyword was specified, and the IRLM NOTIFY to other members has returned a bad return code.

```
SCOPE(GROUP) DATA UNAVAILABLE,
        IRLM RETURN CODE: retcode,
        IRLM REASON DATA: reason
```

retcode Return code from IRLM on the Notify request.

reason Reason code diagnostic information from IRLM on the Notify request.

System action: Processing continues.

Operator response: Review the information that is provided for each line of output and take any necessary action.

| **Related concepts**:

| ➡ Autonomous procedures (DB2 Application programming and SQL)

| ➡ TCP/IP terminology (DB2 Installation and Migration)

| ➡ Using service class periods (System Programmer's Guide to: Workload Manager)

| ➡ Using importance levels (System Programmer's Guide to: Workload Manager)

| ➡ Performance index (System Programmer's Guide to: Workload Manager)

| **Related reference**:

| ➡ -DISPLAY LOCATION (DB2) (DB2 Commands)

| ➡ -DISPLAY THREAD (DB2) (DB2 Commands)

| ➡ -RESET INDOUBT (DB2) (DB2 Commands)

| ➡ Performance Index (MVS Programming: Workload Management Services)

| **Related information**:

| "DSN3201I" on page 1108

| "DSN9022I" on page 1144

| "DSN9023I" on page 1145

| ➡ Defining Service Classes and Performance Goals (z/OS MVS Planning: Workload Management)

| ➡ Setting up a Service Definition (z/OS MVS Planning: Workload Management)

| ➡ Using Performance Periods (z/OS MVS Planning: Workload Management)

DSNV402I   ACTIVE THREADS -

Explanation: This message marks the beginning of a section of output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV406I   INDOUBT THREADS -

Explanation: This message marks the beginning of a section of output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV413I   *csect-name verbname pkwname* **TERMINATED, DEFAULT UNAVAILABLE**

Explanation: This message indicates that the command 'verbname' abnormally terminated because the command was entered from an MVS console and a required parameter was not supplied with the command.

csect-name
>   The name of the command program CSECT that issued the message.

verbname
>   The command verb as entered by the requester. The error depends on which command was entered.
>   • DISPLAY THREAD - neither the 'connection-name', LOCATION, nor LUWID was specified.
>   • RECOVER INDOUBT - neither 'connection-name' nor LUWID was specified.

pkwname
>   The command primary keyword as entered by the requester.

System action: Command processing terminates.

Operator response: Reenter the command specifying one of the required parameters.

DSNV414I   THREAD *xxx* **COMMIT SCHEDULED**

Explanation: In this message, 'xxx' is the recovery identifier specified with the -RECOVER INDOUBT command to identify the indoubt thread and is one of the following:

ID=   The recovery correlation-id specified with the command.

NID=   The recovery network-id specified with the command.

LUWID =
          Luwid=token indicates the luwid and token that identify the thread specified with the command.

System action: The thread specified on the -RECOVER INDOUBT command is scheduled for COMMIT recovery action.

DSNV415I   THREAD *xxx* **ABORT SCHEDULED**

Explanation: In this message, 'xxx' is the recovery identifier specified with the -RECOVER INDOUBT command to identify the indoubt thread and is one of the following:

ID=   The recovery correlation-id specified with the command.

NID=   The recovery network-id specified with the command.

LUWID =
          The luwid=token indicates the luwid and token that identify the thread specified with the command.

The thread specified on the -RECOVER INDOUBT command is scheduled for ABORT recovery action.

System action: Command processing continues.

DSNV416I   THREAD NID=*net-node.number* **IS INVALID**

Explanation: In this message, 'net-node' is the recovery 'network-id' specified with the command.

The -RECOVER INDOUBT command (NID=net-node.number) determined that the input format for the specified network-id is invalid.

System action: Command processing continues.

Operator response: Prior to reentering the command, ensure that the network-id entered is in the correct format as specified on the -RECOVER INDOUBT command.

DSNV417I   THREAD *xxx* **NOT FOUND**

Explanation: In this message, 'xxx' is the recovery identifier specified with the -RECOVER INDOUBT command to identify the indoubt thread and is one of the following:

ID=   The recovery correlation-id specified with the command

NID=   The recovery network-id specified with the command

LUWID =
          The luwid or token specified with the command

The -RECOVER INDOUBT command was unable to locate the specified thread to be scheduled for recovery. Either the thread identifier is incorrect, or the thread no longer is in an indoubt state.

System action: Command processing continues.

Operator response: Before reentering the command, ensure that the thread still is in an indoubt state.

DSNV418I   RECOVER INDOUBT REJECTED FOR ID= *correlation-id*

Explanation: In this message, 'ID=' indicates the recovery 'correlation-id' specified with the command.

The -RECOVER INDOUBT command determined that more than one 'network-id' is associated with the specified 'correlation-id'. No recovery action is taken against the indoubt threads associated with the 'correlation-id'.

System action: Command processing continues.

Operator response: To indicate the specific indoubt thread to be scheduled for resolution, the 'network-id' associated with the thread must be specified.

DSNV424I   INACTIVE THREADS -

Explanation: This message marks the beginning of a section of output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV426I   *csect-name* **THREAD** *token* **HAS BEEN CANCELED**

Explanation: This message is issued in response to the CANCEL THREAD command. The thread is scheduled for termination.

System action: The specified thread is scheduled for termination. The termination of the thread does not occur unless processing is occurring within DB2.

DSNV427I   *csect-name* **INVALID THREAD** *token* **SPECIFIED**

Explanation: This message is issued in response to the CANCEL THREAD command. No thread with the specified *token* was found.

System action: The specified thread is not canceled.

User response: Determine the correct *token* of the thread you want to cancel and reissue the command. You can use the -DISPLAY THREAD command to display all thread activity.

DSNV428I   *csect-name* **THREAD** *token* **WAS PREVIOUSLY CANCELED**

Explanation: The thread with the specified token was previously canceled by either the CANCEL THREAD or CANCEL DDF THREAD command, or is already in the process of termination.

The CANCEL command terminates the thread only if the thread is processing in DB2 or is currently suspended by DB2. The thread might not be processing in DB2 in these situations:

- The thread is processing in the application and has not made a request to DB2. The DISPLAY THREAD command indicates this.
- The user placed the thread on an MVS resource using the ENQUE command. The CANCEL command has no effect on the thread until ENQUE completes and the thread returns to DB2.
- The thread is processing in VTAM and is waiting for a response from a partner. The DISPLAY THREAD DETAIL command indicates this. The thread does not terminate until VTAM returns to DB2. VTAM commands, which will terminate the session or LU, can be used to cause VTAM to return to DB2.

DSNV429I   **V429 CALLING PROCEDURE=***text*, **LOAD MODULE=***text*, **PROC=***text*, **ASID=***hex*, **WLM_ENV=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV431I   POSTPONED ABORT THREADS -

Explanation: This message marks the beginning of a section of output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV434I   *csect-name* **NO POSTPONED ABORT THREADS FOUND**

Explanation: This message is issued by the RECOVER POSTPONED command, automatic online backout processing (invoked by restarting with system parameter LBACKOUT=AUTO), and also by the DISPLAY THREAD TYPE(POSTPONED) command. It indicates that no postponed abort threads were found to process. The 'csect-name' is the name of the command program CSECT that issued the message.

System action: The command (or auto-resolution processing) completes normally.

Related information:
"DSNV401I" on page 942

DSNV435I   *action* **OF POSTPONED ABORT URS HAS BEEN SCHEDULED**

Explanation: This message is issued by the RECOVER POSTPONED command, and also by automatic online backout processing (invoked by restarting with system parameter LBACKOUT=AUTO). It indicates that postponed abort URs exist and their backout processing will proceed asynchronously.

ACTION
> Description of request

RESOLUTION
> Indicates that backout processing for the postponed units of recovery has been requested.

CANCELLATION
> Indicates that the CANCEL option was specified on the RECOVER POSTPONED command and that the backout process will be canceled.

System action: The command (or auto-resolution processing) completes normally.

DSNV436I   **V436-PGM=***text*, **SEC=***num*, **STMNT=***num*, **THREAD-INFO=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

| **DSNV437I**   **V437-WORKSTATION=***text* **USERID=***text* **APPLICATION NAME =***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV439I   *csect-name* **NOBACKOUT OPTION INVALID FOR THREAD** *'token'*

Explanation: This message is issued in response to the CANCEL THREAD command with the NOBACKOUT option. The NOBACKOUT option will not be honored because the canceled thread is part of a global transaction.

System action: The specified thread is not canceled.

DSNV440I   **V440-XID=***hex*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV441I   **V441-ACCOUNTING=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV442I   **V442-CRTKN=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV444I   **V444-***text***=***num* **ACCESSING DATA AT.**

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV445I   **V445-***text***=***num* **ACCESSING DATA FOR** *location*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV447I   **V447--INDEX SESSID A ST TIME**

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV448I   **V448--(***txt***)** *text t tx timestamp*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV449I   **V449-HAS NID=***text* **AND ID=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV450I   **V450-HAS PARTICIPANT INDOUBT AT** *text-address..number*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV451I   **V451-RESYNC WITH COORDINATOR STILL PENDING**

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV452I   **V452-HEURISTIC COMMIT BY PARTICIPANT AT LOCATION** *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV453I   **V453-HEURISTIC ABORT BY PARTICIPANT AT LOCATION** *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV456I   **INVALID PARAMETER WITH LUWID KEYWORD** *luwid-parameter*

Explanation: The DISPLAY THREAD TYPE(ACTIVE), DISPLAY THREAD TYPE(INACTIVE), or the RECOVER INDOUBT command was issued with the keyword LUWID and its required parameter list. However, one of the parameters was syntactically invalid as a logical-unit-of-work identifier.

System action: The command returns without processing.

Operator response: Reenter the command with valid LUWID values.

Related reference:

⤷ -DISPLAY THREAD (DB2) (DB2 Commands)

⤷ -RECOVER INDOUBT (DB2) (DB2 Commands)

DSNV457I   **V457-RESYNC PROTOCOL ERROR WITH PARTICIPANT** *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV458I   V458-CONNECTION-NAME *text* CORRELATION-ID *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV459I   V459-SYNCPOINT PROTOCOL ERROR WITH COORDINATOR

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV460I   V460-SYNCPOINT PROTOCOL ERROR WITH PARTICIPANT *location*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV461I   V461-RESYNC WITH COORDINATOR SHOWS HEURISTIC DAMAGE

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV462I   V462-COLD START BY PARTICIPANT AT LOCATION *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV463I   V463-COORDINATOR HAS COLD STARTED

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV464I   V464-RESYNC PROTOCOL ERROR WITH COORDINATOR

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV465I   V465-THREAD HAS REMAINED IN A PREPARED STATE FOR *timestamp*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV466I   V466-THREAD HAS BEEN INDOUBT FOR *timestamp*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV467I   V467-HAS LUWID *text=num*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV471I   V471-*luwid=token*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV473I   **ACTIVE THREADS FOUND FOR MEMBER:** *memname-*

Explanation: In the DISPLAY THREAD output, message DSNV473I precedes the list of active threads found on another member of a data sharing group. This output is part of message "DSNV401I" on page 942.

DSNV474I   **INDOUBT THREADS FOUND FOR MEMBER:** *memname-*

Explanation: In the DISPLAY THREAD output, message DSNV474I precedes the list of indoubt threads found on another member of a data sharing group. This output is part of message "DSNV401I" on page 942.

DSNV475I   **INACTIVE THREADS FOUND FOR MEMBER:** *memname-*

Explanation: In the DISPLAY THREAD output, message DSNV475I precedes the list of inactive threads found on another member of a data sharing group. This output is part of message "DSNV401I" on page 942.

DSNV476I   **POSTPONED ABORT THREADS FOUND FOR MEMBER:** *memname*

Explanation: In the DISPLAY THREAD output, message DSNV476I precedes the list of postponed abort units of recovery found on another member of a data sharing group. This output is part of message "DSNV401I" on page 942.

DSNV480I   **V480-DB2 IS PARTICIPANT FOR RRS URID=***hex*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV481I   **V481-DB2 IS COORDINATOR FOR RRS URID=***hex*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV482I   **V482 WLM-INFO=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV485I   **V485-TRUSTED CONTEXT=***text***, SYSTEM AUTHID=***text***, ROLE =***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV490I   **V490- SUSPENDED** *timestamp smodule soffset smodptf-level*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV492I   **V492-LONG** *num***K 64VLONG** *num***K 64LONG** *num***K**

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV497I   **SYSTEM THREADS -**

Explanation: This message marks the beginning of a section of output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV501I   COMMAND EXECUTING: *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV502I   V502-PLOCK ENGINE *text* **EXIT, RMID=***text***,FCODE=***texttext*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV503I   SYSTEM THREADS FOUND FOR MEMBER: *memname*

Explanation: In the DISPLAY THREAD output, message DSNV503I precedes the list of system threads found on another member of a data sharing group. This output is part of message "DSNV401I" on page 942.

DSNV504I   V504-TERMINATING CORRID=*text***, CONNID=** *text***, HASID=** *text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV505I   V505-BM PROCESSING DBNAME=*dbname***, SPACE NAME=** *spname***, PART=** *partno*

Explanation: The system thread that is identified in the DSNV402I message, which is issued before this message is issued, is experiencing a long wait reading or writing data for the specified database name, space name *spname*, and part. For non-partitioned data sets, *partno* will not be available. Detailed suspend data can be found in message DSNV490I.

DSNV507I   V507- *active, inactive* **MONITOR, INTERVALS=***monintervals***, STG=***monstg***, BOOSTS=***monboosts***,
               HEALTH=***text*

Explanation: This message contains detail output from the DISPLAY THREAD command, which is part of message "DSNV401I" on page 942.

DSNV508I   DSNVMON - DB2 DBM1 BELOW THE BAR STORAGE *alert-level*

Explanation: The DBM1 address space is consuming a large amount of available 31-bit, below-the-bar storage.

alert-level
          An indication of severity, either NOTIFICATION or WARNING.

The message is followed by information about current resource availability:

```
DSNVMON - DB2 DBM1 BELOW THE BAR STORAGE
          alert-level
        total-consumption% CONSUMED
        db2-consumption% CONSUMED BY DB2
        storage-availableM AVAILABLE OUT OF
          REGION SIZE region-sizeM
        WITH A cushion-sizeM STORAGE CUSHION
```

total-consumption
          The percentage of below-the-bar storage that is in use.

db2-consumption
          The percentage of below-the-bar storage that is in use by DB2.

storage-available
          The amount of available storage, measured in megabytes.

region-size
          The region size, measured in megabytes.

cushion-size
          Amount of storage reserved for use by MVS and critical processes.

System action: Processing continues.

Operator response: Take the appropriate action based on the *alert-level* value in message DSNV508I.

NOTIFICATION
> Monitor workload on the system and look for instances of message DSNV508I where *alert-level* is WARNING. Consider reducing the workload on the system.

WARNING
> Reduce the workload on the system. If storage consumption continues to increase, performance might degrade and non-critical thread processing might abend.

DSNV509I   **DSNVMON - DB2** *monitor-name* **INTERNAL MONITOR STOPPING**

Explanation: An internal DB2 monitor task is terminating due to errors.

monitor-name
> The address space in which the monitor task was running.

System action: DB2 terminates with reason code 00E50731.

Administrator response: This message is for serviceability. However, the stop is likely caused by a series of other abends. For those abends, you should provide LOGREC data and any dumps from the DB2 subsystem to IBM Software Support.

Related concepts:

Retention of DB2 dump data sets (Collecting data)

Related tasks:

Preserving LOGREC data (Collecting data)

Related information:

00E50731 (DB2 Codes)

DSNV510I   **DSNVMON - BEGIN DISPLAY OF DBM1 LARGEST STORAGE CONSUMERS**

Explanation: A summary of the agents that consume the largest amount of DBM1 31-bit, below-the-bar storage follows, in a series of DSNV512I messages.

System action: Processing continues.

DSNV511I   **DSNVMON - END DISPLAY OF DBM1 LARGEST STORAGE CONSUMERS**

Explanation: A summary of the agents that consume the largest amount of DBM1 31-bit, below-the-bar storage precedes this message.

System action: Processing continues.

DSNV512I   **DSNVMON - AGENT** *agentnumber***:**
```
NAME ST A  REQ ID AUTHID PLAN ASID TOKEN
---- -- -  --- -- ------ ---- ---- -----
f1   f2 f3 f4  f5 f6      f7   f8   f9
LONG ctlongstgK VLONG ctvlongstgK 64BIT ct64bstgK
```

Explanation: The agent that is described in this message is the *agentnumber* consumer of 31-bit, below-the-bar storage in the DBM1 address space. Agent identification fields are:

Field label	Field value
f1	name
f2	status
f3	act-id
f4	req-ct
f5	corr-id
f6	auth-id

Field label	Field value
f7	*pname*
f8	*asid*
f9	*token*

See DSNV401I for an explanation of *name*, *status*, *act-ind*, *req-ct*, *corr-id*, *auth-id*, *pname*, *asid*, and *token*. All other fields are for use by IBM Software Support.

Related information:

"DSNV401I" on page 942

DSNV516I   *csect-name* **- BEGINNING STORAGE CONTRACTION MODE**

Explanation: DB2 has detected paging and begins to free real frames for unused cached virtual storage pages that are owned by DB2.

csect-name
>   The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

DSNV517I   *csect-name* **- ENDING STORAGE CONTRACTION MODE**

Explanation: DB2 has detected paging has ceased and begins to cache unused virtual pages.

csect-name
>   The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

DSNV519I   *csect-name* **CANCEL THREAD COMMAND WITH FORCE OPTION FOR** *token-id* **HAS COMPLETED WITH RETURN CODE** *return-code*

Explanation: The CANCEL THREAD command has been issued with the FORCE option. The *return-code* indicates the results of the command.

csect-name
>   The name of the control section that issued the message.

token-id   Either a thread identifier or a logical unit of work identifier (LUWID) returned from the DISPLAY THREAD command.

return-code
>   A numeric value that indicates the success or failure of the CANCEL THREAD command.

>   **X'0000**   The CANCEL THREAD command successfully completed. The *token-id* was cancelled.

>   **X'0001**   The CANCEL THREAD command was not accepted. The FORCE option is not allowed until a CANCEL THREAD without the FORCE option is first attempted.

>   **X'0002**   The CANCEL THREAD command was not accepted. The CANCEL THREAD command with the FORCE option for the same *token-id* cannot be repeated.

>   **X'0003**   The CANCEL THREAD command was not accepted. The *token-id* cannot be found.

>   **X'0004**   The CANCEL THREAD command was not accepted. The *token-id* is associated with a DDF disconnected Database Access Thread (DBAT) on the DB2 server.

System action: The command completes processing as indicated by the *return-code*.

System programmer response: Take the action that corresponds to the *return-code* value:

X'0000   The command was success, no action is required.

| **X'0001** Issue the CANCEL THREAD command without the FORCE option.

| **X'0002** Wait to determine if the original CANCEL THREAD completes successfully.

| **X'0003** Use the DISPLAY THREAD command to identify a DDF related Database Access Thread (DBAT) on the DB2
| server. Ensure the level of z/OS in your DB2 environment supports function to terminate a preemptable
| SRB. Then reissue the command.

| **X'0004** Use the DISPLAY THREAD command to identify a DDF related Database Access Thread (DBAT) on the DB2
| server. Then reissue the command.

| **Related reference**:

| ▪ -CANCEL THREAD (DB2) (DB2 Commands)

| ▪ -DISPLAY THREAD (DB2) (DB2 Commands)

| **DSNV520I**   **V520-AUTONOMOUS PROCEDURE INVOKED BY THREAD WITH TOKEN =** *num*

| **Explanation:** This message contains detail output from the DISPLAY THREAD command, which is part of message
| "DSNV401I" on page 942.

Chapter 18. DSNW messages

Message identifiers that begin with DSNW communicate information about the instrumentation facility.

DSNW001I   ASYNCHRONOUSLY GATHERED DATA IS BEING FORMATTED IN THIS DUMP

Explanation: The subsystem formatting exit is not using DB2 summary dump records for formatting. The formatted control blocks may not contain the same values as they did at the time of the error.

System action: Dump formatting continues.

System programmer response: If you want DB2 summary dump records to be used, do not specify the 'SUMDUMP=NO' operand on the dump control statement.

DSNW002I   SUMMARY DUMP RECORDS ARE BEING FORMATTED IN THIS DUMP

Explanation: The subsystem formatting exit is using DB2 summary dump record information to format its control blocks.

System action: Dump formatting continues.

System programmer response: If you do not want DB2 summary dump records to be used in formatting, specify the 'SUMDUMP=NO' and 'SUBSYS=subsystem name' on the dump control statement. Both operands are required.

DSNW004I   ONE OR MORE INVALID OPERANDS WERE FOUND ON THE DSNWDMP CONTROL CARD. SUBSYSTEM FORMATTING TERMINATED

Explanation: An invalid operand was found on the dump control statement. The operand was not 'SG', 'LG', 'SA', 'AA', or 'ALL'.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: Correct the dump control statement operand specified by message DSNW007I.

DSNW005I   DUMP FORMATTER WORK BUFFER (DSNWDBUF) CANNOT BE LOCATED

Explanation: The subsystem formatting exit could not locate its work buffer.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: Increase the region size.

DSNW006I   THE ERLY BLOCK CANNOT BE ACCESSED OR IT IS INVALID

Explanation: The subsystem formatting exit could not locate its anchor block.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: Specify 'SUBSYS=subsystem name', 'SUMDUMP=NO' on the dump control statement. The subsystem name should have between 1 and 4 characters.

DSNW007I   THE FOLLOWING OPERAND SPECIFIED ON THE DSNWDMP CONTROL CARD IS INVALID: xxxx

Explanation: The specified operand was not one of the valid dump control statement operands.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: Check the dump control statement operands.

DSNW008I   THE SCOM CANNOT BE ACCESSED OR IT IS INVALID

Explanation: An error was encountered while trying to retrieve the SCOM.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: If 'SUMDUMP=NO' was specified on the dump control statement, omit this operand, and resubmit the job. Otherwise, specify this operand on the dump control statement, and resubmit the job.

DSNW009I   THE ADDRESS SPACE REQUESTED IS NOT AVAILABLE

Explanation: The DB2 control blocks for the address space specified on the 'SA=asid' dump operand could not be located.

System action: Formatting continues if other dump operands were specified.

System programmer response: Check the ASID specified on the 'SA=' dump operand. The ASID must be specified in hexadecimal.

DSNW010I   THE TRACE RMFT CANNOT BE ACCESSED OR IT IS INVALID

Explanation: The DB2 trace table could not be located.

System action: Formatting of the DB2 trace table is bypassed, and formatting continues of any other requested dump segment.

System programmer response: If 'SUMDUMP=NO' was specified on the dump control statement, omit this operand. Otherwise, specify this operand on the dump control statement. In either case, resubmit the job.

DSNW011I   A LARGER REGION SIZE IS REQUIRED FOR THIS JOB

Explanation: The subsystem formatting exit could not obtain a large enough work buffer to process the summary dump records.

System action: Subsystem formatting continues.

System programmer response: Rerun the job, specifying a larger region size in the JCL.

DSNW013I   DMPW NOT FOUND IN SUMMARY DUMP

Explanation: The subsystem formatting exit was unable to locate the DMPW control block in the summary record portion of the dump data set. Because the DMPW provides the main anchor block for the dump formatter, processing is terminated.

System action: Control returns to AMDPRDMP or IPCS to terminate the exit.

System programmer response: If this is not a DB2 dump data set, specify 'SUBSYS=xxxx,SUMDUMP=NO' on the dump control statement (where xxxx is the 1- to 4-character subsystem name), so that the subsystem formatting exit does not attempt to use summary dump data.

**DSNW014I   REQUIRED SUMMARY DUMP RECORDS ARE NOT IN THIS DUMP. WILL ATTEMPT TO
            FORMAT FROM NON- SUMMARY DUMP**

Explanation: Expected data could not be found in the summary dump. This message is issued for information only. Dump formatting continues. However, it will take longer, because a greater volume of data COMMAND PQUIT must be analyzed.

System action: Formatting is attempted using the full dump instead of the summary dump.

System programmer response: Though processing continues, this message is an indication of a problem, and should be pursued as such.

DSNW015I   SSCVT NOT LOCATED. CHECK THE SUBSYSTEM NAME ON THE CONTROL CARD

Explanation: In a search through the SSCVT chain, a match of the subsystem name in the SSCVTs and the subsystem name on the control statement was not found.

System action: Formatting for the named subsystem is not done.

Operator response: Check the 'SUBSYS=' operand on the control statement.

System programmer response: Correctly specify the subsystem name on the control statement.

DSNW016I   THE RMVT CANNOT BE ACCESSED OR IT IS INVALID

Explanation: The subsystem formatting exit could not locate the RMVT. The RMVT is required for formatting the DB2 trace table and a number of other DB2 control blocks.

System action: Formatting of the DB2 trace table is bypassed, and formatting of other requested dump segments continues.

System programmer response: If 'SUMDUMP=NO' was specified on the dump control statement, omit this operand. Otherwise, try specifying this operand on the dump control statement. In either case, resubmit the job.

DSNW017I   MAXIMUM STACK LEVEL EXCEEDED

Explanation: This condition is usually caused by the DB2 control block formatter looping. The stack array is depleted and can no longer accommodate control blocks.

System action: Dump formatting is terminated.

System programmer response: Examine the control block chains to see if they are looping. If those chains are not looping, increase the dimension of the stack array.

DSNW018I   SUBSYS= NOT SPECIFIED ON THE CONTROL CARD OR MISSING. REQUIRED IF SUMDUMP=NO SPECIFIED

Explanation: One of the following errors occurred:
- The 'SUMDUMP=NO' option was specified, but either the 'SUBSYS=' operand is missing, or it was incorrectly specified.
- A trace class was specified that would have resulted in an orphaned OP buffer. The trace class specified will not start, but the remainder of the classes are started.

System action: Dump formatting is terminated.

Operator response: Run the dump printing job after the error has been corrected.

DSNW020I   UNSUCCESSFUL SEARCH FOR THE ERLY CONTROL BLOCK

Explanation: The 'ERLY' control block could not be located in the dump.

System action: Dump formatting is terminated.

Operator response: Check that the 'SUBSYS=' operand was correctly specified, and rerun the job.

System programmer response: This message can mean that the 'ERLY' control block has been overlaid. Examine the full dump to determine whether the subsystem is present. If it is, you may be able to determine why the control block could not be located.

DSNW022I   THE RESIDENT TRACE WAS NOT ACTIVE AT THE TIME OF DUMP

Explanation: Trace table formatting has been attempted, but no trace table existed at the time of the dump.

System action: Dump formatting continues with any other control blocks that were to be formatted.

DSNW023I   THE TRACE TABLE ENTRY IS OUT OF SEQUENCE OR OVERLAID

Explanation: A trace entry is overlaid by another trace entry of a different timestamp. This message is issued to flag an unrecognized trace entry. This can occur when the oldest entry in the table is the second half of a double entry.

System action: Formatting of the trace table continues.

DSNW024I   TRACE TABLE

Explanation: The trace table follows.

System action: Trace table formatting follows.

DSNW025I   DUMP ACCESS ERROR ACCESSING THE TRACE TABLE IN THE DUMP

Explanation: A nonzero return code was returned from the storage access routine when accessing the trace table.

System action: Trace table formatting is bypassed.

System programmer response: Check the address and ASID of the trace table.

DSNW026I   CONTROL BLOCK SUMMARY (ALL ADDRESS SPACES)

DSNW027I   CONTROL BLOCK SUMMARY (SINGLE ADDRESS SPACE)

DSNW028I   CONTROL BLOCK SUMMARY (LONG FORM GLOBAL)

DSNW029I   CONTROL BLOCK SUMMARY (SHORT FORM GLOBAL)

Explanation: Messages DSNW026I through DSNW029I provide descriptive information regarding the type of formatting being produced.

System action: Dump formatting continues.

DSNW030I   DUMP ACCESS ERROR ACCESSING THE CONTROL BLOCK STRUCTURE TABLE IN THE DUMP

Explanation: The control block structure table cannot be accessed from the dump.

System action: Control block formatting is terminated.

Operator response: Notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNW032I   DUMP ACCESS ERROR ACCESSING ANCHOR CONTROL BLOCK

Explanation: The anchor control block cannot be accessed from the dump.

System action: Control block formatting is terminated.

Operator response: Notify the system programmer.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNW033I   BEGINNING FORMATTING

Explanation: Formatting of DB2 control blocks is beginning.

DSNW034I   TRACE TABLE AND GLOBAL BLOCKS ALREADY DUMPED

Explanation: An indicative dump is being requested. The DB2 trace table and the global blocks have already been dumped with first dump (full dump) for this abend dump (SNAP) invocation. These are, therefore, not dumped in the dump for this task.

DSNW035I   WARNING - NO TASK RELATED CONTROL BLOCKS FOR THIS TASK

Explanation: The TCB for which the dump is being requested is not IDENTIFIED to DB2. Task-related control blocks are not dumped. The DB2 trace table and global blocks are dumped only if the 'SYSABEND' DD statement is present and only if this is the first of the dump(s) (full dump) for this abend dump (SNAP) invocation.

System action: No DB2 formatting is done for the specified task.

DSNW036I   CONTROL BLOCKS FOR TASKS ASSOCIATED WITH THE ABOVE RECOVERY COORDINATOR TASK

Explanation: The formatted blocks following this message are associated with TCBs that have IDENTIFIED to DB2 with the 'recovery coordinator = no' option. These TCBs may not have invoked SNAP, but they are associated with the TCB that did.

System action: Related control blocks are dumped.

System programmer response: Examine the control blocks for relevant information.

DSNW037I   TASK RELATED CONTROL BLOCKS FOR THIS TASK

Explanation: The formatted blocks following this message are associated with the TCB that is being dumped.

System action: Relevant control blocks are dumped.

System programmer response: Examine the control blocks for relevant information.

DSNW038I   END OF FORMATTING

Explanation: Formatting of DB2 control blocks is completed.

DSNW039I   SUBSYSTEM FORMATTING COMPLETE FOR THIS DUMP

Explanation: The subsystem formatting routine has completed its processing for this dump data set.

DSNW040I   'VVV'CONTROL BLOCK FORMATTER ERROR FLAGS

- X'80': FIELD EXPECTED TO BE ZERO IS NONZERO
- X'40': LAST BLOCK IN CHAIN DOES NOT POINT BACK TO PARENT BLOCK
- X'20': BLOCK NOT FOUND IN DUMP
- X'10': BLOCK LENGTH INCORRECT
- X'08': VERIFICATION CODE INCORRECT
- X'04': BLOCK CHAINED FROM THIS BLOCK NOT FOUND IN DUMP
- X'02': BLOCK CHAINED FROM THIS BLOCK HAS WRONG VERIFICATION CODE
- X'01': VALIDATION CONSTANT DOES NOT MATCH EXPECTED VALUE

Explanation: The message DSNW040I appears at the beginning of a formatted dump. It defines the hexadecimal values associated with the 'VVV=XX' keyword. This keyword is provided on the title line of each control block selected for formatting. It indicates any unusual conditions encountered during the validation and format processing of the current control block structure.

System action: Dump formatting continues.

System programmer response: All nonzero 'XX' values should be examined. If an error is indicated, the printout of the dump should be used for the problem documentation.

Problem determination: Possible errors include:

- X'80'
- X'40'
- X'20'
- X'04'

The current state of the individual control block structures at the time of error might vary with the type of error involved. Examine control block structures with one of these 'XX' values based on the specific ABEND code.

Example: A X'04' associated with the referencing control block and the X'20' associated with the referenced control block may mean the ABEND occurred between the release of the referenced control block and re-initialization of the address within the referencing one.

DB2 validation errors include:

- X'10'
- X'08'
- X'02'
- X'01'

These values indicate that a validation error occurred while formatting the associated control block.

For example, an X'08' associated with the referenced control block indicates the storage area has been overlaid or the address in the referencing control block is incorrect. Subsequent examination of the values in this control block is not reliable.

DSNW041I   THE TAB CANNOT BE ACCESSED OR IT IS INVALID

Explanation: The DB2 trace table anchor block could not be located.

System action: Formatting of the DB2 trace table is bypassed, and formatting of any other requested dump segment continues.

System programmer response: If 'SUMDUMP=NO' was specified on the dump control statement, omit this operand. Otherwise, specify this operand on the dump control statement. In either case, resubmit the job.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNW049I   OLDEST SLOT ADDRESS INVALID, FORMATTING TRACE TABLE FROM FIRST ENTRY

Explanation: The next trace table entry location is invalid.

System action: Dump formatting continues.

System programmer response: The printout of the dump should be used for the problem documentation.

Problem determination: The Trace Anchor Block has been initialized incorrectly, or overlaid during execution.

DSNW050I *ssid*,**NO SDWA/LOGREC, ABND=***comp-reason*,**U=***user-id*, **M=***module*,**C=XYR00.***vrm.comp-function*

Explanation: This message provides the default SVC dump title (SDUMP) associated with the SYS1.DUMP data set when an SDWA was unavailable during recovery processing. The variable fields are:

ssid   4-character DB2 subsystem name

comp-reason
         Abend completion code, followed by the abend reason code.

user-id   Authorization-ID for the individual subsystem user.

module   Functional recovery routine responsible for the dump.

vrm.comp-function
         Component-ID, as follows:

   vrm   DB2 version, release number, and modification level.

   comp-function
            Component-ID function.

System action: Dump processing continues.

System programmer response: Since the SDWA provides important diagnostic information to assist in problem determination, examine the recovery environment at time of error to determine why an SDWA was not provided for this abend.

In a nonrecovery environment, there may be valid reasons for the lack of an SDWA.

Problem determination: In a recovery environment, functional recovery routines (FRRs) are guaranteed an SDWA by Recovery Termination Manager (RTM). Therefore, the recovery routine is most likely an ESTAE recovery routine. The primary reason for an SDWA not being provided to an ESTAE routine is insufficient storage available during recovery processing. Examine the region sizes allocated to the failing function to ensure that sufficient storage is available.

In a nonrecovery environment, no SDWA is provided because only RTM can create one.

DSNW051I   ERROR DURING DUMP PROCESSING

Explanation: This message is generated by the 'SDUMP dump data gathering service' recovery routine when an error is encountered during dump processing.

System action: Processing of the 'SUMLSTA' user storage areas is terminated, an SVC dump is requested, and control is returned to RTM.

System programmer response: This error is documented in a SYS1.LOGREC record. This message can be issued because of an error in the invocation of SDUMP, or because of an error in SDUMP itself, or during control block examination and access.

**DSNW052I   ERROR ENCOUNTERED DURING LOAD OR VALIDATION OF A CONTROL BLOCK
         STRUCTURE TABLE MODULE**

Explanation: The DB2 dump formatting facility cannot be used to format control blocks. An error occurred during the DB2 startup process while attempting to LOAD one of the Control Block Structures Table (CBDS) modules from the DB2 subsystem program library.

System action: Subsystem startup processing continues.

Operator response: If intervention is needed, notify the system programmer.

System programmer response: If intervention is desired, the DB2 subsystem can be terminated by either an MVS CANCEL command or a -STOP DB2 command. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: The modules must reside in a DB2 program library named on the execution procedure used to start a DB2 subsystem address space.

The named modules prohibit the use of the DB2 dump formatting facility to format SVC dumps that occur during

the current execution cycle of the DB2 subsystem. The named modules are not required for DB2 execution.

DSNW053I   VRA DIAGNOSTIC INFORMATION REPORT

Explanation: All DB2 diagnostic information in the Variable Recording Area (VRA) at the time of the error is extracted and displayed in this report.

The address of the SDWA, the contents of SDWARA, and SDWARA's hexadecimal offset into SDWA are also displayed. Individual diagnostic entries are identified by a hexadecimal VRA 'key', followed by the length of the subsequent data. Data fields are displayed in hexadecimal and EBCDIC formats. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

System action: Dump format processing continues.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNW054I   NO VRA DATA RECORDED IN SDWA

Explanation: The SDWA obtained from the SYS1.DUMP data set contained no diagnostic information in the VRA.

System action: VRA report generation is bypassed, dump format processing continues.

System programmer response: If an error is indicated, the printout of the dump should be used for the problem documentation. If VRA data is needed, the abending DB2 subcomponent has failed to supply it. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNW055I   FORMAT ROUTINE UNABLE TO LOCATE SDWA

Explanation: The MVS Summary Dump Data Access Service Routine (IEAVTFRD) was unable to locate the SDWA in the summary data portion of the SYS1.DUMP data set.

System action: VRA report generation is terminated and dump format processing continues.

System programmer response: Take the appropriate action, as indicated in the next section. If an error is indicated, use the printout of the dump for problem documentation.

Problem determination: The SDWA is placed in the summary data portion of the SYS1.DUMP data set by AMDPRDMP or IPCS. It indicates either CSA or Local Storage Residency.

If the SDWA is not in the dump data set, no error is indicated.

DSNW056I   VRA DIAGNOSTIC REPORT COMPLETE

Explanation: The VRA Diagnostic Report is done.

System action: Dump format processing continues.

DSNW060I   BEGIN SAVE AREA TRACE

Explanation: This message identifies the start of the DB2 register Save Area Trace report which appears in the formatted section of a DB2 SVC Dump. The save areas for the current failing agent execution block (EB), and all associated Agent EBs, are traced from the point of error and displayed in order of invocation.

The format of this report is as follows:

Save Area Identification

The address and contents of each save area is displayed and identified by the invoking module.

```
Module-name  Service-level-identifier

SA    .... WD1 ....  HSA ....  LSA ....
RET .... EPA ....  R0  ....
```

```
R1  ....  R2  ....  R3  ....
R4  ....  R5  ....  R6  ....
R7  ....  R8  ....  R9  ....
R10 .... R11 .... R12 ....
```

Module-name

Identifies the module responsible for obtaining the save area as indicated in the DB2 module entry point list (MEPL) at the time of the error.

If the MEPL does not contain an entry that corresponds to the module entry point address (EPA), this field is set to:

MODULE INFORMATION UNAVAILABLE

Service-level-identifier

The service level information associated with the module name is obtained from the corresponding module MEPL entry, and consists of the compilation-date and the PTF number of the latest maintenance applied.

If no maintenance has been applied for this module, this field defaults to the Function Modification Identifier (FMID).

SA   Identifies the address and contents of each word of the save area and is displayed using the format and register abbreviation conventions defined by MVS SNAP. All save area data references indicate the primary address space at the time of execution.

Execution Environment

Displays the current execution environment at the time of error for the save areas that follow. This information is repeated each time the current Agent execution EB changed during processing.

```
CURRENT EB=address JOBNAME=name
HASID=hasid PASID=pasid
EXECUTION MODE=mode
```

address

Address of the associated Agent EB.

name   The 8-character MVS jobname associated with the home address space.

hasid   Home address space identifier.

pasid   Primary address space identifier.

mode   Agent EB MVS execution mode:

TCB   Task Control Mode

SRB   Supervisory Request Mode

Register Save Area Contents for Agent EBs

The contents of the register save area chains for the current failing Agent EB, and all suspended Agent EBs at the time of error, are displayed using the save area (SA) format.

```
REGISTERS IN (CB-name) AT TIME OF
execution-status
```

CB-name

Identifies the name of the control block from which the registers for the current Agent EB were obtained:

SDWA MVS System Diagnostic Workarea (Failure).

TCB   MVS Task Control Block (Suspend).

STSV   MVS SRB Status Save Area (Suspend).

execution-status

Indicates the current execution status of the Agent EB at the time of error:

Failure Failing execution EB.

Suspend
> Suspended execution EB.

Change of Primary Address Space

Since all register data references refer to the primary address space associated with the currently active execution EB, the following information is displayed each time the primary address changed during execution:

CHANGE OF ADDRESS SPACE, NEW PASID=pasid

pasid   Indicates the new primary address space identifier associated with the save areas that follow.

System action: save area trace format processing continues for the failing agent EB, and all associated agent EBs.

System programmer response: See the Problem Determination section.

Problem determination: This report may be used to determine the register contents, module invocation sequence, and execution environments leading up to the point of error.

DSNW061I   **SAVE AREA TRACE COMPLETE**

Explanation: This message indicates that the DB2 formatted save area trace report (DSNW060I) is complete.

System action: Dump format processing continues.

DSNW062I   **R6 (R6-contents) DOES NOT CONTAIN A VALID EB ADDRESS**

Explanation: During dump format processing of the DB2 formatted Save Area Trace report (DSNW060I), register 6 (R6) did not contain the address of a valid Agent execution block (EB).

R6-contents:
> The contents of register 6.

System action: save area trace format processing is terminated for the current Agent EB, and all prior EBs.

System programmer response: The dump formatter expects the contents of register 6 to contain the address of a valid EB.

Problem determination: Register 6 does not contain the current EB address or a prior EB address.

Refer to the abend reason and completion codes associated with the original error to determine the use of register 6 prior to the failure.

DSNW063I   **name (address) ASID (asid) NOT FOUND IN DUMP**

Explanation: During dump formatter processing of the save area trace report (DSNW060I), a control block or save area was not found in the dump data set.

Since the dump formatter uses the DB2 and MVS control blocks defined under the 'name' field of this message to locate individual register save areas, subsequent save areas located via the 'named' control block or save area will not be displayed in the report.

name   Identifies the name of the control block or save area that was not found in the dump data set:

SA   Indicates a Save Area

ASCE   DB2 Address Space Control Element

EB   DB2 Execution Block

TCB   MVS Task Control Block

RB   MVS Request Block

XSB   MVS Extended Status Block

PSA   MVS Prefix Save Area

SDWA
> MVS System Diagnostic Work Area

STSV   MVS SRB Status Save Area

STKE   MVS Cross Memory Stack Element

address
         The address of the 'named' control block or save area.

asid   The address space identifier associated with the control block or save area.

System action: Register save area trace format processing for the current save area chains is terminated. Subsequent save area processing will vary depending on the specific control block or save area that was available, and the DB2 Agent execution environments at the time of the error.

System programmer response: Due to the execution structures and environmental restrictions of selected DB2 and MVS control structures, some control blocks and save areas associated with these execution environments will not be included in the dump data set.

Problem determination: During MVS/RTM recovery processing, IFC Dump Services attempts to include all control blocks (both DB2 and MVS), and the pertinent DB2 save areas in the dump data set, regardless of the type of error. Control blocks and save areas associated with the following address spaces at time of error will be included in the dump data set:

- Primary, Secondary, and Home address spaces.
- DB2 System Services and Database Services address spaces.

DSNW108I   UNABLE TO AUTOMATICALLY START *type* **TRACE**

Explanation: System parameters indicated that a DB2 trace should be started automatically during DB2 subsystem initialization, but the DB2 subsystem was unable to start the trace. A nonzero return code was received from the -START TRACE command.

System action: Subsystem initialization continues.

Operator response: Start the trace via the -START TRACE command after DB2 subsystem initialization is complete.

DSNW109I   TRACE INITIALIZATION PARAMETERS UNAVAILABLE, DEFAULTS ASSUMED

Explanation: The trace function was unable to access the trace initialization parameters defined by the DSN6SYSP macro. Default values as defined by that macro are assumed for trace parameters.

System action: Subsystem initialization continues.

Operator response: Determine if the system parameters module DSNZPARM is missing or inaccessible. If not, determine if the CSECT DSN6SYSP is missing. Trace can be started with the -START TRACE command.

DSNW120I   A SPECIFIED *keyword* **VALUE IS INVALID FOR** *trace-type* **TRACE**

Explanation: A trace command has been entered, but a specified keyword value is not valid for the trace type requested.

System action: Processing for the TRACE command is terminated.

Operator response: If a -START TRACE command was entered, specify a valid destination for the trace. Otherwise, issue a -DISPLAY TRACE command to determine what traces are currently active.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related concepts:

 Types of DB2 traces (DB2 Performance)

Related reference:

 -START TRACE (DB2) (DB2 Commands)

 -DISPLAY TRACE (DB2) (DB2 Commands)

DSNW121I   A SPECIFIED CLASS VALUE IS INVALID FOR *trace-type* **TRACE**

Explanation: A trace command has been entered, but a specified class value is not valid for the trace type requested.

System action: Processing for the TRACE command is terminated.

Operator response: If a -START TRACE command was entered, specify a valid class for the trace. Otherwise, issue a -DISPLAY TRACE command to determine what options are currently active.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related concepts:

➥ Types of DB2 traces (DB2 Performance)

Related reference:

➥ -START TRACE (DB2) (DB2 Commands)

➥ -DISPLAY TRACE (DB2) (DB2 Commands)

DSNW122I   THE KEYWORD *invalid-keyword* **IS NOT VALID FOR** *trace-type* **TRACE**

Explanation: A trace command has been entered, but *invalid-keyword* is not valid for the trace type specified.

System action: Processing for the trace command is terminated.

Operator response: Omit the specified keyword from the trace command, or specify a different type of trace.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related concepts:

➥ Types of DB2 traces (DB2 Performance)

Related reference:

➥ -START TRACE (DB2) (DB2 Commands)

➥ -MODIFY TRACE (DB2) (DB2 Commands)

DSNW123I   *csect-name* TRACE RECORDING HAS BEEN RESUMED ON *dest*

Explanation: 'dest' destination resumes acceptance of trace data after a failure.

System action: Data recording is resumed.

DSNW124I   *csect-name type* TRACE TERMINATED RC=*code* RMID=*nn*

Explanation: During processing 'type' trace, processing abended due to an error. A trace type of blank indicates all tracing has stopped. 'RMID', displayed in decimal, identifies the resource manager. 'RC', displayed in hexadecimal, specifies the return/reason/abend code associated with the action.

Further collection of the named trace is stopped. If it is necessary to resume collection of the trace, a -START TRACE command may be issued. However if another failure is experienced, the problem should be resolved before starting the trace collection again.

System action: Processing for the named trace type is stopped. The message is not externalized by the functional recovery routine, but is outputted whenever a IFC event is driven at a later time. A trace type of blank indicates all tracing has stopped.

Operator response: If necessary to collect the named trace, issue a -START TRACE command to resume processing.

User response: Investigate the reasons for the failure.

Problem determination: Obtain a copy of the system or terminal log. If a dump was produced, have it printed. Print a copy of the contents of the SYS1.LOGREC for the time period involved. Forward these items along with a copy of the system log to the system programmer.

DSNW125I   MULTIPLE ENTRIES NOT ALLOWED FOR *constraint-option* **AND** *constraint-option*

Explanation: A trace command contained multiple values for the indicated options. When used separately, both options permit multiple values. However, when two or more constraint options are used together, only one can contain multiple values.

constraint-option
> The name of the constraint option. Message DSNW125I identifies the first two constraint options that contain multiple values. The command might contain additional constraint options that contain multiple values, but they are not listed.

System action: Command processing is terminated.

Operator response: Rewrite the command so that only one of the constraint options specifies multiple values.

For example, the following command would cause message DSNW125I:

Incorrect command:
```
START TRACE ACCTG PLAN(a,b,c) AUTHID (d,e)
```

Instead, issue two commands, where each command contains a different user ID:

Correct commands:
```
START TRACE ACCTG PLAN(a,b,c) AUTHID (d)
START TRACE ACCTG PLAN(a,b,c) AUTHID (e)
```

In some cases, it might be more efficient to use the corresponding filtering option instead of the constraint option. Filtering options with multiple values can be used in conjunction with a constraint option that has multiple values. Referring to the preceding example, you could issue this command, where the command uses XAUTHID to enumerate all users other than *d* and *e*:

Correct command:
```
START TRACE ACCTG PLAN(a,b,c) XAUTHID (f,g)
```

Make sure that no other constraint options in the command contain multiple values.

Problem determination: If the cause of message DSNW125I is not immediately obvious, obtain a copy of the system or terminal log to verify the command entry.

Related reference:

➡ -START TRACE (DB2) (DB2 Commands)

➡ -DISPLAY TRACE (DB2) (DB2 Commands)

➡ -MODIFY TRACE (DB2) (DB2 Commands)

➡ -STOP TRACE (DB2) (DB2 Commands)

DSNW126I   *type* TRACE NOT ALLOWED, ACTIVE TRACE TABLE FULL

Explanation: The 'type' trace cannot be started because the active trace table is has reached the maximum number of active traces allowed.

System action: Processing for the TRACE command is terminated.

Operator response: Use the -DISPLAY TRACE command to see if an active trace could be stopped. An active trace must be stopped before any other start trace command will be processed.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

DSNW127I   CURRENT TRACE ACTIVITY IS-

Explanation: This message is the title for the -DISPLAY TRACE command output. The DSNW127I message is followed by the DSNW128I message, one or more DSNW129I messages, and the DSNW147I message.

DSNW128I   TNO TYPE CLASS DEST QUAL IFCID

Explanation: This message is issued before any -DISPLAY TRACE responses. The message follows message DSNW127I. The title indicates the trace number that is active, the type of trace, the class within type, the destination specified for the active trace entries, and the IFCIDs specified using the IFCID keyword. Whether the trace is qualified is also displayed. If QUAL=YES, the trace is qualified by authorization ID, plan, RMID, or location. If QUAL=NO, the trace is not qualified. Qualification information can be requested by specifying DETAIL(2) on -DISPLAY TRACE. Refer to messages DSNW143I, DSNW144I, DSNW145I, and DSNW148I for more information about -DISPLAY TRACE. The message number is not displayed to save space on the message line.

DSNW129I   *trace-number trace-type type-class destination qualified ifcid*

Explanation: This message is issued for -DISPLAY TRACE responses.

trace-number
> The trace number that is active.

trace-type
> The type of trace that is active.

> **AUDIT**
>> For a trace of type AUDIT, if no class was specified, then the audit policy was used.
>>
>> Message DSNW185I provides additional information about the audit policy.

type-class
> Within the trace type, the class of trace that is active.

destination
> The destination that is specified for the active trace entries.

qualified
> Indication of whether or not the trace is qualified.

> **YES**   The trace is qualified by authorization ID, plan, resource manager identifier (RMID), or location.

> **NO**   The trace is not qualified.

> To display qualification information, specify DETAIL(2) on -DISPLAY TRACE.

> Message DSNW143I also provides trace qualification information.

ifcid   The IFCIDs that were specified with the IFCID keyword.

The message follows message number DSNW128I. If more information is available than can be displayed on one line, subsequent lines of information are displayed. The message number is not displayed to save space on the message line.

System action: Processing continues.

User response: No action is required.

Related reference:

➥ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:
"DSNW128I"
"DSNW143I" on page 991
DSNW185I

DSNW130I   *type* **TRACE STARTED, ASSIGNED TRACE NUMBER** *tno*

Explanation: 'type' trace has been started and assigned the decimal trace number 'tno'. Multiple messages are possible when the start request specifies multiple plans or multiple authorization ids.

System action: Processing for the -START TRACE command will continue.

DSNW131I   **STOP TRACE SUCCESSFUL FOR TRACE NUMBER(S)** *tno***,...**

Explanation: The trace number(s), 'tno,...', have been stopped. Up to five decimal trace numbers may be listed. If more than five traces have been stopped, then another DSNW131I message is sent.

System action: Processing for the -STOP TRACE command will continue.

DSNW132I   **MODIFY TRACE SUCCESSFUL FOR TRACE NUMBER(S) tno,...**

Explanation: The trace numbers, 'tno,...', have been modified. Up to five numbers may be listed. If more than five traces have been modified, then another DSNW131I message is sent.

System action: Processing for the modify command will continue.

DSNW133I   *csect-name* **TRACE DATA LOST,** *destination* **NOT ACCESSIBLE RC=***return-code*

Explanation: The destination specified stopped accepting trace data during a trace. Some condition external to DB2 caused the data rejection.

The reason for failure is defined by the *return-code* value. The *return-code* contains one of the following values:
- The hexadecimal return code from SMF.
- The hexadecimal return code from the GTF request.

    04   GTF trace and/or USR tracing is not active.
- The hexadecimal return code from the OPn request.

    04   No OPn buffer is assigned to an application to collect data.
- The hexadecimal return code from the OPX request.

    04   No OP*n* buffer is assigned to an application to collect data.

    08   The OPX trace destination buffer is full causing data to be lost.
- The hexadecimal return code from the SRV request.

    10   The serviceability routine is absent.

    xx   The serviceability routine return code.

System action: Trace processing continues although data will be lost.

This message will also be issued if a -START TRACE command is entered from the console or from the DB2I commands panel to an unassigned OPn or OPX destination. This is because the OP destination is not activated until an application program claims the buffer. DB2 will begin writing trace records to the buffer when an application program issues a -START TRACE to this same OP buffer destination.

Operator response: Investigate the GTF or SMF facility to determine why data is not being accepted. You may want to issue a -START TRACE command to record the data at another destination. The -DISPLAY TRACE command shows what types of data were recorded at the specified destination. Message DSNW133I with RC=8 is due to exhaustion of storage in an OPx buffer.

Problem determination: Obtain a copy of the system or terminal log to verify command. The OPx buffer to which the records were to be written has been filled, and all subsequent records intended for the buffer will be lost until an IFI READA request is issued to read and clear the contents of the buffer. Suggested actions include increasing the size of the OPx buffer and/or issuing the IFI READA request more frequently so that the OPx buffer is read and cleared before buffer storage is exhausted.

Related concepts:

⬀ z/OS Generalized Trace Facility (GTF) (MVS Diagnosis: Tools and Service Aids)

Related tasks:

⬀ Gathering active log records into a buffer (DB2 Administration Guide)

⬀ Requesting data asynchronously from a monitor program (DB2 Performance)

Related reference:

⬀ -START TRACE (DB2) (DB2 Commands)

⬀ -DISPLAY TRACE (DB2) (DB2 Commands)

⬀ COMMAND (DB2 Performance)

⬀ z/OS V1R13.0 MVS System Management Facilities (SMF)

DSNW135I *type* **TRACE ALREADY ACTIVE, TRACE NUMBER** *tno*

Explanation: 'type' trace was already active with trace number 'tno'.

System action: Processing for the -START TRACE command will continue.

DSNW137I **SPECIFIED TRACE NOT ACTIVE**

Explanation: The specified trace could not be found in the active trace table.

System action: Processing for the -STOP or -DISPLAY TRACE command will continue.

Operator response: You may want to issue an unqualified -DISPLAY TRACE command to determine all the active trace entries. If no traces are active then an unqualified -DISPLAY TRACE command response message is a DSNW137I message.

DSNW138I **IFCID** *ifcid-number* **IS INVALID**

Explanation: The specified IFCID number is outside the range of valid IFCID numbers or is an IFCID number which is not allowed on a trace command.

System action: Processing of the trace command is terminated before any trace functions are performed.

Operator response: Investigate the reason for the problem.

Problem determination: Remove any non-valid IFCID values and reissue the command.

Related concepts:

⬀ DB2 trace (DB2 Performance)

Related reference:

⬀ -START TRACE (DB2) (DB2 Commands)

⬀ -MODIFY TRACE (DB2) (DB2 Commands)

⬀ Trace field descriptions (DB2 Performance)

DSNW139I **AUTHORIZATION FAILURE BECAUSE** *privilege-name* **PRIVILEGE REQUIRED FOR TRACE NUMBER(S)** *trace-number*

Explanation: The user of the command does not have the required authorization privilege to execute the command that was specified for the *trace-number* trace entries. Up to five trace numbers can appear in a single message. Multiple messages might be issued when the required authorization privilege is missing for multiple trace numbers.

privilege-name
         The name of the privilege.

trace-number
>   The trace number that requires an authorization privilege.

System action: Command processing is stopped.

Operator response: Notify the security administrator.

User response: Notify the security administrator.

Problem determination: Obtain a copy of the command input and verify the user's current authorization from the DB2 subsystem catalog.

DSNW140I   ALL OPN DESTINATIONS ARE ASSIGNED

Explanation: The request could not be processed because destinations OP1 - OP8 were assigned.

System action: Processing of the trace command is terminated before any trace functions are performed.

Operator response: Investigate the reason for the problem by doing a display trace to analyze the active traces.

User response: Investigate the reason for the failure.

Problem determination: Analyze why all the OPn destinations are active after doing a display trace.

DSNW141I   STORAGE NOT AVAILABLE FOR BUFFER SIZE *size* **K BYTES**

Explanation: The request could not be processed because storage cannot be obtained for the OPn destination storage buffer.

System action: Processing of the trace command is terminated.

Operator response: Investigate the reason for the problem by doing a display trace to analyze the active traces.

User response: Investigate the reason for the failure.

Problem determination: The problem is caused either because one or more prior start traces to an OPn destination used all the storage in the IFC OPn storage pool or because ECSA contiguous storage was not available. The IFC OPn storage pool limit is approximately 4 megabytes. When a trace is stopped the buffer storage is not free. If another start trace is issued to the same destination, the storage will be reused. If the requested storage is greater than the current unassigned buffer, it is free and a larger buffer is obtained. If the requested storage is less than the current unassigned buffer, the buffer is reused. Fragmentation can occur if either application programs or an operator specify numerous start commands for varying size buffers.

DSNW142I   SPECIFIED BUFSIZE IS INCOMPATIBLE WITH ASSIGNED BUFFER

Explanation: The request could not be processed because the start trace command specified a BUFSIZE value for an OPn destination which has a buffer already assigned with a different BUFSIZE.

System action: Processing of the trace command is terminated.

Operator response: Investigate the reason for the problem.

User response: Investigate the reason for the failure.

Problem determination: The problem may be caused by one application program attempting to start a trace to a destination owned by another program. This is a site problem and can be solved by having application programs start traces to an OPX destination rather than a specific OPn destination. If only one application program is starting traces, then the program should either specify the same BUFSIZE for all requests or should not specify BUFSIZE.

DSNW143I   CURRENT TRACE QUALIFICATIONS ARE -

Explanation: This message is issued before -DISPLAY TRACE responses when DETAIL(2) or DETAIL(*) is specified. Message DSNW143I is followed by the DSNW144I message, one or more DSNW145I messages, and the DSNW148I message.

DSNW147I   ***END OF DISPLAY TRACE SUMMARY DATA***

Explanation: This message is produced when all summary information for the -DISPLAY TRACE command has been written. Summary information is written when the DETAIL keyword is specified as '1', '*', or not specified at all. Message DSNW147I follows one or more DSNW129I messages. The message number is not displayed to save space on the message line.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNW148I   ***END OF DISPLAY TRACE QUALIFICATION DATA***

Explanation: This message is produced when all qualification data for the -DISPLAY TRACE command has been written. Qualification data is written when the DETAIL keyword of the -DISPLAY TRACE command is specified as '2', or '*'. Message DSNW148I follows one or more DSNW145I messages. The message number is not displayed to save space on the message line.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNW149I   STOP TRACE COMMAND WAS UNSUCCESSFUL BECAUSE MONITOR TRACE CLASS 6 IS ACTIVE FOR QUALIFYING TRACE NUMBER(S) *class-number***,...**

Explanation: No traces were stopped by the -STOP TRACE command. Either -STOP TRACE(*) or -STOP TRACE(MON) CLASS(*) was issued when monitor trace class 6 was active. An asterisk (*) is not allowed for the TRACE and CLASS keywords when monitor trace class 6 is active. The trace numbers, *class-number*, indicate the traces that met any other specified -STOP TRACE qualifying criteria (PLAN, AUTHID, RMID, LOCATION, TNO) and also activated monitor trace class 6, thereby preventing all qualifying traces from being stopped. This special handling of the asterisk on the -STOP TRACE command for monitor trace class 6 reduces the possibility of accidentally stopping monitor trace class 6, which enables and disables DB2 data propagation.

System action: No traces are stopped. Processing of the -STOP TRACE command is terminated.

Operator response: If you do not want DB2 data propagation capability, you can stop monitor trace class 6 to allow full use of the asterisk. First stop the traces indicated by the trace numbers, *class-number*, and then reissue the command.

If you want DB2 data propagation capability, do not stop monitor trace class 6. You must specify the trace TYPE. You can stop traces without specifying the trace CLASS as long as the trace TYPE is not (MON). If the trace TYPE is (MON), then you must specify the CLASS keyword explicitly (for example, -STOP TRACE(MON) CLASS(1,2)).

User response: Reissue the -STOP TRACE command without using an asterisk for the trace type. You can use a asterisk for the CLASS keyword to stop all trace types except the monitor (MON) trace.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Related concepts:

➡ Monitor trace (DB2 Performance)

Related reference:

➡ -STOP TRACE (DB2) (DB2 Commands)

DSNW150I   **EXCLUDE FOR ALL** *keyword* **VALUES IS NOT ALLOWED**

Explanation: You have entered a start trace command with an invalid value for the X*keyword* keyword.

System action: Processing the -START TRACE command is terminated.

Operator response: Enter valid values for the X*keyword* keyword on a -START TRACE command. Refer to the DB2 Command Reference for details about valid X*keyword* keyword values.

DSNW151I   **ERROR CONVERTING** *keyword* **VALUE TO UTF-8**

Explanation: A START, STOP, or DISPLAY TRACE command has failed because the value specified in the keyword could not be converted to UTF-8.

System action: Processing the command is terminated.

Operator response: Re-enter the command. If the failure persists, contact DB2 Support.

DSNW152I   **BEGIN TNO** *trace-number* **QUALIFICATIONS**

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message indicates that the trace qualifications for the specified trace number specified are to follow.

DSNW153I   **END TNO** *trace-number* **QUALIFICATIONS**

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message indicates that the trace qualifications for the specified trace number have been displayed above.

DSNW154I   **NO QUALIFICATIONS**

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message appears between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has no trace qualifications.

DSNW155I   **PLAN:** *plan-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has a *plan-qualification*.

DSNW156I   **AUTHID:** *authid-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has an *authid-qualification*.

DSNW157I   **LOCATION:** *location-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has a *location-qualification*.

DSNW158I   USERID: *userid-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has a *userid-qualification*.

DSNW159I   APPLICATION/TRANSACTION NAME: *application/transaction-name-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has an *application/transaction name qualification*.

DSNW160I   WORKSTATION: *workstation-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has a *workstation-qualification*.

DSNW161I   PACKAGE LOCATION: *package-location-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has a *package-location-qualification*.

DSNW162I   PACKAGE COLLECTION ID: *package-collection-ID-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has a *package-collection-ID-qualification*.

DSNW163I   PACKAGE PROGRAM NAME: *package-program-name-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has a *package-program-name-qualification*.

DSNW164I   PACKAGE COLLECTION ID: *exclude-keywords*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has *exclude-keywords* specified.

DSNW165I   RMID: *rmid-qualifications*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has *rmid-qualifications* .

DSNW166I   *qualification-overflow*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear immediately after a DSNW155I thru DSNW168I message, if the preceding message is greater than 70 characters. Multiple DSNW166I messages appear until the entire qualification has been displayed.

DSNW167I   CONNECTION ID: *connection-ID-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has a *connection-ID-qualification*.

DSNW168I   CORRELATION ID: *correlation-ID-qualification*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has a *correlation-ID-qualification* .

DSNW169I   THE MAXIMUM OF 16 TRACE QUALIFICATIONS FOR A TRACE HAS BEEN EXCEEDED.

Explanation: A START, STOP, OR DISPLAY trace command has been entered that would result in greater than 16 qualifications for a trace entry.

System action: Processing of the command is terminated.

Operator response: Reduce the number of trace qualifications specified. Refer to the DB2 Command Reference for details about trace qualifications.

DSNW170I   EXCLUDE PLAN: *xplan qual1*,*xplan qual2*,...,*xplan qual8*

Explanation: This message is issued as a part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and dSNW153I message if the trace number specified in the DSNW125I message has an XPLAN qualification.

DSNW171I   EXCLUDE AUTHID: *xauthid qual1*,*xauthid qual2*,...,*authid qual8*

Explanation: This message is issued as a part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between messages DSNW152I and DSNW153I if the trace number specified in the DSNW125I message has an XAUTHID qualification.

DSNW172I   EXCLUDE LOCATION: *xloc qual1*,*xloc qual2*,...,*xloc qual8*

Explanation: When you specify a DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XLOC qualification.

DSNW173I   EXCLUDE USERID: *xuserid qual1*,*xuserid qual2*,...,*xuserid qual8*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I has an XUSERID qualification.

DSNW174I   EXCLUDE APP/TXACT NAME: *xappname qual1*,*xappname qual2*,...,*xappname qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XAPPNAME qualification.

DSNW175I   EXCLUDE WORKSTATION: *xwrkstn qual1*,*xwrkstn qual2*,...,*xwrkstn qual8*

Explanation: This message is issued as a part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XWRKSTN qualification.

DSNW176I   EXCLUDE PACKAGE LOC: *xpkgloc qual1*,*xpkgloc qual2*,...,*xpkgloc qual8*

Explanation: This message is issued as part of the -DISPLAY TRACE output when DETAIL(2) is specified. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XPKGLOC qualification.

DSNW177I   EXCLUDE PACKAGE COLLID: *xpkgcol qual1*,*xpkgcol qual2*,**...**,*xpkgcol qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XPKGCOL qualification.

DSNW178I   EXCLUDE PACKAGE PROGNAME: *xpkgprog qual1*,*xpkgprog qual2*,**...**,*xpkgprog qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XPKGPROG qualification.

DSNW179I   EXCLUDE CONNECTION ID: *xconnid qual1*,*xconnid qual2*,**...**,*xconnid qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XCONNID qualification.

DSNW180I   EXCLUDE CORRELATION ID: *xcorrid qual1*,*xcorrid qual2*,**...**,*xcorrid qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XCORRID qualification.

DSNW181I   EXCLUDE ROLE: *xrole qual1*,*xrole qual2*,**...**,*xrole qual8*

Explanation: When you specify DETAIL(2), this message is issued as part of the -DISPLAY TRACE output. This message will appear between a DSNW152I and DSNW153I message if the trace number specified in the DSNW125I message has an XROLE qualification.

DSNW182I   *csect* **AN INTERNAL STOP HAS BEEN ISSUED FOR TRACE NUMBER** *traceno* **BECAUSE THE OP BUFFER OWNER HAS TERMINATED**

Explanation: This message is issued when the thread that owns the OP buffer terminates. DB2 has stopped the trace that you specified because the trace was started with exclusively OP buffer destinations. OP buffer destinations require a valid owning thread. DB2 may also issue an internal stop for traces started with a combination of OP and non-OP destinations. This occurs when an internal modify results in a duplicate trace entry.

DSNW183I   *csect* **AN INTERNAL MODIFY HAS BEEN ISSUED FOR TRACE NUMBER** *traceno* **BECAUSE THE OP BUFFER OWNER HAS TERMINATED. TRACE NUMBER** *traceno* **REMAINS ACTIVE USING DESTINATIONS** *dest1, dest2, ...*

Explanation: This message is issued when the thread that owns the OP buffer terminates. The trace *traceno* remains active using only the destinations specified. OP buffer destinations have been stopped because these destinations require a valid owning thread.

DSNW184I   *csect* **OP BUFFER** *opbuff* **NOT AVAILABLE**

Explanation: This message is issued when a -START TRACE command has DEST(OP1-8) specified. The thread issuing the -START TRACE command does not currently own an in use buffer *opbuff* or the values in IFCAOWNER, IFCAGLBL, and WBUFECB do not match the values that are specified by previously started traces.

System action: Processing for the -START TRACE command is terminated.

Operator response: Change the DEST keyword to one of the following:

- OPX
- an OPX buffer that is owned by the current thread
- an OP buffer that is not in use

DSNW185I   BEGIN TNO *trace-number* **AUDIT POLICIES**

Explanation: This message is issued as part of the -DISPLAY TRACE responses if DETAIL(2) is specified. This message indicates the start of the output and is followed by the audit policies for the specified trace number.

trace-number
        The number assigned to the trace.

Output from the command begins with this message and ends with message DSNW186I.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:
"DSNW186I"

DSNW186I   END TNO *trace-number* **AUDIT POLICIES**

Explanation: This message is issued as part of the -DISPLAY TRACE responses if DETAIL(2) is specified, and indicates the end of the output from the command.

trace-number
        The number assigned to the trace.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:
"DSNW185I"

DSNW187I   AUDIT POLICY: *policy-name*

Explanation: This message is issued as part of the -DISPLAY TRACE output if DETAIL(2) is specified and if the trace number that is indicated in the *trace-number* value from message DSNW185I has a policy name.

policy-name
        The name of the audit policy that is active.

Output from the command begins with message DSNW185I and ends with message DSNW186I.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:
"DSNW185I"

DSNW188I   NO AUDIT POLICIES ACTIVE

Explanation: This message is issued as part of the -DISPLAY TRACE output if DETAIL(2) is specified and if the trace number that is indicated in the *trace-number* value from message DSNW185I has no active audit policies.

Output from the command begins with message DSNW185I and ends with message DSNW186I.

System action: Processing continues.

User response: No action is required.

Related reference:

▶ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:

"DSNW185I" on page 997

DSNW189I   THE AUDTPLCY KEYWORD IS NOT VALID WITH CLASS KEYWORD

Explanation: A trace command has been issued with conflicting keywords. The AUDTPLCY keyword cannot be specified with the CLASS keyword.

System action: Processing for the trace command is terminated.

Operator response: Correct the syntax, and reissue the command

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related reference:

▶ -START TRACE (DB2) (DB2 Commands)

▶ -DISPLAY TRACE (DB2) (DB2 Commands)

▶ -STOP TRACE (DB2) (DB2 Commands)

DSNW190I   THE AUDITPOLICY KEYWORD IS NOT VALID WITH IFCID OR CLASS KEYWORD IS NOT NEEDED.

Explanation: A trace command has been entered, but the AUDITPOLICY keyword cannot be specified with the IFCID keyword.

System action: The processing for the trace command is terminated.

Operator response: Correct the syntax, and reissue the statement.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related reference:

▶ -START TRACE (DB2) (DB2 Commands)

▶ -DISPLAY TRACE (DB2) (DB2 Commands)

▶ -STOP TRACE (DB2) (DB2 Commands)

DSNW192I   AUDIT POLICY SUMMARY

Explanation: This message is issued as part of the -START TRACE output when AUDTPLCY is specified. This message indicates the start of the output and is followed by the summary of audit policies that started successfully.

Output from the command begins with this message and ends with message DSNW193I.

System action: Processing continues.

User response: No action is required.

Related reference:

▶ -START TRACE (DB2) (DB2 Commands)

Related information:

"DSNW193I" on page 999

DSNW193I   END AUDIT POLICY SUMMARY

Explanation: This message is issued as part of the -START TRACE output when AUDTPLCY is specified, and indicates the end of the output from the command.

System action: Processing continues.

User response: No action is required.

Related reference:

➥ -START TRACE (DB2) (DB2 Commands)

Related information:
"DSNW192I" on page 998

DSNW194I   AUDIT POLICY *policy-name* **STARTED**

Explanation: This message is issued as part of the -START TRACE output when AUDTPLCY is specified if an audit policy was started successfully.

policy-name
        The name of the audit policy that is active.

Output from the -START TRACE command begins with message DSNW192I and ends with message DSNW193I.

System action: Processing continues.

User response: No action is required.

Related reference:

➥ -START TRACE (DB2) (DB2 Commands)

Related information:
"DSNW192I" on page 998

DSNW195I   NO AUDIT POLICY STARTED SUCCESSFULLY

Explanation: A trace command has been entered to start audit policies, but audit policies cannot be started.

System action: The processing for the trace command is terminated.

The IFCID 362 trace record contains information about the failure.

Problem determination: Obtain a copy of the IFCID 362 trace record from the destination that is specified in the START TRACE command.

DSNW196I   AUDIT POLICY *policy-name* **NOT STARTED. REASON** *reason-code*

Explanation: This message is issued as part of the -START TRACE output when AUDTPLCY is specified and at least one of the specified audit policies cannot be started.

Output from the command begins with message DSNW192I and ends with message DSNW193I.

policy-name
        The name of the audit policy

reason-code
        A DB2 reason code that indicates the cause of the failure.

System action: The processing for the trace command continues.

Operator response: Refer to the IFCID 362 trace record for additional information about the audit policy that was not started.

Problem determination: Obtain a copy of the IFCID 362 trace record from the destination that is specified in the START TRACE command.

Related concepts:

⮕ DB2 reason codes (DB2 Codes)

Related reference:

⮕ -START TRACE (DB2) (DB2 Commands)

Related information:

"DSNW192I" on page 998

DSNW198I   DISPLAY TRACE OUTPUT TO BE CONTINUED (CMDID=_command-id_**)**

Explanation: A DISPLAY TRACE command output directed to an MVS console must be continued in another group of messages beginning with DSNW199I. The related DSNW199I message will have an identical CMDID value.

command-id
        The identifier for the command.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:

"DSNW199I"

DSNW199I   DISPLAY TRACE OUTPUT CONTINUED (CMDID=_command-id_**)**

Explanation: A DISPLAY TRACE command output directed to an MVS console is continued. This output is linked to the group of DISPLAY TRACE messages that end with a DSNW198I message with the same CMDID value.

command-id
        The identifier for the command.

System action: Processing continues.

User response: No action is required.

Related reference:

⮕ -DISPLAY TRACE (DB2) (DB2 Commands)

Related information:

"DSNW198I"

Chapter 19. DSNX messages

Message identifiers that begin with DSNX communicate information about the relational data system.

DSNX100I   **bind-type SQL WARNING USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number object name IS NOT DEFINED**

Explanation: The indicated object is not defined in the DB2 catalog.

bind-type
    Type of BIND subcommand that caused the warning: BIND, REBIND

authorization-id
    Authorization ID used during the BIND process: for a BIND subcommand, indicates the authorization ID of the plan owner; for a REBIND subcommand, indicates the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote rebind, and the OWNER was not specified in the subcommand, then REMOTE OWNER will be output.

    Attention: If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:

    • ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand, blank if the application plan is not specified.

dbrm name
    Name of the DBRM that contains the reference to the indicated object.

statement number
    Statement number of the SQL statement referencing the name of the object.

object name
    Name of the object that is not defined in the DB2 catalog.

System action: A valid plan is created if no errors are detected and VALIDATE(RUN) is in effect.

System programmer response: Ensure that the indicated table is created before the statement is executed. Otherwise, execution of this statement results in an error.

This statement is bound dynamically on each execution of the statement. For better performance, REBIND the application program after the indicated object is created.

DSNX101I   **bind-type SQL WARNING USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number THE SPECIFIED AUTHORIZATION-ID DOES NOT HAVE THE PRIVILEGE TO PERFORM OPERATION op name ON OBJECT object name**

Explanation: The identified authorization ID does not have a privilege required for application plan execution. If the access control authorization exit is active and the AUTHEXIT_CHECK system parameter is set to DB2, this error might occur if ACEE cannot be created for the package owner to perform the specified operation.

bind-type
    Type of BIND subcommand that caused the warning: BIND or REBIND

authorization-id
    Authorization ID used during the bind process. For a BIND subcommand, it is the authorization ID of the plan owner. For a REBIND subcommand, it is the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote operation, and the OWNER was not specified in the subcommand, then REMOTE OWNER is output. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was

---

DSNX102I

in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:

- ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand. If the application plan is not specified, this field is blank.

dbrm name
 Name of the DBRM that contains the unauthorized operation.

statement number
 Statement number of the SQL statement containing the unauthorized operation.

System action: A valid plan is created if no errors are detected and VALIDATE(RUN) is in effect.

System programmer response: You must take one of the following actions before the statement can be executed. Otherwise, the execution of the statement results in an error.

- Ensure that the appropriate authority is granted to the invoker of the latest BIND subcommand against this application plan
- Invoke the BIND subcommand again using an authorization ID with authority to perform all the operations in this application plan.
- If you use the access control authorization exit and if the ACEE of the package owner cannot be created, ensure that the package owner is defined in RACF.

This statement is bound dynamically on each execution of the statement. For better performance, rebind the application program after the appropriate authority is granted to the invoker of the BIND subcommand against this application plan.

DSNX102I   **bind-type SQL WARNING USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number THE REQUIRED INDEX HAS NOT BEEN CREATED ON TABLE table name IN PARTITIONED TABLESPACE table space name**

Explanation: There was an attempt to reference the table in the table space (both specified in the message) when the required index was not yet created.

bind-type
 Type of BIND subcommand that caused the warning: BIND or REBIND

authorization-id
 Authorization ID used during the bind process. For a BIND subcommand, it is the authorization ID of the plan owner. For a REBIND subcommand, it is the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote operation, and the OWNER was not specified in the subcommand, then REMOTE OWNER is output. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:

- ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand. If the application plan is not specified, this field is blank.

dbrm name
 Name of the DBRM that contains the reference to the indicated table.

statement number
 Statement number of the SQL statement referencing the indicated table.

System action: A valid plan is created if no errors are detected and VALIDATE(RUN) is in effect.

System programmer response: Ensure that the required index on the indicated table in the partitioned table space is created before the statement is executed. Otherwise, execution of this statement results in an error. This statement is bound dynamically on each execution of the statement. For better performance, rebind the application program after the required index is created.

DSNX103I **bind-type SQL WARNING USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number BIND ISOLATION LEVEL RR CONFLICTS WITH TABLESPACE LOCKSIZE PAGE OR LOCKSIZE ROW**

Explanation: Because of the possible inconsistency between the isolation level of the application and the LOCKSIZE of the table space, table space locking is used to protect the integrity of the application.

bind-type
> Type of BIND subcommand that caused the warning: BIND or REBIND

authorization-id
> Authorization ID used during the bind process. For a BIND subcommand, it is the authorization ID of the plan owner. For a REBIND subcommand, it is the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote operation, and the OWNER was not specified in the subcommand, then REMOTE OWNER is output.

> **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:
> * ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand. If the application plan is not specified, this field is blank.

dbrm name
> Name of the DBRM that contains the reference to the indicated table.

statement number
> Statement number of the SQL statement referencing the indicated table.

System action: A valid plan is created if no errors are detected. Table space locking is used. RR isolation level is preserved.

System programmer response: If you do not want table space locking, use isolation level UR, CS or RS.

DSNX104I *bind-type* **SQL WARNING USING** *authorization-id* **AUTHORITY PLAN=***plan-id* **DBRM=***dbrm-name* **STATEMENT=***statement-number* **THE RESULT OF DECIMAL MULTIPLICATION MAY CAUSE OVERFLOW**

Explanation: An arithmetic expression contains a decimal multiplication that might cause an overflow condition when the statement is executed. The problem can be corrected by restructuring the arithmetic expression so that decimal multiplication precedes decimal division or by changing the precision and scale of the operands in the arithmetic expression.

bind-type
> Type of BIND subcommand that caused the warning: BIND or REBIND

authorization-id
> Authorization ID used during the bind process. For a BIND subcommand, it is the authorization ID of the plan owner. For a REBIND subcommand, it is the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote operation, and the OWNER was not specified in the subcommand, then REMOTE OWNER is output.

> **Important:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:
> * ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand. If the application plan is not specified, this field is blank.

dbrm-name
> Name of the DBRM that contains the reference to the indicated table.

statement-number
    Statement number of the SQL statement referencing the indicated table.

System action: A valid plan is created if no errors are detected.

Related concepts:

▶ Decimal multiplication (DB2 SQL)

▶ Decimal division (DB2 SQL)

▶ Language elements (DB2 SQL)

DSNX105I   **bind-type SQL WARNING USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number SQLCODE=sqlcode SQLSTATE=sqlstate TOKENS=token list**

Explanation: An SQL warning code was generated while processing an SQL statement.

bind-type
    The type of subcommand that caused the warning: BIND or REBIND

authorization-id
    Indicates the authorization ID used during the bind process. For a BIND subcommand, it is the authorization ID of the plan owner. For a REBIND subcommand, it is the authorization ID of the owner specified by the subcommand, or the latest owner of the plan. If this is a remote operation, and the OWNER was not specified in the subcommand, then REMOTE OWNER is output.

    Attention: If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:

    • ROLE: *role-name*

plan-id Name of the application plan specified in the BIND subcommand. If the application plan is not specified, this field is blank.

sqlcode
    A product specific code which identifies the particular SQL warning condition.

sqlstate
    A common code which identifies the particular SQL warning condition.

dbrm name
    Name of the DBRM that contains the problem.

statement number
    Statement number of the SQL statement containing the problem.

token list
    Sequence of one or more tokens separated by blanks.

System action: A valid plan is created if no errors are detected.

System programmer response: Refer to the information for the particular abend reason code issued. If the reason code is negative, this statement is bound dynamically on each execution of the statement. For better performance, rebind the application program after correcting this statement or after the required object is created or altered.

Related concepts:

▶ SQL codes (DB2 Codes)

▶ DB2 reason codes (DB2 Codes)

DSNX106I   **ATTEMPTED REBIND OF PACKAGE pkg-id RESULTED IN A TOTAL OF number ERRORS**

Explanation: Errors have been detected during the remote REBIND process. Only the first REBIND error will produce a specific message. Additional errors will simply be totaled, and the number will be returned to the user.

pkg-id Name of the package specified in the subcommand.

number
> Total number of errors in the remote REBIND

System action: The package was not rebound.

System programmer response: See previously output message to determine the initial error that caused the subcommand to be terminated unsuccessfully.

DSNX200I   *bind-type* **SQL ERROR USING** *authorization-id* **AUTHORITY PLAN=***plan-id* **DBRM=***dbrm-name* **STATEMENT=***statement-number* **SQLCODE=***sql-code* **SQLSTATE=***sql-state***TOKENS=***token-list* **CSECT=***csect-name* **RDS CODE=***location-code*

Explanation: An error has been detected while processing an SQL statement.

bind-type
> BIND or REBIND indicates type of bind subcommand that caused the error.

authorization-id
> indicates the authorization ID used during the bind process. For a BIND subcommand, indicates the authorization ID of the plan owner. For a REBIND subcommand, indicates the authorization ID of the owner specified by the subcommand, or the latest owner of the plan.
>
> **Important:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. Otherwise an authorization ID is returned. A role is returned in the following format as a single token:

plan-id   Name of the application plan specified in the BIND subcommand. Blank if the application plan is not specified.

sql-code   A product specific code which identifies the particular SQL error detected.

sql-state   A common code which identifies the particular SQL error detected.

dbrm-name
> Name of the DBRM that contains the error.

statement-number
> Statement number containing the SQL error.

token-list
> Optional sequence of one or more tokens separated by blanks.

csect-name
> Name of the CSECT detecting the problem.

location-code
> Unique code that identifies the location within the CSECT when the problem is detected.

System action: No application plan is created.

System programmer response: Correct the SQL errors in the application program associated with the indicated DBRM. Precompile the application program, and issue the bind subcommand again.

Related concepts:

⇨ DB2 reason codes (DB2 Codes)

⇨ SQL codes (DB2 Codes)

DSNX201I   **bind-type SQL ERROR USING authorization-id AUTHORITY PLAN=plan-id DBRM=dbrm name STATEMENT=statement number SQL CODE=sql code TOKENS=token list; bind-type DBRM ERROR USING authorization-id AUTHORITY PLAN=plan-id DBRM dbrm name IS IN AN INCONSISTENT STATE**

Explanation: The content of a DBRM has been modified after the precompilation process.

bind-type
> Type of BIND subcommand that caused the warning: BIND, REBIND

authorization-id
> Authorization ID of the plan owner. **Attention:** If you are using a trusted context, the token *auth-id* might return a role instead of an authorization ID. A role is returned if a role was in effect and the authorization checking is performed against the role, rather than the authorization ID of the session, when the condition was encountered. The role is returned in the following format as a single token: `ROLE: role-name`.

plan-id Name of the application plan specified in the BIND or REBIND subcommand (or blank if the application plan is not specified).

dbrm name
> Name of the DBRM that is in error.

System action: No application plan is created.

System programmer response: Precompile the application program associated with the indicated DBRM. Reinvoke the BIND subcommand.

DSNX202I   INVALID AMODE ATTRIBUTE FOUND FOR DATE OR TIME INSTALLATION REPLACEABLE EXIT ROUTINE *module-name*

Explanation: During DB2 startup, the Relational Data System subcomponent loads the Date (DSNXVDTX) and Time (DSNXVTMX) exit routines. These routines must have the link-edited attributes AMODE(31) and RMODE(ANY). The module listed in the message 'module-name' did not have the correct addressing attributes. If the date and time LOCAL installation option is specified, DB2 cannot execute the date and time functions required by the exit routine that failed to load.

System action: The load module is deleted. DB2 startup continues.

Operator response: Notify the system programmer.

System programmer response: When DB2 is stopped, the date and time exit routine must be link-edited again with the correct AMODE and RMODE control statements.

Related reference:

➡ Coding rules for exit routines (DB2 Administration Guide)

Related information:

➡ Exit routines (DB2 Administration Guide)

DSNX203I   LOAD OF THE DATE OR TIME INSTALLATION REPLACEABLE EXIT ROUTINE module-name FAILED WITH MVS ABEND CODE = abend-code REASON CODE = reason-code DB2 STARTUP WILL CONTINUE

Explanation: During DB2 startup, the Relational Data System subcomponent loads the date (DSNXVDTX) and time (DSNXVTMX) exit routines. During the load of EXIT ROUTINE 'module-name', MVS issued the abend code and reason code that would cause abnormal termination of the task, but the abend was suppressed. If the date and time LOCAL option is specified, DB2 will be unable to execute the date and time functions required by the exit routine that failed to load.

System action: The load failed. DB2 startup continues.

Operator response: Notify the system programmer.

System programmer response: When DB2 is stopped, the date and time exit routine must be link-edited correctly.

Related reference:

➡ Coding rules for exit routines (DB2 Administration Guide)

Related information:

➡ Exit routines (DB2 Administration Guide)

DSNX204I   *csect-name* **ERROR LOADING THE DB2 CATALOG, REASON** *reason*: **DB2 STARTED ACCESS(***MAINT***)**

Explanation: The internal structure (DBD) for the catalog database (DSNDB06) was either not found or was inconsistent when DB2 was started. See the appropriate reason for a more detailed description of the problem.

The value for *reason* is one of the following reason codes:

0   An error occurred reading the DBD01 table space. Check the Console for messages and SYS1.LOGREC for abends.

1   The expected level of the DBD for DSNDB06 is the previous release, but objects exist in the DBD for a current or later release.

2   The expected level of the DBD for DSNDB06 is the current release, but objects exist in the DBD for a later release.

3   The expected level of the DBD for DSNDB06 is a later release, but objects do not exist in the DBD for a later release.

4   The level of the DBD for DSNDB06 is unknown to this version of DB2.

The *mode* is one of the following:

*****   The DB01 tablespace could not be opened. Code/Catalog checks are deferred until it can be opened.

MAINT
            The DB01 tablespace could be opened and a problem was detected. Prohibits access to any authorization IDs other than installation SYSADM and installation SYSOPR.

System action: DB2 was started ACCESS(MAINT).

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

System programmer response: DB2 was started in ACCESS(*mode*). Do not run any CREATES, ALTERS, DROPS, GRANTs, REVOKEs or BINDs when in this state; it might corrupt your catalog and directory. Determine the problem with the DSNDB01.DBD01 table space. An improper REPAIR or RECOVER of the table space might have been run. Fix the problem while in ACCESS(MAINT) mode and then stop and start DB2.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:
* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).

Related concepts:

▶ Recovery of catalog and directory tables (DB2 Administration Guide)

Related tasks:

▶ Recovering catalog and directory objects (DB2 Utilities)

Related reference:

▶ Table spaces and indexes (DB2 SQL)

DSNX205I   *csect-name* **THE DATA DEFINITION CONTROL OBJECT DEFINITIONS AND ATTRIBUTES ON MEMBER** *member-name* **ARE NOT THE SAME.**

Explanation: One or more data definition control object definitions and attributes on member *member-name* and on this member are not the same. The data definition control object definitions and attributes are defined in the initialization parameter module DSNZPARM.

System action: Processing continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

System programmer response: Use the same data definition control table object definitions and attributes for all members of the DB2 data sharing group.

Related tasks:

⮕ Installation step 17: Define and bind DB2 objects: DSNTIJSG (DB2 Installation and Migration)

Related reference:

⮕ Data definition control support panel: DSNTIPZ (DB2 Installation and Migration)

Related information:

⮕ Managing access through data definition control (Managing Security)

DSNX206I **LOAD OF THE ACCESS CONTROL AUTHORIZATION INSTALLATION REPLACEABLE EXIT ROUTINE** *module-name* **FAILED WITH MVS ABEND CODE =** *abend-code* **AND REASON CODE =** *reason-code* **DB2 STARTUP WILL CONTINUE**

Explanation: During DB2 startup, the Relational Data System subcomponent loads the Access Control Authorization (DSNX@XAC) exit routine. During the load of EXIT ROUTINE *module-name*, MVS issued the abend code and reason code that indicates abnormal termination of the task. However, the abend was suppressed. DB2 will not use the Access Control Authorization (DSNX@XAC) exit to perform authorization checking. DB2 authorization checking will be performed.

System action: The load failed. DB2 startup continues.

Operator response: Notify the system programmer.

System programmer response: After DB2 has stopped, you must properly link-edit the Access Control Authorization exit routine. \

Related reference:

⮕ Coding rules for exit routines (DB2 Administration Guide)

Related information:

⮕ Exit routines (DB2 Administration Guide)

DSNX207I **INVALID AMODE ATTRIBUTE FOUND FOR ACCESS CONTROL AUTHORIZATION INSTALLATION REPLACEABLE EXIT ROUTINE** *module-name*

Explanation: During DB2 startup, the Relational Data System subcomponent loads the Access Control Authorization (DSNX@XAC) exit routine. This routine must have the link-edited attributes AMODE(31) and RMODE(ANY). The module listed in the message *module-name* did not have the correct addressing attributes. DB2 will not use the Access Control Authorization (DSNX@XAC) exit to perform authorization checking. DB2 authorization checking will be performed.

System action: The system deletes the load module. DB2 startup continues.

Operator response: Notify the system programmer.

System programmer response: After DB2 has stopped, you must properly link-edit the Access Control Authorization exit routine with the correct AMODE and RMODE control statements.

Related reference:

⮕ Coding rules for exit routines (DB2 Administration Guide)

Related information:

⮕ Exit routines (DB2 Administration Guide)

DSNX208E *csect-name* **ATTEMPTING TO START DB2 WITH AN UNSUPPORTED CODE LEVEL. REASON CODE** *reason* **MISSING MAINTENANCE:** *maint* **CURRENT LEVEL** *current-code-level* **VALID LEVEL(S)** *valid-code-levels*

Explanation: The DB2 member that is starting has a code level that is not compatible with the catalog or with some other component of DB2.

reason indicates the problem:

1   DB2 is attempting to start with a level of code that is not valid. The missing maintenance, *maint*, is unknown.

2   The level of code that is required by the catalog, which is indicated by *current-code-level*, is not a valid code level. The missing maintenance, *maint*, is unknown.

3   A preconditioning or toleration APAR has not been applied. The required APAR or level of code is specified by *maint*.

4   The preconditioning or toleration maintenance is on the DB2 subsystem, but maintenance to the catalog that is specified by *maint*, has not been applied.

maint is an APAR number or a token that indicates a level of code. The value of *maint* is UNKNOWN if DB2 cannot determine the missing maintenance level.

current-code-level and *valid-code-levels* are for the use of your IBM service representative.

This message can occur in the following situations:
- The starting DB2 fell back from a later release and does not have the proper *maint* applied.
- The starting DB2 does not have the proper coexistence SPE, specified by *maint*, to start in a data sharing group, which is at a later release level.
- The starting DB2 is at a release in which coexistence with the data sharing group is not supported.

System action: DB2 terminates with reason code 00E7009D.

System programmer response: Verify that:
- Coexistence is supported between the releases you are trying to run.
- The DB2 startup procedure points to the correct set of load libraries.
- The correct SPE has been applied.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.

DSNX209E   *csect-name* **MEMBER** *member-name* **IS AT A CODE LEVEL WHICH IS NOT COMPATIBLE WITH THIS STARTING OR MIGRATING MEMBER. REQUIRED LEVEL:** *new-code-level* **KNOWN LEVELS:** *known-code-levels*

Explanation: The DB2 member that is starting will update the DB2 catalog, but the code level of that DB2 member is not compatible with the code level of the DB2 member that has already started. The DB2 subsystem at the earlier code level requires the correct SPE. The code level information in this message is for the use of your IBM service representative.

System action: DB2 terminates with reason code 00E7009D after detecting the DB2 members that cannot coexist.

System programmer response: Check the DB2 Program Directory and Preventive Service Planning updates (PSP) for more information about the SPE that is needed.

Verify that:
- Coexistence is supported between the releases you are trying to run.
- The DB2 startup procedure points to the correct set of load libraries.
- The correct SPE has been applied to all active members in the group.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).

DSNX210I

- Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004) utility.
- Unformatted DSN1PRNT output of the DSNDB01.DBD01 page set. Use the method described in Part 5 of *Diagnosis Guide and Reference* to locate the inconsistent DBD in the DSNDB01.DBD01 page set.

DSNX210I   *csect-name* **- ACCESS CONTROL AUTHORIZATION EXIT ROUTINE (DSNX@XAC) HAS INDICATED THAT IT SHOULD NOT BE CALLED, HAS ABENDED OR HAS RETURNED AN INVALID RETURN CODE DURING** *function-code*. **RETURN CODE=***return-code* **REASON CODE=***reason-code*. **CUMULATIVE ABENDS DURING EXIT PROCESSING=***abend-count*. **EXIT ROUTINE STATUS:** *stopped-running*.

Explanation: This message is issued during the initialization (DB2 startup) call or during an authorization check call to the exit routine, as identified by the *function-code*. The *return-code* and *reason-code* are returned in fields EXPLRC1 and EXPLRC2 of the EXPL parameter list for the exit routine.

This message is issued under the following circumstances:

Initialization

- The version of the exit routine is incompatible with this version of DB2.
- The exit routine returned an invalid code.
- The exit routine terminated abnormally during initialization.
- The exit routine returned-code is 12 indicating that it cannot perform authorization checks and should not be called.

In all of these cases the exit routine is stopped, as indicated by stopped- running=STOPPED.

Authorization checking

- The exit routine returned an invalid return code.
- The return code from the exit routine indicates that it cannot perform authorization checks and should not be called.
- The exit routine terminated abnormally during initialization.

System action: DB2 performs different actions, depending on whether this error occurs during DB2 startup or during an authorization check:

Initialization - DB2 startup:

- If the version of the exit routine is incompatible with this version of DB2, an abend with code 00E70016 occurs and DB2 terminates.
- If the return-code is 12 or an unexpected value, then:
  - If *reason_code* is 16, an abend with code 00E70015 occurs and DB2 terminates.
  - If *reason_code* is not 16, an abend with code 00E70009 occurs, DB2 performs all subsequent authorization checks, and the exit is not called again.
- If an abend occurs during exit routine processing:
  - If the *reason_code* is 16, DB2 terminates.
  - If the *reason_code* is not 16, DB2 performs all subsequent authorization checks, and the exit is not called again.

Authorization check:

- If the return-code is 12 or an unexpected value, then:
  - If the initialization *reason_code* was 16, an abend with code 00E70015 occurs and DB2 terminates.
  - If the initialization *reason_code* was not 16, an abend with code 00E70009 occurs, DB2 performs all subsequent authorization checks, and the exit is not called again.
- If an abend occurs during exit routine processing, a certain number of abends are tolerated before the exit stops being called. Subsystem parameter AEXITLIM (AUTH EXIT LIMIT) determines this limit. The exit routine will continue running until it exceeds the abend tolerance limit. In addition, the following will occur:
  - If the initialization *reason_code* was 16, DB2 terminates.
  - If the initialization *reason_code* was not 16, DB2 performs all subsequent authorization checks.

Operator response: Notify the system programmer.

System programmer response: Refer to the documentation of the access control authorization exit routine for an explanation of *reason_code*. Correct the situation that caused the reason code.

If this message was issued because of an abend during exit routine processing, then search the system log to determine its cause and correct the situation.

If abend code 00E70016 was issued indicating that the version of the exit routine is incompatible with this version of DB2, install the correct version of the exit routine. Refer to the documentation in the Administration Guide for information on the access control authorization exit. For information on the RACF access control module shipped with DB2 refer to the RACF Access Control Module Guide.

If DB2 terminated, restart DB2. If DB2 did not terminate, you need to restart DB2 to reactivate the exit.

Problem determination: Since the access control authorization exit routine is external to DB2, refer to the documentation of the exit routine for an explanation of *return_code* and *reason_code* and a resolution to the problem.

If this message was issued because of an abend during exit routine processing, then determine the cause of the abend from messages in the system log and address it. Look for system messages that occurred at approximately the same time to find more information about the abend.

DSNX211I **LOAD OF THE ICSF LOAD MODULE** *module-name* **FAILED WITH THE RETURN CODE** *return-code* **AND REASON CODE** *reason-code* **DB2 STARTUP CONTINUES.**

Explanation: DB2 cannot load the ICSF load module *module-name*. Refer to the z/OS ICSF documentation for explanations of the *return-code* reason code.

System action: DB2 startup continues but the encryption/decryption services are not available.

User response: Ensure that the ICSF SCSFMOD0 load module is in the LINKLIST for the z/OS system.

DSNX230I **A RESOURCE IS UNAVAILABLE WHEN TRYING TO BUILD THE TRUSTED CONTEXT CACHE REASON=***reason-code* **TYPE OF RESOURCE=***resource-type* **RESOURCE NAME=***resource-name*

Explanation: During DB2 startup or at the first attempt to create a connection or at a commit related to the TRUSTED CONTEXT DDL statements, an error is encountered while building the trusted context cache and accessing the catalog.

reason-code
The reason code returned by Data Manager during DB2 catalog access.

resource-type
The resource type returned by Data Manager during DB2 catalog access.

resource-name
The resource name returned by Data Manager during DB2 catalog access.

An attempt to rebuild the trusted context cache will be made at the next DB2 startup, or during commit for the TRUSTED CONTEXT related DDL statements or attempt to establish a connection.

System action: The trusted context cache is not built and subsequent connections are not trusted connections, until the problem is resolved. DB2 startup continues if the console message is issued during the startup. If the message is issued during a remote connect attempt to use the trusted connection, SQLCODE +20360 is returned. If the message is issued when DSN ASUSER is specified, DSNE137I message is returned.

If the message is issued during the commit for TRUSTED CONTEXT related DDL statements, the DB2 catalog is updated, but the trusted context cache is not updated.

Operator response: Notify the system programmer.

System programmer response: The message is issued when an error is encountered while trying to access the catalog tablespace, DSNDB06.SYSCONTX. The reason code indicates the cause of the failure. Verify the identity of the resource that was not available. Check if there is any prior abend messages in the console. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSNX231I   THE TRUSTED CONTEXT CACHE BUILD PROCESS COMPLETED SUCCESSFULLY

Explanation: The trusted context cache is built successfully with trusted context entries in the DB2 catalog. DB2 uses the trusted context cache to determine if a connection is trusted or not.

System action: DB2 is ready for trusted context connections.

DSNX232E   A RESOURCE IS UNAVAILABLE WHEN TRYING TO DETERMINE IF AUDIT POLICIES ARE TO BE STARTED DURING DB2 START UP REASON=*reason-code* **TYPE OF RESOURCE=***resource-type* **RESOURCE NAME=***resource-name*

Explanation: During DB2 startup, an error was encountered accessing the catalog. This error occurred when trying to determine if audit policies are to be started.

reason-code
> A numeric value that indicates the reason for the failure.

resource-type
> The type of resource.

resource-name
> The name of the resource.

System action: The audit policies that are defined to be started during DB2 startup are not started. DB2 startup continues.

Operator response: Notify the system programmer.

System programmer response: The message is issued when an error is encountered while trying to access the catalog table space, DSNDB06.SYSTSADT. The *reason-code* value indicates the cause of the failure.

Verify the identity of the resource that was not available. Check if there any prior abend messages are on the console. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Related concepts:

➡ DB2 reason codes (DB2 Codes)

Related information:

Appendix C, "Resource types," on page 1153

DSNX233E   AUDIT POLICIES SPECIFIED TO BE STARTED AT DB2 STARTUP EXCEED THE ALLOWED LIMIT

Explanation: The number of audit policies that are specified to be started during DB2 startup exceeded the allowed limit. Up to eight audit policies can be specified for the same value in the DB2START column of the SYSIBM.SYSAUDITPOLICIES table.

System action: Audit policies are not started for a DB2START column value that has more than eight audit policies specified. All other audit policies that are defined to be started at DB2 startup are started. DB2 startup continues.

System programmer response:

1. Correct the audit policy definition in the SYSIBM.SYSAUDITPOLICIES table.
2. Use the START TRACE command to start the required audit policies.

Related reference:

➡ -START TRACE (DB2) (DB2 Commands)

➡ DB2 catalog tables (DB2 SQL)

DSNX234I   *csect-name* CANNOT INITIALIZE XML SCHEMA VALIDATION FUNCTION: *function-name*. **RETURN CODE='***return-code***'X AND REASON CODE='***reason-code***'X.**

Explanation: The invocation of the z/OS XML System Services macro that loads the required module for XML schema validation failed.

csect-name
> The name of the control section that issued the message.

function-name
> The name of the XML schema validation function that failed.

return-code
> The return code from the z/OS XML System Services call.

reason-code
> The reason code from the z/OS XML System Services call.

System action: DB2 subsystem startup processing continues.

System programmer response: If you use XML schema validation, complete one of the following actions:

- Use the return and reason codes issued by XML System Services to correct the problem, and restart DB2.
- Execute the following command to load the required z/OS XML System Services module:

```
SETPROG LPA,ADD,MODNAME=(GXLIMODV),DSNAME=
SYS1.SIEALNKE
```

Related reference:

z/OS XML User's Guide and Reference

DSNX235I REGISTRATION OF THE RACF ENF MESSAGE LISTENER EXIT WITH THE ENF FACILITY FOR ENF *enf-signal* **FAILED WITH RETURN CODE** *return-code*.

Explanation: The process that registers DB2 with the RACF Event Notification Facility (ENF) message listener exit for listening to the specified ENF signal returned an unexpected return code.

enf-signal
> The possible values are 62, 71, and 79.

return-code
> Hexadecimal value of the z/OS ENFREQ macro return code.

System action: DB2 is not registered to listen to the specified ENF signal notification. DB2 continues its startup.

System administrator response: Notify the security administrator. Restart DB2 after the ENF signal notification problem is fixed.

Security administrator response: Make sure that DB2 is registered to listen to the specified ENF signal notification.

Operator response: No action is required.

DSNX236I A RESOURCE *resource-name* **TYPE OF RESOURCE** *resource-type* **FOR PROCESSING ENF SIGNAL FOR AUTHID** *authid* **OPERATION** *operation* **ON OBJECT** *object-name* **IS NOT AVAILABLE FOR REASON** *reason-code*. **ENF SIGNALS RECEIVED FOR CLASS** *class-name* **ARE NOT PROCESSED FOR PACKAGE INVALIDATION.**

Explanation: The ENF signal process for package invalidation has failed because a required resource *resource-name* is not available. The ENF signals received for the class *class-name* are not processed.

resource-name
> The name of the resource.

resource-type
> The type of the resource.

authorization-ID
> The authorization identifier that is identified in the message. The authorization-ID can be a role.

operation
> The operation that is performed.

object-name
> The name of the object. If the operation is EXECUTE PACKAGE, the object name consists of the collection ID and the package name. For all other operations, the object name consists of the schema name and the object name.

reason-code
> A numeric value that indicates the reason for the failure of the operation.

class-name
> The name of the RACF resource class for DB2 objects and administrative authorities.

System action: The ENF signal processing continues.

System administrator response: Ensure that the required resource is available for the ENF signal process. Manually restart the package invalidation process. Notify the security administrator.

Security administrator response: Identify the RACF commands that were issued to remove resource access for the specified RACF class. Permit the user access to the identified resources and then delete the permissions in RACF.

Operator response: No action is required.

DSNX237I AN ABEND HAS OCCURRED DURING ENF SIGNAL PROCESSING. ENF SIGNALS RECEIVED FOR CLASS *class-name* **ARE NOT PROCESSED.**

Explanation: An abend has occurred in DB2 when processing the ENF signal received from the security server. The ENF signals received for the class-name are not processed.

class-name
> RACF resource class for DB2 objects and administrative authorities

System action: The ENF signal processing continues.

System administrator response: Manually restart the cache refresh and package invalidation processes. Notify the security administrator.

Security administrator response: Identify the RACF commands that were issued to remove resource access for the specified RACF class. Permit the user access to the identified resources and then delete the permissions in RACF.

Operator response: No action is required.

DSNX240I *csect-name* **EXTERNALIZATION OF STATISTICS RECOMMENDATIONS HAS BEEN STOPPED.**

Explanation: The task used to populate SYSIBM.SYSSTATFEEDBACK is no longer running. This result can occur at DB2 startup if DB2 is unable to start the task or if this task terminates abnormally.

System action: Processing continues, but DB2 no longer externalizes statistics recommendations to SYSIBM.SYSSTATFEEDBACK automatically until DB2 is recycled. In most cases, the ACCESS DATABASE command with parameter MODE(STATS) can still be used to populate SYSIBM.SYSSTATFEEDBACK.

Missing statistics that are identified during the explain process continue to be written to the DSN_STAT_FEEDBACK explain table.

System programmer response: This message is issued when the asynchronous task that handles externalization of the missing statistics to the SYSIBM.SYSSTATFEEDBACK table is no longer running. If this situation happens at DB2 startup, check if the system is short on storage or has other restrictions that might preclude the creation of a TCB. Otherwise, check the system log to determine the cause of the abnormal task termination.

DSNX800I *csect-name* **ACCELERATOR FUNCTION IS NOT AVAILABLE**

Explanation: The accelerator service is not available. Either a subsystem parameter is not set to enable query acceleration or an error occurred during initialization.

csect-name
> The name of the control section that issued the message.

System action: Processing continues without acceleration.

User response: Check the value of the ACCEL subsystem parameter. If the value is NO, query acceleration is disabled. Change the value to AUTO or COMMAND, recycle DB2, and reissue the command.

If the ACCEL subsystem parameter is already set to AUTO or COMMAND, determine the cause of the initialization failure. Check that the accelerator catalog tables (SYSACCEL.SYSACCELERATORS and SYSACCEL.SYSACCELERATEDTABLES) were created correctly. Also, check the accelerator installation information.

Severity: 4 (warning)

Related concepts:

Accelerators for DB2 for z/OS queries (DB2 Performance)

Related reference:

ACCELERATION STARTUP field (ACCEL subsystem parameter) (DB2 Installation and Migration)

SYSACCEL.SYSACCELERATORS table (DB2 SQL)

SYSACCEL.SYSACCELERATEDTABLES table (DB2 SQL)

Information about one example of an IBM accelerator product

DSNX801I *csect-name* **ACCELERATOR PROCESSING STARTING**

Explanation: Support for the accelerator service is initializing.

csect-name
> The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNX802I *csect-name* **ACCELERATOR PROCESSING ON** *accelerator-name* **DETECTED A PROBLEM**

Explanation: The accelerator service for the specified server detected a problem and did not start.

csect-name
> The name of the control section that issued the message.

accelerator-name
> The name of the accelerator server.

System action: The accelerator service does not start. Reason code 00E7000A is issued and a dump is generated.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.
- The dump that accompanies reason code 00E7000A.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 4 (warning)

Related information:

00E7000A (DB2 Codes)

DSNX809I *csect-name* **ACCELERATOR PROCESSING STOP COMPLETE**

Explanation: The accelerator service is no longer active.

csect-name
> The name of the control section that issued the message.

System action: Processing continues without acceleration.

User response: No action is required.

DSNX810I • DSNX812I

Severity: 0

DSNX810I   *csect-name command-name* **ACCEL FOLLOWS -**

Explanation: This message is part of the output of an accelerator command and provides a heading for other messages that are generated by the command.

csect-name
> The name of the control section that issued the message.

command-name
> The command that was issued.

System action: Processing continues.

User response: No action is required.

Related reference:

⬛➡ -DISPLAY ACCEL (DB2) (DB2 Commands)

⬛➡ -START ACCEL (DB2) (DB2 Commands)

⬛➡ -STOP ACCEL (DB2) (DB2 Commands)

DSNX811I   *csect-name* **NAME** *accelerator-name* **INVALID, ACCELERATOR IGNORED.**

Explanation: The specified accelerator server is not known.

csect-name
> The name of the control section that issued the message.

accelerator-name
> The name of the accelerator server.

System action: Processing continues.

User response: Correct the accelerator name, and resubmit the command.

Severity: 8 (error)

DSNX812I   *csect-name* **SCOPE(GROUP) CANNOT BE PROCESSED DUE TO COMMAND ALREADY IN PROGRESS**

Explanation: A DISPLAY ACCEL, START ACCEL, or STOP ACCEL command was issued with the with SCOPE(GROUP) parameter at the same time that another data sharing member was processing a command. DB2 could not obtain the lock necessary to serialize these commands.

csect-name
> The name of the control section that issued the message.

System action: The command is not processed on any members of the data sharing group.

User response: Wait until the command that is already in progress completes, and resubmit this command. If the command that is already in progress does not complete, stop it.

Severity: 8 (error)

Related reference:

⬛➡ -DISPLAY ACCEL (DB2) (DB2 Commands)

⬛➡ -START ACCEL (DB2) (DB2 Commands)

⬛➡ -STOP ACCEL (DB2) (DB2 Commands)

DSNX813I   *csect-name* **MEMBER(***member-name***) COMMAND CANNOT BE PROCESSED DUE TO COMMAND ALREADY IN PROGRESS**

Explanation: A DISPLAY ACCEL, START ACCEL, or STOP ACCEL command was issued with the MEMBER(*member-name*) option at the same time that another data sharing member was processing a SCOPE(GROUP) or MEMBER(*member-name*) command. DB2 could not obtain the lock necessary to serialize these commands.

csect-name
         The name of the control section that issued the message.

member-name
         The member name of the DB2 data sharing member that is not in this data sharing group.

System action: The command is not processed on any members of the data sharing group.

User response: Wait until the command that is already in progress completes, and resubmit this command. If the command that is already in progress does not complete, stop it.

Severity: 8 (error)

Related reference:

▶ -DISPLAY ACCEL (DB2) (DB2 Commands)

▶ -START ACCEL (DB2) (DB2 Commands)

▶ -STOP ACCEL (DB2) (DB2 Commands)

DSNX814I   *csect-name* **MEMBER** *member-name* **IS NOT IN THIS DATA SHARING GROUP**

Explanation: The member list contains an invalid member name.

csect-name
         The name of the control section that issued the message.

member-name
         The member name of the DB2 data sharing member that is not in this data sharing group.

System action: Processing continues.

User response: Correct the member name, and resubmit the command.

Severity: 8 (error)

DSNX815I   *csect-name* **ACCELERATOR** *accelerator-name* **NOT FOUND**

Explanation: The specified accelerator has not been started.

csect-name
         The name of the control section that issued the message.

accelerator-name
         The name of the accelerator server.

System action: Processing continues.

User response: Correct the accelerator name, and resubmit the command.

Severity: 8 (error)

DSNX816I   *csect-name* **NO ACCELERATORS FOUND**

Explanation: No accelerators have been started.

csect-name
         The name of the control section that issued the message.

System action: Processing continues.

User response: Define one or more accelerators, and resubmit the command.

Severity: 8 (error)

DSNX818I   *csect-name command-name* **ACCEL INCOMPLETE**

Explanation: A multi-line message is incomplete for a command. There is not enough storage space available to store the required information from this command.

csect-name
> The name of the control section that issued the message.

command-name
> The command that was issued.

System action: Processing continues.

Message DSNX819I follows and provides additional information.

User response: No action is required.

Related reference:

⮕ -DISPLAY ACCEL (DB2) (DB2 Commands)

⮕ -START ACCEL (DB2) (DB2 Commands)

⮕ -STOP ACCEL (DB2) (DB2 Commands)

Related information:

"DSNX819I"

DSNX819I   *csect-name* **DISPLAY TERMINATED DUE TO INSUFFICIENT MESSAGE SPACE**

Explanation: There is not enough storage available to store the required information from this command.

csect-name
> The name of the control section that issued the message.

System action: The display is truncated.

System programmer response: Resubmit the command, specifying a smaller number of accelerator servers.

Related reference:

⮕ -DISPLAY ACCEL (DB2) (DB2 Commands)

⮕ -START ACCEL (DB2) (DB2 Commands)

⮕ -STOP ACCEL (DB2) (DB2 Commands)

Related information:

"DSNX818I"

DSNX820I   *csect-name* **START ACCELERATOR SUCCESSFUL FOR** *accelerator-name*

Explanation: The specified accelerator server is started.

csect-name
> The name of the control section that issued the message.

accelerator-name
> The name of the accelerator server.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related reference:

➡ -START ACCEL (DB2) (DB2 Commands)

DSNX821I   *csect-name* **ALL ACCELERATORS STARTED.**

Explanation: All accelerator servers are started.

csect-name
    The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNX822I   *csect-name* **START ACCELERATOR UNSUCCESSFUL FOR** *accelerator-name* **REASON** *reason-code*

Explanation: The accelerator server is not started.

csect-name
    The name of the control section that issued the message.

accelerator-name
    The name of the accelerator server.

reason-code
    A numeric value that indicates the reason for the failure.

System action: Processing continues.

User response: Use the information that is provided for the reason code to resolve the problem, then try to start the service.

Severity: 8 (error)

Related concepts:

➡ DB2 reason codes (DB2 Codes)

Related reference:

➡ -START ACCEL (DB2) (DB2 Commands)

DSNX823I   *csect-name* **ACCELERATOR** *accelerator-name* **ALREADY STARTED**

Explanation: The specified accelerator server is already active.

csect-name
    The name of the control section that issued the message.

accelerator-name
    The name of the accelerator server.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNX830I   *csect-name*

Explanation: This message displays output from the DISPLAY ACCEL command.

csect-name
    The name of the control section that issued the message.

The output provides the following information about the accelerator server or servers:

DSNX830I

ACCELERATOR
>The name of the accelerator server.

MEMB The name of the DB2 data-sharing member.

STATUS
>The status of the accelerator server. The status can be any of the following values:

>**STARTED**
>>The accelerator server is able to accept requests.

>**STARTEXP**
>>The accelerator server was started with the EXPLAINONLY option and is available only for EXPLAIN requests.

>**STOPPEND**
>>The accelerator server is no longer accepting new requests. Active accelerator threads are allowed to complete normally, and queued accelerator threads are terminated. The accelerator server was placed in this status by the STOP ACCEL MODE(QUIESCE) command.

>**STOPPED**
>>The accelerator server is not active. New requests for the accelerator are rejected. The accelerator server was placed in this status by the STOP ACCEL command.

>**STOPERR**
>>The accelerator server is not active. The accelerator server was placed in this status by an error condition. New requests for the accelerator are rejected.

REQUESTS
>The number of query requests that have been processed successfully.

ACTV The current number of active, accelerated queries.

QUED The current number of queued requests.

MAXQ The highest number of queued requests that have been reached since the accelerator was started.

Detailed output: If the DISPLAY ACCEL command is issued with the DETAIL option, the following additional information is provided about the accelerator server.

LOCATION
>The location name of the accelerator server as specified in table SYSACCEL.SYSACCELERATORS.

heartbeat-status
>The health status of the accelerator server. The status can be any of the following values:

>**HEALTHY**
>>The accelerator server is accepting and processing requests.

>**BUSY** The accelerator server is processing requests, but it is not accepting new requests.

>**FLATLINE**
>>The accelerator server is not responding.

>**AUTHFAIL**
>>DB2 and the accelerator server could not establish the credentials that are required to communicate with each other. The accelerator server did not accept the value of the ACCELERATORAUTHTOKEN column in the SYSACCELERATORS table for this accelerator.

>**DDFFAIL**
>>A DDF problem occurred during connection to the accelerator server. Messages on the z/OS console that begin with DSNL provide information for resolving the problem.

LEVEL The product level of the accelerator.

STATUS
>The status of the accelerator server. The status can be any of the following values:

>**CLUSTER IS INITIALIZING**
>>Data is still being loaded. The accelerator server is not ready for use.

FULLY OPERATIONAL
> All nodes are operational.

RECOVERY MODE
> A coordinator node has converted to a worker node as part of a failover. The accelerator server is operational.

RECOVERING
> The accelerator server is recovering from a failing worker node.

ERROR
> The accelerator server is not operational because of an unrecoverable problem.

MAINTENANCE
> The accelerator server is in maintenance mode and is not operational.

MAINTENANCE PENDING
> The accelerator server is preparing to enter maintenance mode. It will complete currently active and queued work but rejects new requests.

BUSY The accelerator server is operational, but queues are exceeding the configured limit. The accelerator rejects new requests until the number of queues drops below the limit.

FAILED QUERY REQUESTS
> The number of failed requests for the accelerator.

AVERAGE QUEUE WAIT
> The length of the average queue wait time within the accelerator, measured in milliseconds.

MAXIMUM QUEUE WAIT
> The length of the longest queue wait time within the accelerator, measured in milliseconds.

TOTAL NUMBER OF PROCESSORS
> The number of processors within the accelerator. For a multi-core processor, each core is counted as a separate processor.

AVERAGE CPU UTILIZATION ON COORDINATOR NODES
> Average processor utilization for coordinator node in this accelerator server. This value is a number in the range 0.00-100.00, where 0.00 indicates that all processors are idle, and 100.00 means that all processors are fully used.
>
> This value does not represent the current state of processor utilization, but rather the average utilization within the last 60 seconds. The returned value is the average, calculated over all coordinator nodes within the accelerator since startup.

AVERAGE CPU UTILIZATION ON WORKER NODES
> The average processor utilization for worker nodes in this accelerator server. This value is a number in the range 0.00-100.00, where 0.00 indicates that all processors are idle, and 100.00 means that all processors are fully used.
>
> This value does not represent the current state of processor utilization, but rather the average utilization within the last 60 seconds. The returned value is the average, calculated over all worker nodes within the accelerator since startup.

NUMBER OF ACTIVE WORKER NODES
> The number of active worker nodes in the accelerator.

TOTAL DISK STORAGE AVAILABLE
> The total amount of disk storage that is available in the accelerator, measured in megabytes (MB).

TOTAL DISK STORAGE IN USE
> The percentage of the total disk storage within the accelerator that is being used by the accelerator. This value is a number in the range 0.00-100.00, where 0.00 indicates that none of the disk storage is in use, and 100.00 means that all of the disk storage is in use.

DISK STORAGE IN USE FOR DATABASE
> The amount of disk storage that is being used by the database within the accelerator, measured in megabytes (MB).

The output terminates with message DSNX859I.

System action: Processing continues.

User response: No action is required.

Related reference:

➡ -DISPLAY ACCEL (DB2) (DB2 Commands)

➡ SYSACCEL.SYSACCELERATORS table (DB2 SQL)

DSNX859I   *csect-name command-name* **REPORT COMPLETE**

Explanation: This message marks the end of the report that started with message DSNX830I.

csect-name
        The name of the control section that issued the message.

command-name
        The command that was issued.

System action: Processing continues.

User response: No action is required.

Related information:

"DSNX830I" on page 1019

DSNX860I   *csect-name* **STOP ACCELERATOR SUCCESSFUL FOR** *accelerator-name*

Explanation: The accelerator server is stopped.

csect-name
        The name of the control section that issued the message.

accelerator-name
        The name of the accelerator server.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNX861I   *csect-name* **ALL OTHER ACCELERATORS STOPPED**

Explanation: A STOP ACCEL(*) command is currently in effect. Any accelerators that are not listed in the preceding DSNX860I messages are stopped.

csect-name
        The name of the control section that issued the message.

System action: Processing continues.

User response: No action is required.

Severity: 0

Related information:

"DSNX860I"

DSNX862I   *csect-name* **ACCELERATOR** *accelerator-name* **ALREADY STOPPED**

Explanation: The accelerator server is already stopped.

csect-name
        The name of the control section that issued the message.

accelerator-name
        The name of the accelerator server.

System action: Processing continues.

User response: No action is required.

Severity: 0

DSNX880I   *csect-name* **DDF** *operation* **FAILED WITH RETURN CODE=***return-code* **AND SQLCODE =** *sql-code*
SQLERRMT = *sqlca-tokens* **SQLERRP =** *sqlerrp* **SQLERRD =** *sqlerrd1 sqlerrd2 sqlerrd3 sqlerrd4 sqlerrd5*
sqlerrd6 **SQLWARN 0=***sqlwarn0*, **1=***sqlwarn1*, **2=***sqlwarn2*, **3=***sqlwarn3*, **4=***sqlwarn4*, **5=***sqlwarn5*, **6=***sqlwarn6*,
7=*sqlwarn7*, **8=***sqlwarn8*, **9=***sqlwarn9*, **A=***sqlwarna*, **SQLSTATE =** *sql-state*

Explanation: A distributed data facility (DDF) operation failed.

csect-name
       The name of the control section that issued the message.

operation
       The name of the DDF operation that failed. Possible values are CONNECT or SNDMON.

return-code
       The return code that was reported by DDF.

sql-code   A numeric value that indicates why the SQL statement execution was unsuccessful.

sqlca-tokens
       The message tokens from the SQLERRM field of the SQLCA that is returned by DDF.

sqlerrp   The contents of the SQLERRP field of the SQLCA that is returned by DDF.

sqlerrd1, sqlerrd2, sqlerrd3, sqlerrd4, sqlerrd5, sqlerrd6
       The contents of the SQLERRD field of the SQLCA that is returned by DDF.

sqlerrd1, sqlerrd2, sqlerrd3, sqlerrd4, sqlerrd5, sqlerrd6
       The contents of the SQLERRD field of the SQLCA that is returned by DDF.

sqlwarn0, sqlwarn1, sqlwarn2, sqlwarn3, sqlwarn4, sqlwarn5, sqlwarn6, sqlwarn7, sqlwarn8, sqlwarn9, sqlwarna
       The contents of the SQLWARN0, SQLWARN1, SQLWARN2, SQLWARN3, SQLWARN4, SQLWARN5,
       SQLWARN6, SQLWARN7, SQLWARN8, SQLWARN9, and SQLWARNA fields of the SQLCA that is returned
       by DDF.

sql-state   The corresponding SQLSTATE for the SQLCODE.

If the *operation* is CONNECT and the *return-code* is 8, the location name in the row that defines the accelerator in the
SYSACCEL.SYSACCELERATORS table is incorrectly set to the local DB2 location.

System action: The accelerator server is placed in the STOPERR state. As a result, the accelerator server is not
active, and new requests for the accelerator server are rejected.

System programmer response: Refer to the documentation of the SQLCODE that is specified in the message. Use
the suggested actions in that documentation to correct the problem. Then issue the -START ACCEL command for the
accelerator server.

If the *operation* is CONNECT and the *return-code* is 8, ensure that the LOCATION column of the row that defines the
accelerator in SYSACCEL.SYSACCELERATORS is a valid accelerator location name.

User response: Notify the system programmer.

Related reference:

↪ Description of SQLCA fields (DB2 SQL)

↪ SYSACCEL.SYSACCELERATORS table (DB2 SQL)

DSNX881I   *csect-name message*

Explanation: An accelerator server returned an alert message.

message   The message that is returned from the accelerator server.

System action: Processing continues.

Operator response: Refer to the product documentation for your accelerator server for more information about the message.

Related reference:

▶ Information about one example of an IBM accelerator product

Related information:

DSNX881I message structure

DSNX905I *csect-name* **PROCEDURE OR FUNCTION** *name* **TERMINATED ABNORMALLY ASID=** *asid*
WLM_ENV= *wlmenv*

Explanation: A stored procedure terminated abnormally while attempting to satisfy an SQL CALL statement for an SQL application.

name The name of the stored procedure or function that terminated abnormally.

asid The MVS ASID of the address space where the stored procedure executed.

wlmenv The name of the WLM application environment where the stored procedure executed. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

System action: The SQL CALL statement fails, and the SQL application is forced to roll back.

System programmer response: Correct the cause of the abnormal termination. If the correction requires the stored procedure to be linkedited again, then it might be necessary to cause the corrected MVS load module to be reloaded.

If it was run in a WLM-established stored procedures address space, issue the MVS VARY WLM,APPLENV=name,REFRESH command.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX906I *csect-name* **PROCEDURE OR FUNCTION** *name* **TERMINATED ABNORMALLY. THE PROCEDURE
HAS BEEN STOPPED. ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: A stored procedure terminated abnormally while attempting to satisfy an SQL CALL statement for an SQL application.

name The name of the stored procedure or function that terminated abnormally

asid The MVS ASID of the address space where the stored procedure executed.

wlmenv The name of the WLM application environment where the stored procedure executed. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

System action: The SQL CALL statement fails, and the SQL application is forced to roll back.

The abnormal termination count for procedure *procname* was reached. The procedure is stopped and further requests for the procedure are rejected.

System programmer response: Correct the cause of the abnormal termination. If the correction requires the stored procedure to be linkedited again, then it might be necessary to cause the corrected MVS load module to be reloaded.

If it was run in a WLM-established stored procedures address space, issue the MVS VARY WLM,APPLENV=name,REFRESH command.

After the appropriate command has been issued to refresh the stored procedure load module, use the -START PROCEDURE command to activate the stored procedure.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.

- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX908I   *csect-name* **PROCEDURE OR FUNCTION** *name* **WITH LOAD MODULE** *module* **EXCEEDED CPU RESOURCE LIMIT SSN=** *ssn* **PROC=** *jclproc***ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: A stored procedure or function consumed more CPU service units than allowed by the ASUTIME column of the ALTER PROCEDURES catalog table. The stored procedure might be in a loop.

name   The name of the stored procedure or function being processed

module   The MVS load module associated with the stored procedure

ssn   The name of the DB2 subsystem that called the stored procedure.

jclproc   The name of the JCL PROC used to start this stored procedures address space.

asid   The MVS ASID of the address space where the stored procedure executed.

wlmenv   The name of the WLM application environment where the stored procedure executed. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

System action: DB2 detaches the TCB associated with the stored procedure, which causes abnormal termination of the stored procedure.

Operator response: Contact the system programmer.

System programmer response: If the stored procedure is looping, correct the stored procedure.

If the stored procedure is performing correctly, increase the ASUTIME value in the CREATE PROCEDURES row associated with the stored procedure.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX922I   *csect-name* **BEGINNING DISCONNECTION OF STORED PROCEDURE ADDRESS SPACES**

Explanation: This message is produced during -STOP DB2 processing. It indicates that DB2 SQL processing has been quiesced, and DB2 is now beginning the termination process for the stored procedures address spaces.

DSNX923I   *csect-name* **ALL STORED PROCEDURE ADDRESS SPACES ARE NOW DISCONNECTED.**

Explanation: This message is produced during -STOP DB2 processing. It indicates that all of the stored procedure address spaces are now disconnected from DB2 and the stop of DB2 will continue.

WLM-established stored procedure address spaces are terminated by WLM after the disconnection from DB2.

DSNX930I   **DB2-MANAGED STORED PROCEDURES ADDRESS SPACE NOT SUPPORTED.**

Explanation: An address space referencing module DSNX9STP was started. In prior releases of DB2, this was the module used to execute DB2-managed stored procedures, which were defined as NO WLM ENVIRONMENT. This function is no longer supported by DB2.

System action: The address space terminates and this message is issued.

System programmer response: Use only WLM-managed stored procedures in V9. Do not attempt to start an address space referencing DSNX9STP.

DSNX940I   *csect-name* **- DISPLAY PROCEDURE REPORT FOLLOWS-**

Explanation: This message displays output from the DISPLAY PROCEDURE command. The output from this command provides information about the requested stored procedures and has the following general structure:

DSNX940I

```
DSNX940I csect-name - DISPLAY PROCEDURE REPORT FOLLOWS-

------ SCHEMA = schema
PROCEDURE  STATUS  ACTIVE  QUED MAXQ  TIMEOUT FAIL WLM_ENV
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env


------ SCHEMA = schema
csect-name ALL PROCEDURES STARTED FOR THIS SCHEMA.
PROCEDURE  STATUS  ACTIVE  QUED MAXQ  TIMEOUT FAIL WLM_ENV
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env

------ SCHEMA = schema
PROCEDURE  STATUS  ACTIVE  QUED MAXQ  TIMEOUT FAIL WLM_ENV
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env
procedure  status  active  qued  maxq  timeout  fail wlm_env
csect-name PROCEDURES begin - end STOP (act)
...

csect-name ALL OTHER PROCEDURES IN OTHER SCHEMAS STOPPED act

csect-name DISPLAY PROCEDURE REPORT COMPLETE
```

In your output, some of these lines might be included more than once and some might not be included at all. Which lines are included depends on the situation, as noted in the following descriptions of the variables in the DISPLAY PROCEDURE output.

"Error messages" on page 1027 might also be included in your output if a problem occurred.

SCHEMA = *schema*

> The schema name for the stored procedures that follow in the output.

> The schema name and lines that follow are repeated as many times as needed to satisfy the DISPLAY PROCEDURE request.

PROCEDURE *procedure*

> The name of the stored procedure.

> One line of output is included for each requested stored procedure.

STATUS *status*

> The status of the stored procedure. The status can be any of the following values:

> **STARTED**

>> The stored procedure can accept SQL CALL requests.

> **STOPABN**

>> The stored procedure was stopped because of a previous abnormal termination of the stored procedure application. SQL CALL requests for the stored procedure are rejected.

> **STOPQUE**

>> The stored procedure was stopped. SQL CALL requests for the stored procedure are queued until the stored procedure is started or the timeout interval that is defined by the STORTIME subsystem parameter expires.

> **STOPREJ**

>> The stored procedure was placed in this status by the STOP PROCEDURE command with the ACTION(REJECT) option. SQL CALL requests for the stored procedure are rejected.

> If the stored procedure is in any of the stopped states, you can use the START PROCEDURE command to place the stored procedure in STARTED status.

ACTIVE *active*

> The number of DB2 agents that are currently running this stored procedure.

QUED *qued*

> The number of DB2 agents that are currently waiting for this stored procedure to be scheduled.

MAXQ *maxq*

> The maximum number of DB2 agents that have waited for this stored procedure to be scheduled since DB2 was started.

TIMEOUT *timeout*

> The number of times that an SQL CALL statement timed out while it waited for a stored procedure to be scheduled since DB2 was started.

> An SQL CALL statement can time out for any of the following reasons:

> - The procedure can be stopped by the STOP PROCEDURE command with the ACTION(QUEUE) option. This specification causes requests for the procedure to time out after the interval that is defined by the STORTIME subsystem parameter.

> - The DB2 stored procedures address space might not have enough MVS TCBs to handle the volume of SQL CALL statements.

> - The MVS dispatching priority of the DB2 stored procedures address space might be too low to process the stored procedure requests in a timely manner.

FAIL *fail*

> The number of times that this procedure abended since it was last started.

WLM_ENV *wlm_env*

> The MVS Workload Manager environment name for this stored procedure.

csect-name

> The name of the control section that issued the message.

PROCEDURES *begin* **-** *end* **STOP (***act***)**

> This line is displayed when a STOP PROCEDURE command caused a range of procedures to be stopped. This situation can occur when the STOP PROCEDURE command specifies a partial name with a pattern-matching character (*), as in the following example: `-STOP PROCEDURE(ABC*)` .

> *begin* The first procedure name in the range of procedures that are currently stopped.

> *end*   The last procedure name in the range of procedures that are currently stopped.

> *act*   The action that was specified on the STOP PROCEDURE command.

ALL PROCEDURES STARTED FOR THIS SCHEMA.

> This line is displayed when both of the following conditions are true:

> - A STOP PROCEDURE(*.*) command operation was in effect when DISPLAY PROCEDURE was issued.

> - All of the procedures are started for the schema that is identified in a preceding line in the output.

ALL OTHER PROCEDURES IN OTHER SCHEMAS STOPPED *act*

> This line is displayed when both of the following conditions are true:

> - A STOP PROCEDURE(*.*) command operation was in effect when DISPLAY PROCEDURE was issued.

> - Any procedures that are not already listed in this DISPLAY PROCEDURE output are stopped.

> *act*   The action that was specified on the STOP PROCEDURE command.

DISPLAY PROCEDURE REPORT COMPLETE

> The report completed successfully.

Error messages:

The following messages in the DISPLAY PROCEDURE output indicate that an error occurred:

csect-name **PROCEDURE HAS NOT BEEN ACCESSED OR IS NOT DEFINED**
csect-name **FUNCTION HAS NOT BEEN ACCESSED OR IS NOT DEFINED**

> The routine name that was supplied on the DISPLAY PROCEDURE command either does not exist or has not been accessed since DB2 started. The specific procedure or function name might not be present in the SYSIBM.SYSROUTINES catalog table.

Processing continues for any additional routine names that were supplied on the command.

DISPLAY TERMINATED WITH MAX LINES

The DISPLAY PROCEDURE report was truncated. The report was generated in response to a command from an MVS console, and more than 254 response messages were generated. Only 254 response messages are returned.

Command processing completes. The truncated report is written to the MVS console.

Reissue the DISPLAY PROCEDURE command and specify a smaller number of routines. Specify a list of routines, a schema qualifier, or a partial routine name in the DISPLAY command rather than the default (*.*). Alternatively, reissue the command from the DB2 Commands DB2I panel or from a batch job rather than from the MVS console.

DSNX978I DISPLAY TERMINATED DUE TO INSUFFICIENT MESSAGE SPACE

See "DSNX978I" on page 1035.

DSNX944I *csect-name* **THE STORED PROCEDURE FUNCTION IS NOT AVAILABLE**

See "DSNX944I."

DSNX994I SCOPE(GROUP) COMMAND CANNOT BE PROCESSED DUE TO COMMAND ALREADY IN PROGRESS

See "DSNX994I" on page 1039.

System action: If no errors occurred, the DISPLAY PROCEDURE command processing continues.

User response: No action is required.

Related reference:

➥ -DISPLAY PROCEDURE (DB2) (DB2 Commands)

➥ -START PROCEDURE (DB2) (DB2 Commands)

➥ -STOP PROCEDURE (DB2) (DB2 Commands)

➥ TIMEOUT VALUE field (STORTIME subsystem parameter) (DB2 Installation and Migration)

DSNX944I   *csect-name* **THE STORED PROCEDURE FUNCTION IS NOT AVAILABLE**

Explanation: This message is returned if stored procedure function is invoked when DB2 is started in ACCESS(MAINT) or RESTART LIGHT mode.

System action: If a START, STOP, or DISPLAY PROCEDURE command is in progress, the command terminates.

DSNX946I   *csect-name* **START PROCEDURE SUCCESSFUL FOR** *token*

Explanation: The START PROCEDURE command completed successfully.

token   The procedure name supplied on the START PROCEDURE command.

System action: The START PROCEDURE COMMAND processing continues.

DSNX947I   *csect-name* **STOP PROCEDURE SUCCESSFUL FOR** *token*

Explanation: The STOP PROCEDURE command completed successfully.

token   The procedure name supplied on the STOP PROCEDURE command

System action: The STOP PROCEDURE command processing continues.

DSNX951I   *csect-name* **A STORED PROCEDURE CANNOT ISSUE START/STOP PROCEDURE COMMANDS**

Explanation: A stored procedure attempted to issue the START or STOP PROCEDURE command. This is not allowed because the START and STOP PROCEDURE commands might cause the stored procedures address space tasks to terminate.

System action: The START or STOP PROCEDURE command is rejected.

DSNX957I   *csect-name* **INSUFFICIENT AUTHORITY FOR OBJECT** *object*

Explanation: A START, STOP, or DISPLAY command is issued for a stored procedure or user-defined function that the user is not authorized to start, stop, or display.

csect-name
> The name of the control section that issued the message.

object   The stored procedure or user-defined function that failed the authorization test.

System action: Processing continues.

System programmer response: If the message is issued at a remote member when SCOPE(GROUP) is specified for the command, verify that the SYSOPR ID is granted the necessary privileges at the remote member.

DSNX960E   *csect-name* **SOME SQL CALL STATEMENTS CANNOT BE PROCESSED, BECAUSE THE LANGUAGE ENVIRONMENT ROUTINE CEEPIPI CANNOT BE LOADED SSN=** *ssn* **PROC=** *jclproc* **ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: To execute an SQL CALL statement, DB2 must load the Language Environment interface module (CEEPIPI). If the CEEPIPI module is not included in an accessible data set such as STEPLIB data sets, JOBLIB data sets, or the LPA library, DB2 is unable to call Language Environment to process stored procedure or function requests.

ssn   The name of the DB2 subsystem to which the stored procedures address space was connected.

ssn   The name of the JCL PROC used to start this stored procedures address space.

asid   The MVS ASID of the address space where the load was issued.

wlmenv   The name of the WLM application environment where the load was issued. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

System action: The address space attempting to process stored procedures or functions terminates.

System programmer response: Change the STEPLIB DD statement in the JCL used to start the stored procedure address space, or install the CEEPIPI module in one of the libraries already accessible to the address space.

After the CEEPIPI module is accessible, restart the failing stored procedure address space. If the error was in a WLM-established stored procedures address space, use the MVS VARY WLM,APPLENV=name,RESUME command.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

DSNX961I   *csect-name* **ATTEMPT TO PERFORM OPERATION** *operation* **FAILED FOR ROUTINE** *routine-name* **SSN=***ssn* **PROC=***jclproc* **ASID=***asid* **CLASS=***class-name* **METHOD=***method-name* **ERROR INFO=***error-information*

Explanation: A WLM-established JAVA stored procedures address space could not perform a necessary operation. The following list contains the possible values for each variable in the message:

operation
> The operation that failed. The following list contains the operations that could have failed:
> - CallStaticMethod
> - CreateDebugJVM
> - CreateJavaVM
> - DEBUG_ROUTINE
> - DestroyJavaVM
> - FindClass
> - INTERNAL_ERROR
> - JDBC Reset

routine-name
> The name of the routine that was invoked.

DSNX961I

ssn
> The name of the DB2 subsystem to which the stored procedures address space is connected.

jclproc
> The name of the JCL procedure that was used to start the stored procedures address space.

asid
> The ASID of the address space in which *operation* was performed.

class-name
> The name of the class that contains the Java method to be invoked.

method-name
> The name of the Java method to be invoked.

error-information
> The error information provided for this error.

System action: Either the CALL statement fails, or, in the case of operations DEBUG_ROUTINE and CreateDebugJVM, the CALL statement executes but the Java stored procedure cannot be debugged.

System programmer response: Use the value of *operation* and this message's *error-information* to correct the problem. If the WLM-managed stored procedure address space is running with a //JSPDEBUG DD card, then, for further information about this error, check its output (typically, the WLM-managed stored procedure address space output) for information.

Also, check the HFS directory identified as the home directory for the ID associated with the WLM-managed stored procedure address space for any dumps that the JVM could have taken. Check either the /tmp/java directory, or if WORK_DIR is specified then its directory, for information that could have been written to file server_stderr.txt.

CallStaticMethod
> This *operation* indicates that an exception has taken place that was not a Java SQLException.

CreateDebugJVM
> This *operation* indicates that the JVM could not be created in debug mode. Debugging of Java stored procedures can only be performed on routines running in a WLM-managed stored procedure address space whose NUMTCB value is 1.

CreateJavaVM
> This *operation* indicates that the JVM could not be created. You must have the IBM Developer Kit for OS/390, Java 2 Technology Edition, and the JAVA_HOME environmental variable in your JAVAENV data set must reference its home directory.

DEBUG_ROUTINE
> This *operation* indicates that the WLM-managed stored procedure address space could not establish or maintain proper communication with a debug session. This message's *error-information* describes how the operation failed and provides an associated reason code. For all cases except for "ReservePort Failure: NO AVAILABLE PORT" record the information in this message's *error-information* and contact your IBM Support Center for assistance. In the case of "ReservePort Failure: NO AVAILABLE PORT" either wait for the simultaneous debugging activity to decrease or increase the number of ports available to the WLM-managed stored procedure address space.

DestroyJavaVM
> This *operation* indicates that the JVM could not be destroyed without error.

FindClass
> This *operation* indicates that either a required system-provided class could not be found or an error occurred in accessing a system- or user-provided class. You need to take a specific action based on what type of class could not be found or accessed:
>
> • For a system-provided class, check the settings for JAVA_HOME, and JCC_HOME in the JAVAENV DD statement in the stored procedures address space start-up procedure, and ensure that they are set correctly.
>
> • For a user-provided class, check the settings for TMSUFFIX and CLASSPATH in the JAVAENV DD statement in the stored procedures address space start-up procedure, and check the routine's use of an installed JAR. The JVM run time loads all classes that are referenced in the identified class. Ensure that all referenced, user-provided, classes are in the CLASSPATH or TMSUFFIX, or in a referenced installed JAR.

INTERNAL_ERROR

This *operation* indicates that an internal error has occurred. Record the following information and contact your IBM Support Center for assistance:

- Any information reported in this message's *error-information*
- The results of the recommended checks of the WLM-managed stored procedure address space
- Any dumps that the JVM could have taken
- Any information written to file `server_stderr.txt`

JDBC Reset

This *operation* indicates that a specific internal error has occurred. Record the following information and contact your IBM Support Center for assistance:

- Any information reported in this message's *error-information*
- The results of the recommended checks of the WLM-managed stored procedure address space
- Any dumps that the JVM could have taken
- Any information written to file `server_stderr.txt`

DSNX962I   *csect-name* **THE Language Environment CEEPIPI SERVICE RETURNED AN UNEXPECTED RETURN CODE** *rc* **FOR FUNCTION** *func* **SSN=** *ssn* **PROC=** *jclproc* **ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: DB2 uses the Language Environment CEEPIPI service to load and execute user-written stored procedures. The Language Environment return code *rc* was not expected for the indicated Language Environment function call.

rc   The return code received from CEEPIPI.

func   The function requested by DB2. Possible values are:

    add_entry
        Add an entry to the Language Environment PIPI table dynamically.

    call_main
        Run a main program stored procedure using an existing Language Environment run time environment.

    call_sub
        Run a sub program stored procedure using an existing Language Environment run time environment.

    delete_entry
        Delete an entry from the Language Environment PIPI table dynamically.

    end_seq
        End a sequence of calls using CEEPIPI.

    identify
        Identify the language the module was written in.

    init_main
        Create Language Environment run time environment.

    init_sub_dp
        Create Language Environment sub run time environment.

    term   Terminate the Language Environment run time environment.

    start_seq
        Start a sequence of calls using CEEPIPI.

ssn   The name of the DB2 subsystem to which the stored procedures address space was connected.

jclproc The name of the JCL PROC used to start this stored procedures address space.

asid   The MVS ASID of the address space where the CEEPIPI error was encountered.

wlmenv The name of the WLM application environment where the CEEPIPI error was encountered. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

DSNX966I • DSNX967I

System action: The attempt to run a stored procedure fails. The DB2 stored procedure address space attempts any new stored procedure requests it receives.

System programmer response: The Language Environment return codes from CEEPIPI are documented in *Language Environment for OS/390 & VM Programming Guide*.

If the values of *func* and *rc* are 'ADD_ENTRY' and X'0000000C', the error might be caused by using a compiler that is not Language Environment enabled to create the stored procedure. DB2 stored procedures must be compiled with a compiler that supports Language Environment.

Problem determination: Collect the following diagnostic items:

*   Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.

DSNX966I   *csect-name* **STORED PROCEDURE OR FUNCTION** *sp* **FAILED. Language Environment ENCLAVE RC =** *erc***, FEEDBACK INFORMATION =** *fi* **SSN=** *ssn* **PROC=** *jclproc* **ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: DB2 uses the Language Environment CEEPIPI call_main or call_sub service to execute user-written stored procedures or user-defined function. The CEEPIPI interface returned information indicating a failure while executing the stored procedure.

sp   The stored procedure or function name that was called using CEEPIPI.

erc   The Language Environment enclave return code.

fi   Contains one or two message identifiers. The message identifiers are constructed from the Language Environment condition token.

The message identifiers are constructed as follows:

*   Single message: Facility_Id combined with Msg_No
*   Two messages: Facility_Id combined with Class_Code and Facility_Id combined with Cause_Code

ssn   The name of the DB2 subsystem that called the stored procedure or function.

jclproc The name of the JCL PROC used to start this stored procedures address space.

asid   The MVS ASID of the address space where the stored procedure executed.

wlmenv   The name of the WLM application environment where the stored procedure executed. If it executed in the DB2-established stored procedures address space, this field will contain blanks.

System action: The attempt to run this stored procedure failed. The DB2 stored procedure address space attempts any new stored procedure requests it receives.

System programmer response: A description of the Language Environment enclave return code and the condition token are documented in *Language Environment for OS/390 & VM Programming Guide*. In many cases the feedback information contains a Language Environment message number. These are documented in z/OS Language Environment.

Problem determination: Collect the following diagnostic items:

*   Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
*   SYSABEND, SYSUDUMP, or SYSMDUMP output.

DSNX967I   *csect-name* **ATTEMPT TO PERFORM WLM FUNCTION** *func* **FAILED WITH WLM RC=** *rc* **RSN=** *rsn* **SSN=** *ssn* **PROC=** *jclproc* **ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: A WLM-established stored procedures address space was unable to perform a WLM operation required to communicate with WLM.

func   The WLM function that failed. Possible values are:

*   IWMCONN
*   IWMSSEL
*   IWMSTBGN
*   IWMSTEND

ssn   The name of the DB2 subsystem to which this stored procedure address space was connected.

jclproc   The name of the JCL PROC used to start this stored procedures address space.

asid   The MVS ASID of the address space where this function was invoked.

wlmenv The name of the WLM application environment where this function was invoked.

System action: The stored procedures service task that encountered the error terminates. If this is the last service task in that address space, then that address space terminates.

The WLM return codes and reason codes are documented in *MVS/ESA SP 5.3 Workload Management Services*. In some cases the failures are caused by the WLM application environment being incorrectly defined or in a quiesced state. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX968I   *csect-name* **STORED PROCEDURE ADDRESS SPACE IS UNABLE TO CONNECT TO WLM BECAUSE WLM_ENV =** *wlmenv* **IS STOPPED OR QUIESCED**

Explanation: A DB2 stored procedures address space could not connect to the Workload Manager (WLM) services of MVS because the WLM application environment was in a stopped or quiesced state.

wlmenv The name of the WLM application environment for which the stored procedure address space was being started.

System action: The stored procedures address space terminates.

System programmer response: Correct the WLM application environment name that is being used by stored procedures address space JCL proc.

To change the status of the application environment use the MVS VARY WLM,APPLENV=wlmenv,RESUME command.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX973I   *csect-name* **START FUNCTION SPECIFIC SUCCESSFUL FOR** *function-name*

Explanation: The function name that was supplied on the START FUNCTION SPECIFIC command completed successfully.

csect-name
        The name of the control section that issued the message.

function-name
        The function name that was supplied on the START FUNCTION SPECIFIC command.

System action: The START FUNCTION SPECIFIC command processing continues.

User response: No action is required.

DSNX974I • DSNX975I

Related reference:

➡ -START FUNCTION SPECIFIC (DB2) (DB2 Commands)

DSNX974I *csect-name* **STOP FUNCTION SPECIFIC SUCCESSFUL FOR** *token*

Explanation: The function name supplied on the STOP FUNCTION SPECIFIC command completed successfully.

token The function name supplied on the STOP FUNCTION SPECIFIC command.

System action: The STOP FUNCTION SPECIFIC command processing continues.

DSNX975I *csect-name* **- DISPLAY FUNCTION SPECIFIC REPORT FOLLOWS-**

Explanation: This message displays output from the DISPLAY FUNCTION SPECIFIC command. The output from this command provides information about the requested external functions and has the following general structure:

```
DSNX975I csect-name - DISPLAY FUNCTION SPECIFIC REPORT FOLLOWS-
------ SCHEMA = schema
FUNCTION STATUS ACTIVE QUED  MAXQ TIMEOUT FAIL WLM_ENV
function status active qued maxq timeout fail wlm_env
function status active qued maxq timeout fail wlm_env
function status active qued maxq timeout fail wlm_env
 FUNCTIONS begin - end STOP (act)

DISPLAY FUNCTION SPECIFIC REPORT COMPLETE
```

In your output, some of these lines might be included more than once and some might not be included at all. Which lines are included depends on the situation, as noted in the following descriptions of the variables in the DISPLAY FUNCTION SPECIFIC output.

If your output ends with "DISPLAY TERMINATED WITH MAX LINES" on page 1035 instead, the report was truncated.

csect-name
> The name of the control section that issued the message.

SCHEMA = *schema*
> The schema name for the functions that follow in the output.
>
> The schema name and lines that follow are repeated as many times as needed to satisfy the DISPLAY FUNCTION SPECIFIC request.

FUNCTION *function*
> The name of the external function.
>
> One line of output is included for each requested function.

STATUS *status*
> The status of the function. The status can be any of the following values:
>
> **STARTED**
>> The function can accept SQL statement requests.
>
> **STOPABN**
>> The function was stopped because of a previous abnormal termination of the function application. SQL statement requests for the function are rejected.
>
> **STOPQUE**
>> The function was stopped. SQL statement requests for the function are queued until the function is started or the timeout interval that is defined by the STORTIME subsystem parameter expires.
>
> **STOPREJ**
>> The function was placed in this status by the STOP SPECIFIC FUNCTION command with the ACTION(REJECT) option. SQL statement requests for the function are rejected.
>
> If the function is in any of the stopped states, you can use the START SPECIFIC FUNCTION command to place the function in STARTED status.

ACTIVE *active*
> The number of DB2 agents that are currently running this function.

QUED *qued*
> The number of DB2 agents that are currently waiting for this function to be scheduled.

MAXQ *maxq*
> The maximum number of DB2 agents that have waited for this function to be scheduled since the last START SPECIFIC FUNCTION command was issued for this function.

TIMEOUT *timeout*
> The number of times that an SQL statement timed out while waiting for the function to be scheduled since the last START SPECIFIC FUNCTION command was issued for this function.
>
> An SQL statement can time out if the function is stopped by the STOP SPECIFIC FUNCTION command with the ACTION(QUEUE) option. This specification causes requests for the function to time out after the interval that is defined by the STORTIME subsystem parameter.

FAIL *fail*
> The number of times that this function has abended since it was last started.

WLM_ENV *wlm_env*
> The MVS Workload Manager environment name for this function.

FUNCTIONS *begin* **-** *end* **STOP (***act***)**
> This line is displayed when a STOP SPECIFIC FUNCTION command caused a range of functions to be stopped. This situation can occur when the command was issued with an asterisk (*) in the function name.
>
> *begin* The first function name in the range of functions that are currently stopped.
>
> *end* The last function name in the range of functions that are currently stopped.
>
> *act* The action that was specified on the STOP SPECIFIC FUNCTION command.

DISPLAY FUNCTION SPECIFIC REPORT COMPLETE
> The report completed successfully.

DISPLAY TERMINATED WITH MAX LINES:

This line indicates that the DISPLAY FUNCTION SPECIFIC report was truncated. The report was generated in response to a command from an MVS console, and more than 254 response messages were generated. Only 254 response messages are returned.

Command processing completes. The truncated report is written to the MVS console.

Reissue the DISPLAY FUNCTION SPECIFIC command and specify a smaller number of routines. Specify a list of routines, a schema qualifier, or a partial routine name in the DISPLAY command rather than the default (*.*). Alternatively, reissue the command from the DB2 Commands DB2I panel or from a batch job rather than from the MVS console.

System action: The DISPLAY FUNCTION SPECIFIC command processing continues.

Related reference:

➡ -DISPLAY FUNCTION SPECIFIC (DB2) (DB2 Commands)

➡ -START FUNCTION SPECIFIC (DB2) (DB2 Commands)

➡ -STOP FUNCTION SPECIFIC (DB2) (DB2 Commands)

➡ TIMEOUT VALUE field (STORTIME subsystem parameter) (DB2 Installation and Migration)

DSNX978I   **DISPLAY TERMINATED DUE TO INSUFFICIENT MESSAGE SPACE**

Explanation: The DISPLAY PROCEDURE command could not obtain storage for additional messages. This situation occurs for only long display requests, such as one that requests all stored procedures that have been accessed by DB2 applications since DB2 was started.

System action: The display is truncated.

System programmer response: Reissue the DISPLAY PROCEDURE command, and specify a smaller number of

procedures. For example, specify a partial procedure name in the DISPLAY PROCEDURE command rather than the default (*).

DSNX980E   *csect-name* **ATTEMPT TO PERFORM CALL ATTACH FUNCTION** *function-name* **FAILED WITH DB2 SUBSYSTEM** *subsystem-id*. **RETURN CODE=***caf-return-code*, **REASON CODE=***reason-code*

Explanation: The DB2 stored procedures address space was unable to perform a DB2 call attach operation required to communicate with DB2.

csect-name
    The name of the control section that issued the message.

function-name
    The DB2 call attach function that failed. Possible values are:
    • CONNECT
    • OPEN
    • CLOSE
    • DISCONNECT

subsystem-id
    The DB2 subsystem identifier.

caf-return-code
    The DB2 call attach return code

reason-code
    The DB2 reason code

System action: The stored procedures service task that encountered the error terminates. If this is the last service task in the address space, the DB2 stored procedures address space terminates.

System programmer response: Analyze the DB2 call attach return code and the reason code, and take the appropriate action.

Related tasks:

⤷ Invoking the call attachment facility (DB2 Application programming and SQL)

Related reference:

⤷ CAF return codes and reason codes (DB2 Application programming and SQL)

DSNX981E   *csect-name* **THE PARAMETER** *parm* **CONTAINS AN INVALID VALUE "***string***" PROC=** *jclproc*

Explanation: The DB2 stored procedures address space received an invalid value in the JCL parameter used to start the address space.

parm   The JCL parameter is specified incorrectly. Possible values are:

SUBSYS
        This parameter can be invalid for the following reasons:
        • The SUBSYS parameter is null.
        • The SUBSYS parameter is too long.
        • The SUBSYS parameter contained an invalid character.

NUMTCB
        This parameter can be invalid for the following reasons:
        • A comma was not present in the parameter list to separate the NUMTCB parameter from the previous parameter.
        • The NUMTCB parameter is null.
        • The NUMTCB parameter contained nonnumeric characters.
        • The value of the NUMTCB parameter is zero.
        • The value of the NUMTCB parameter is negative.

- The value of the NUMTCB parameter is greater than the maximum number of TCBs allowed in the DB2 stored procedures address space.

APPLENV

This parameter can be invalid for the following reasons:

- A comma was not present in the parameter list to separate the APPLENV parameter from the previous parameter.
- The value of the APPLENV parameter is longer than eighteen characters.

string   The value provided for the parameter in error.

jclproc   The name of the JCL PROC used to start this stored procedures address space.

System action: The DB2 stored procedures address space terminates.

If MVS WLM is used to establish the stored procedure address space, WLM will retry the start of the address space two more times. If all three attempts fail, WLM will quiesce the WLM environment associated with the stored procedure address space.

System programmer response: Correct the JCL parameter used to invoke the DB2 stored procedure address space.

If MVS WLM is used to establish the stored procedure address space, the MVS VARY WLM,APPLENV=*wlmenv*,RESUME command might be required to resume the WLM application environment (*wlmenv*) associated with the stored procedures address space.

DSNX982I   *csect-name* **ATTEMPT TO PERFORM RRS ATTACH FUNCTION** *func* **FAILED WITH RRS ATTACH RC=** *rc* **RSN=** *rsn* **SSN=** *ssn* **PROC =** *jclproc* **ASID=** *asid* **WLM_ENV=** *wlmenv*

Explanation: A WLM-established stored procedures address space was unable to perform an RRS attach operation required to communicate with DB2.

func   The RRS attach function that failed. Possible values are:

- SPAS_ID
- SPAS_INIT_SP
- SPAS_TERM_SP

ssn   The name of the DB2 subsystem involved with the RRS attach function.

jclproc   The name of the JCL PROC used to start the stored procedures address space.

asid   The MVS ASID of the address space where the function was attempted.

wlmenv   The name of the WLM application environment where the function was attempted.

System action: The stored procedures service task that encountered the error terminates. If this is the last service task in that address space, then that address space terminates.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- SYSABEND, SYSUDUMP, or SYSMDUMP output.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX983I   *csect-name* **DB2 STORED PROCEDURE SUPPORT CODE AND SUBSYSTEM CODE ARE NOT AT THE SAME LEVEL SSN=** *ssn* **PROC =** *jclproc*

Explanation: A DB2 stored procedures address space could not connect to the DB2 subsystem. This occurred because the DB2 code in the stored procedures address space was not at the same release level as the code in the DB2 subsystem.

ssn   The name of the DB2 subsystem that the stored procedure address space was attempting to connect with.

jclproc   The name of the JCL PROC used to start the stored procedures address space.

System action: The stored procedures address space terminates.

System programmer response: Correct the DB2 library name in the STEPLIB for the JCL proc for the stored procedures address space. Make the library name the same as the library name used for the DB2 subsystem address spaces. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: Collect the following diagnostic items:

* Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
* SYSABEND, SYSUDUMP, or SYSMDUMP output.
* Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSNX984I   *csect-name* **ALL FUNCTIONS STARTED FOR THIS SCHEMA.**

Explanation: All functions are started for the schema listed in the preceding DSNX956I message. This message is issued only when STOP FUNCTION SPECIFIC(*.*) was in effect.

System action: The DISPLAY SPECIFIC FUNCTION command processing continues.

DSNX986I   **ATTEMPT TO** *func* **FAILED WITH RV =** *rv* **RC =** *rc* **RSN =** *rsn* **SSN =** *ssn* **PROC =** *jclproc*

Explanation: A WLM-established stored procedures address space was unable to register itself to z/OS UNIX System Services. z/OS UNIX System Services are not available to stored procedures running in the address space.

func   The dub setting used by z/OS UNIX System Services BPX1SDD callable service. This can be DUBPROCESS or DUBTHREAD.

rv   The return value given by z/OS UNIX System Services BPX1SDD callable service.

rc   The return code given by z/OS UNIX System Services BPX1SDD callable service.

rsn   The reason code given by z/OS UNIX System Services BPX1SDD callable service.

ssn   The name of the DB2 subsystem to which this stored procedure address space was connected.

proc   The name of the JCL PROC used to start this stored procedures address space.

System action: The stored procedure address space initialization continues. z/OS UNIX System Services callable services' return values, return codes, and reason codes are documented in *z/OS UNIX System Services Messages and Codes*.

Problem determination: Most failures are due to the user ID associated with the WLM-established stored procedures address space not being RACF defined. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

It is also possible that the number of processes was exceeded for the user ID. If the NUMBER OF TCBS is set too high during install, NUMTCB may exceed the value of the UNIX System Services MACPROCUSER setting. Refer to *z/OS UNIX System Services Messages and Codes* for more information.

DSNX987I   *csect-name* **NAME** *schema-name* **INVALID, COMMAND IGNORED.**

Explanation: The function name or procedure name supplied on the DISPLAY, START, or STOP, FUNCTION SPECIFIC or PROCEDURE command uses a invalid wildcard schema name. The qualified routine name must specify a full schema name, "*", "*.*", or blanks.

schema-name
     The schema name containing invalid characters.

System action: The START, STOP, or DISPLAY command is rejected.

DSNX988I   *csect-name* **ALL OTHER FUNCTIONS IN OTHER SCHEMAS STOPPED** *act*

Explanation: A STOP FUNCTION SPECIFIC(*.*) is currently in effect. Any functions not listed in the preceding DSNX969, DSNX971, and DSNX976 messages are stopped *act*.

act   The action provided on the STOP FUNCTION command

System action: The DISPLAY SPECIFIC FUNCTION command processing continues.

DSNX990E   *csect-name* **IMS ODBA INITIALIZATION FAILED BECAUSE THE CALLABLE INTERFACE CANNOT BE LOADED**

Explanation: The IMS ODBA callable interface module cannot be loaded. No IMS database DL/I calls will be supported from stored procedures. Ensure the load module named AERTDLI is located within the partitioned data set that is specified in the stored procedure address space startup procedure's DFSRESLB DD card statement.

System action: The stored procedure address space initialization continues.

DSNX991I   *csect-name* **IMS ODBA INITIALIZATION COMPLETED**

Explanation: The initialization of the IMS ODBA callable interface has completed. Stored procedure applications may use DL/I calls to access IMS databases.

System action: The stored procedure address space initialization continues.

DSNX992E   *csect-name* **IMS ODBA INITIALIZATION FAILURE, AIB RC =** *rc*

Explanation: The initialization of the IMS ODBA callable interface has failed. Stored procedure applications cannot use DL/I calls to access IMS databases.

rc   The AIB return code provided by IMS for the CIMS INIT call. Refer to the section on DL/I Return and Reason Codes in *IMS Application Programming: Database Manager* for more information regarding the failure.

System action: The stored procedure address space initialization continues.

DSNX993I   *csect-name* **CALL TO REXX PROCEDURE** *execname* **FAILED, FUNCTION =** *function* **RC =** *rc* **RSN =** *rsn*

Explanation: An attempt to establish an environment to run a REXX stored procedure has failed. Use the following pieces of information to identify the error:

execname
        The name of the REXX exec.

function The TSO or REXX function call that failed. It can be one of three values: IKJTSOEV, IRXINIT, or IRXEXEC. For more information regarding the IKJTSOEV function, refer to *z/OS TSO/E Programming Services*. For information regarding IRXINIT and IRXEXEC, refer to *z/OS TSO/E REXX Reference*

rc   The return code for the specific function, provided by TSO or REXX. For return code explanations, refer to *z/OS TSO/E Programming Services* and *z/OS TSO/E REXX Reference*.

reason   The reason code for the specified function, provided by TSO or REXX. For reason code explanations, refer to *z/OS TSO/E Programming Services* and *z/OS TSO/E REXX Reference*.

System action: The stored procedure address space initialization continues.

Problem determination: View the file associated with the SYSTSPRT DD card in the Stored Procedure Address Space start-up procedure. This file contains diagnostic messages issued by TSO regarding the failing function. Refer to *z/OS TSO/E Messages* for any messages found in the SYSTSPRT.

DSNX994I   **SCOPE(GROUP) COMMAND CANNOT BE PROCESSED DUE TO COMMAND ALREADY IN PROGRESS**

Explanation: One of the following commands with the SCOPE(GROUP) option was issued at the same time that another data sharing member was processing a SCOPE(GROUP) command:

• DISPLAY PROCEDURE

DSNX994I

- START PROCEDURE
- STOP PROCEDURE
- DISPLAY FUNCTION
- START FUNCTION
- STOP FUNCTION

DB2 could not obtain the lock necessary to serialize these commands.

System action: The command is not processed on any members of the data sharing group.

User response: Enter the command again.

Severity: 12 (severe error)

Related reference:

 DB2 commands (DB2 Commands)

Chapter 20. DSNY messages

Message identifiers that begin with DSNY communicate information about initialization procedures.

DSNY001I   **SUBSYSTEM STARTING**

Explanation: The -START DB2 command is accepted. This message is issued to the MVS console that issued the -START command. The DSN9022I message is sent when the DB2 subsystem startup process has completed. Refer to the DSN9022I message for additional information.

System action: DB2 startup processing begins.

DSNY002I   **SUBSYSTEM STOPPING**

Explanation: The -STOP DB2 command is accepted. The DSN9022I message is issued when the DB2 subsystem shutdown process has completed. The message is issued either to the origin of the -STOP DB2 command (for MVS MCS console) or to the MVS console from which the -START DB2 command was received. Refer to the DSN9022I message for additional information.

System action: DB2 subsystem shutdown is initiated.

DSNY003I   **SUBSYSTEM IS ALREADY ACTIVE**

Explanation: Either a START DB2 or REFRESH DB2, EARLY command was issued while DB2 is active and was not accepted. These commands are allowed and can be processed correctly only when DB2 is inactive.

The DSN9023I message is issued after this message and contains additional information.

Related reference:

➡ -START DB2 (DB2) (DB2 Commands)

➡ -REFRESH DB2,EARLY (DB2) (DB2 Commands)

Related information:

DSNY004I   **SUBSYSTEM IS ALREADY STOPPING**

Explanation: The -STOP DB2 command has not been accepted either because the DB2 subsystem shutdown is in progress for the specified option (QUIESCE or FORCE), or because the QUIESCE option was specified after a FORCE option had been previously accepted. The DSN9023I message is issued after this message. Refer to the DSN9023I message for additional information.

System action: DB2 subsystem shutdown is continued.

DSNY005I   **SUBSYSTEM STARTUP TERMINATED, INVALID START COMMAND**

Explanation: The DB2 subsystem can be started only by a -START DB2 command.

System action: DB2 subsystem startup is terminated.

Operator response: Start the DB2 subsystem via the -START DB2 command, and reenter the rejected command.

DSNY006I   *csect-name* **INVALID AMODE OR RMODE ATTRIBUTE FOUND FOR LOAD MODULE** *module name*

Explanation: The Initialization Procedures subcomponent detected that a module had an invalid AMODE or RMODE attribute when it was loaded. In the error message, 'module name' is the name of the load module with an invalid addressing or residency mode.

System action: DB2 subsystem startup is terminated.

Operator response: Notify the system programmer of the failure.

System programmer response: Verify that all installation and maintenance activities against DB2 were executed using SMP, that the JCLIN for SMP includes the correct AMODE and RMODE control statements, and that the Linkage Editor provided with the Data Facility Product (DFP) was invoked by SMP for the link edits. Correct the procedure or JCL that caused the error and rerun the necessary SMP jobs.

Problem determination: A DB2 dump was requested to an MVS SYS1.DUMP data set. The MVS service aid AMBLIST provides Linkage Editor module attributes in the summary section of the LISTLOAD function output.

DSNY007I   LOAD MODULE *module-name* **DOES NOT HAVE RMODE(24) ATTRIBUTE**

Explanation: The named load module was not link-edited with the required RMODE(24) attribute.

System action: DB2 startup is abended with reason code 00E80160.

Operator response: Notify the system programmer.

System programmer response: Verify that all installation and maintenance activities against the specified load module were executed using SMP. JCLIN for SMP must have included the correct AMODE and RMODE control statements. The Linkage Editor, provided with the MVS, was invoked by SMP for the link-edits. Correct the procedure or JCL that caused the error and rerun the necessary SMP jobs.

Problem determination: The MVS service aid AMBLIST provides Linkage Editor module attributes in the summary section of the LISTLOAD function output.

DSNY008I   SUBSYSTEM SHUTDOWN REQUEST NOT ACCEPTED

Explanation: The -STOP DB2 command has not been accepted because a subsystem startup has not completed to the point where shutdown can occur. The DSN9023I message is issued after this message. Refer to the DSN9023I message for additional information.

System action: DB2 startup continues and the -STOP DB2 command is ignored.

User response: Reissue the -STOP DB2 command after subsystem startup has completed.

DSNY009I   *csect-name* SUBSYSTEM STARTING IN LIGHT MODE, NORMAL TERMINATION TO FOLLOW RELEASE OF RETAINED LOCKS IF NO INDOUBTS

Explanation: DB2 issues this message during DB2 startup if it is restarted in light mode.

System action: DB2 startup continues normally. The DB2 member will automatically shut itself at the end of restart after the retained locks are removed if there are no indoubt units of recovery. If indoubt units of recovery exist, the DB2 member remains up and running in light mode so that the indoubts can be resolved.

DSNY010I   LOAD MODULE module name IS NOT AT THE CORRECT RELEASE/VERSION LEVEL

Explanation: The named load module is not at the correct release/version level for the version of DB2 that was being started.

System action: DB2 startup is abended with reason code 00E80161.

Operator response: Notify the system programmer.

System programmer response: Verify that the correct load libraries are being used. Also verify that the application of maintenance did not result in modules being link edited into the wrong load libraries.

DSNY011I   *csect-name* Z/OS DOES NOT HAVE SUFFICIENT 64-BIT SHARED AND COMMON STORAGE AVAILABLE

Explanation: Insufficient contiguous shared virtual space or 64-bit common storage remains. For shared virtual space, at least 1 TB of contiguous storage is required for each DB2 member. For 64-bit common storage, 6 GB of contiguous storage is required for each DB2 member.

csect-name
   The name of the control section that issued the message.

| **System action:** DB2 startup terminates with reason code 00E8005A.

| **System programmer response:** For each Version 11 or later DB2 database that you plan to start on each z/OS
| image, verify that you have the following items:
| • A minimum of 1 TB of contiguous shared storage
| • 6 GB of contiguous 64-bit common storage
| **Related information**:

| ➡ 00E8005A (DB2 Codes)

| ➡ IEASYSxx (symbol definitions and IEASYSxx members) (MVS Initialization and Tuning Reference)

DSNY012I   *csect-name* ******** **THIS RELEASE OF DB2 REQUIRES Z/OS** *os-version* **OR ABOVE**

Explanation: An attempt was made to start a DB2 subsystem on an operating system image that does not meet the
minimum version requirement for z/OS.

csect-name
        The name of the control section that issued the message.

os-version
        The minimum version of z/OS that is required for DB2 11 for z/OS.

System action: DB2 startup abnormally terminates with reason code 00E80058.

System programmer response: Ensure that the required version of z/OS is enabled.

Related concepts:

➡ Hardware and software requirements (DB2 Installation and Migration)

Related information:

➡ 00E80058 (DB2 Codes)

➡ DB2 Program Directory

DSNY014I   *n csect* **DB2 HAS BEEN STARTED WITH ACCESS(MAINT)**

Explanation: This DB2 has been started with option ACCESS(MAINT).

System action: None.

Programmer response: None.

Severity: 0 (informational)

DSNY015I   **LIGHT(YES) ON START DB2 COMMAND WAS IGNORED, SYSTEM IS NOT ENABLED FOR
                DATA SHARING**

Explanation: DB2 issues this message during startup if DB2 is started with the LIGHT(YES) keyword and the
system is not enabled for data sharing.

System action: DB2 startup continues normally, ignoring the LIGHT(YES) keyword.

| **DSNY016I**   *csect-name* **SUBSYSTEM IS STARTING IN LIGHT MODE. NORMAL TERMINATION FOLLOWS
| RELEASE OF RETAINED LOCKS. INDOUBT UNITS OF RECOVERY ARE IGNORED.**

| **Explanation:** DB2 issues this message during DB2 startup processing if the LIGHT(NOINDOUBTS) option was
| specified on the START DB2 command.
| *csect-name*
| The name of the control section that issued the message.

| **System action:** DB2 startup continues normally. The DB2 member automatically shuts itself down at the end of
| restart processing after the retained locks are removed. The DB2 member ignores indoubt units of recovery.

| **User response:** No action is required.

| **Related concepts**:

| [→] Restart light (DB2 Data Sharing Planning and Administration)

| **Related reference**:

| [→] -START DB2 (DB2) (DB2 Commands)

DSNY020I   *csect-name* **MEASURED USAGE DATA IS NOT BEING RECORDED FOR** *subsystem/address-space*

Explanation: For measured usage data to be recorded, SMF type 89 records must be collected. Two PTFs must be applied: one to the early code and one to the base. Until both are active, measured usage data cannot be recorded.

System action: Processing continues, but measured usage for either the DB2 subsystem or the specified DB2 address space is not recorded.

Programmer response: If measured usage data is needed, be sure that SMF type 89 records are collected and that both PTFs are applied to the running system. An IPL is required for the change to early code.

If the records are being recorded, the PTFs are applied, and an address-space is in the message, then the return code for the MVS service is located in ASCEMULC for that address space.

DSNY024I   *facility* **INITIALIZATION IS STARTING**

Explanation: The facility that you specified is starting.

facility is MSTR, DBM1, or DIST address spaces.

System action: The facility is starting.

DSNY025I   *facility* **SHUTDOWN IS COMPLETE**

Explanation: The specified utility has completed shutdown processing.

facility is MSTR, DBM1, or DIST address spaces.

System action: The facility has stopped.

| **DSNY027I**   *csect-name processor-type* **IS THE MINIMUM LEVEL OF PROCESSOR SUPPORTED BY THIS**
| **RELEASE OF DB2**

| **Explanation:**   An attempt was made to start a DB2 subsystem on a processor that does not meet the minimum requirement for this release of DB2.

| *csect-name*
| The name of the control section that issued the message.

| *processor-type*
| The minimum processor level that is required by DB2 11 for z/OS.

| **System action:**   DB2 startup abnormally terminates with reason code 00E8005C.

| **System programmer response:**   Start this DB2 subsystem on an appropriate processor or start a subsystem that is on previous release of DB2 on this processor.

| **Related concepts**:

| [→] Hardware and software requirements (DB2 Installation and Migration)

| **Related information**:

| [→] 00E8005C (DB2 Codes)

| [→] DB2 Program Directory

Chapter 21. DSNZ messages

Message identifiers that begin with DSNZ communicate information about the system parameter manager.

DSNZ002I   **src DSNZINIT SUBSYSTEM ssnm SYSTEM PARAMETERS LOAD MODULE NAME IS dsnzparm-name**

Explanation: DB2 subsystem 'ssnm' is being started with the 'dsnzparm-name' DSNZPARM load module. This message is written to the MVS console which entered the -START DB2 command. The message will also appear in the JES SYSLOG and on the JES job log of *ssnm*MSTR. The message contains the name of the DSNZPARM load module entered by the operator or the default name (DSNZPARM) if no name was entered.

System action: DB2 startup processing continues.

DSNZ003I   *csect-name* **DSNZINIT CSECT** *parm-csect* **NOT FOUND IN SYSTEM PARAMETERS.**

Explanation: The named *parm-csect* was not found in the system parameters (ZPARM) specified when starting DB2. This CSECT is required to start DB2. The name of the system parameters load module that is missing this CSECT is identified in previously issued message DSNZ002I.

This error can occur if the system parameters load module specified is for a different release of DB2.

System action: DB2 subsystem startup processing stops.

Operator response: Verify that the correct system parameters name was specified. If the correct name was specified, notify the system programmer.

System programmer response: Verify that:
- The system parameters load module is for the release of DB2 being started.
- The correct load libraries are being used to load the system parameters load module.

If these are correct, then reassemble and relink-edit your system parameters load module.

DSNZ004I   *csect-name* **DSNZINIT INCORRECT LEVEL** *wrong-level* **FOUND IN** *parm-csect* **SYSTEM PARAMETERS. LEVEL** *correct-level* **IS REQUIRED.**

Explanation: The system parameters load module CSECT *parm-csect* contains an incorrect level identifier. The CSECT contains a level identifier of *wrong-level* and the DB2 that is starting requires a level identifier of *correct-level*.

This error can occur if the system parameters load module specified is for a different release of DB2. This error can also occur if the system parameters load module is missing some maintenance. In this case the *parm-csect* is down level and must be reassembled.

System action: DB2 subsystem startup processing stops.

Operator response: Verify that the correct system parameters name was specified. If the correct name was specified, then notify the system programmer.

System programmer response: Verify that:
- The system parameters load module is for the release of DB2 being started.
- The correct load libraries are being used to load the system parameters load module.

If these are correct, then reassemble and relink-edit your system parameters load module.

DSNZ006I   *src* **SUBSYS** *ssnm* **SYSTEM PARAMETERS LOAD MODULE NAME** *module-name* **IS BEING LOADED**

Explanation: DB2 subsystem *ssnm* is loading parameters from *module-name* subsystem parameters load module.

System action: DB2 online refresh of subsystem parameters continues.

DSNZ007I   *src* **SUBSYS** *ssnm* **SYSTEM PARAMETERS LOAD MODULE NAME** *module-name* **LOAD COMPLETE**

Explanation: DB2 subsystem *ssnm* has loaded subsystem parameters load module *module-name*, and has notified all resource managers.

DSNZ008I   *src* **CSECT** *parm-csect* **NOT FOUND IN SYSTEM PARAMETERS**

Explanation: The named *parm-csect* was not found in the subsystem parameters load module specified in the -SET SYSPARM command. Message DSNZ007I, which was issued prior to this message, identifies the name of the subsystem parameters load module that is missing this CSECT.

System action: DB2 online refresh of subsystem parameters terminates.

System programmer response: Verify that:

- The system parameters load module matches the release level of the DB2 that was started.
- The correct load libraries are being used to load the system parameters load module.

If these are correct, then reassemble and relink-edit your system parameters load module.

DSNZ009I   *src* **INCORRECT LEVEL** *wrong-level* **FOUND IN** *parm-csect* **SYSTEM PARAMETERS. LEVEL** *correct-level* **IS REQUIRED.**

Explanation: This message, issued during -SET SYSPARM command processing, indicates the specified subsystem parameters load module CSECT *parm-csect* contains an incorrect level identifier. The CSECT contains a level identifier of *wrong-level*.

The active DB2 subsystem requires a level identifier of *correct-level*. This error can occur if the specified subsystem parameters load module does not match the release level of the DB2 system. This error can also occur if the subsystem parameters load module requires additional maintenance. In the latter case, the *parm-csect* is down-level and requires reassembly.

System action: DB2 online refresh of subsystem parameters terminates.

DSNZ010I   *src* **SUBSYS** *ssnm* *module-type* **LOAD MODULE NAME** *module-name* **NOT FOUND, BLDL R15=***return-code***, R0=***reason-code*

Explanation: During START DB2 or SET SYSPARM command processing on subsystem *ssnm*, the specified subsystem parameters load module *module-name* could not be located during MVS BLDL processing. The BLDL issued a return code of *return-code* and a reason code of *reason-code*.

src   The DB2 command prefix.

ssnm   The subsystem name.

module-name
         The name of the module.

return-code
         The return code issued by the command.

reason-code
         The DB2 reason code.

System action: DB2 startup or online refresh of subsystem parameters terminates.

System programmer response: Verify that the named load module exists, and that DB2 can access it.

Related reference:

⮕ -START DB2 (DB2) (DB2 Commands)

⮕ -SET SYSPARM (DB2) (DB2 Commands)

DSNZ011I   **src SUBSYS ssnm SYSTEM PARAMETERS SET TO STARTUP**

Explanation: DB2 issues this message when the STARTUP parameter is specified on the -SET SYSPARM command. The message indicates that the subsystem parameters specified at DB2 startup are now in effect.

System action: Processing continues.

DSNZ012I   *src* **SUBSYS** *ssnm* **SYSTEM PARAMETERS LOAD MODULE NAME** *module-name* **NOT LOADED,** **LOAD R15=***return-code***, R1=***reason-code*

Explanation: DB2 issues this message during START DB2 or SET SYSPARM command processing on subsystem *ssnm*, where the load of *module-name* failed. *return-code* and *reason-code* are the return and reason codes from the MVS LOAD.

src   The DB2 command prefix.

ssnm   The DB2 subsystem name.

module-name
> The name of the module.

return-code
> The return code issued by the command.

reason-code
> The DB2 reason code.

System action: DB2 startup or online refresh of subsystem parameters terminates.

Related reference:

⮕ -START DB2 (DB2) (DB2 Commands)

⮕ -SET SYSPARM (DB2) (DB2 Commands)

DSNZ013I   *src* **SUBSYS** *ssnm* **SYSPARM COMMAND REJECTED, COMMAND ALREADY IN PROGRESS**

Explanation: The current -SET SYSPARM command is rejected due to processing of a previous -SET SYSPARM command.

System action: Online refresh of subsystem parameters for this command is rejected.

User response: Retry the command after the previous -SET SYSPARM command completes.

DSNZ014I   *src* **PARAMETER** *parm-name* **IN CSECT** *load-csect-name* **CANNOT BE CHANGED ONLINE.** **PARAMETER CHANGE IGNORED.**

Explanation: Subsystem parameter *parm-name* in *load-csect-name* is different from its start-up value, but *parm-name* cannot be changed online.

System action: DB2 uses the start-up value for parameter *parm-name*.

DSNZ015I   *source* **YOU MUST HAVE** *authority* **AUTHORITY TO CHANGE PARAMETER** *parameter-name* **ONLINE. PARAMETER CHANGE IS IGNORED.**

Explanation: The SET SYSPARM command was issued with insufficient authority to change the parameter.

authority
> The authority that is required to make the change.

parameter-name
> The name of the parameter that SET SYSPARM was supposed to change.

The previous value for *parameter-name* is used.

System action: The value for the *parameter-name* parameter is not changed in the newly-loaded module.

User response: Reissue the command with sufficient authority.

DSNZ016I   *src* **DSNZINIT SUBSYSTEM** *ssnm* **SYSTEM PARAMETERS ARE AS FOLLOWS: SYSTEM PARM** *zparm-name* **DSNHDECP** *decp-name* **ACCESS CONTROL EXIT** *access-control-exit-name* **IDAUT EXIT** *identity-auth-exit-name* **SIGNONEXIT** *signon-exit-name*

Explanation: DB2 subsystem *ssnm* is being started with the following parameters specified, either implicitly or explicitly:

zparm-name
> The fully qualified data set name where the subsystems parameters were found.

decp-name
> The fully qualified data set name where the application programming defaults, commonly called the DECP parameters, were found.

access-control-exit-name
> The fully qualified data set name where the Access Control exit was found.

identity-auth-exit-name
> The fully qualified data set name where the Identity or Authorization exit was found.

signon-exit-name
> The fully qualified data set name where the SIGNON exit was found.

System action: DB2 startup processing continues.

Related reference:

➡ -START DB2 (DB2) (DB2 Commands)

➡ -SET SYSPARM (DB2) (DB2 Commands)

DSNZ017I   *src* **DSNZINIT SUBSYSTEM SYSTEM** *subsystem-id* **DATE AND TIME EXITS ARE AS FOLLOWS: ASCII DATE** *ascii-date-exitname* **EBCDIC DATE** *ebcdic-date-exitname* **UNICODE DATE** *unicode-date-exitname* **ASCII TIME** *ascii-time-exitname* **EBCDIC TIME** *ebcdic-time-exitname* **UNICODE TIME** *unicode-time-exitname*

Explanation: The DB2 subsystem uses the specified exit modules:

src   The DB2 command prefix.

subsystem-id
> The DB2 subsystem identifier.

ascii-date-exitname
> The fully-qualified name of the ASCII date exit module DSNXVDTA

ebcdic-date-exitname
> The fully-qualified name of the EBCDIC date exit module DSNXVDTX

unicode-date-exitname
> The fully-qualified name of the Unicode date exit module DSNXVDTU

ascii-time-exitname
> The fully-qualified name of the ASCII time exit module DSNXVTMA

ebcdic-time-exitname
> The fully-qualified name of the EBCDIC time exit module DSNXVTMX

unicode-time-exitname
> The fully-qualified name of the Unicode time exit module DSNXVTMU

System action: Processing continues.

User response: No action is required.

Related reference:

□➡ -START DB2 (DB2) (DB2 Commands)

□➡ -SET SYSPARM (DB2) (DB2 Commands)

Chapter 22. DSN1 messages

Message identifiers that begin with DSN1 communicate information about service facilities.

These messages apply to DSN1LOGP, DSN1COPY, DSN1COMP, and DSN1PRNT.

The value shown for 'severity' is the value returned as the job-step condition code from the job step in which the message is issued. If additional messages having higher severity values are issued during the same job step, the higher value is reflected as the job-step condition code.

Severity values have the following meanings:

0   An informational message. No error occurred.

4   A warning message. A condition was detected of which the user should be aware. The user might need to take further action.

8   An error message. An error situation was detected and processing could not continue.

12   A severe error message. A severe error situation was detected and processing could not continue.

DSN1110E   LIMIT OF 50 STATEMENTS EXCEEDED

Explanation: DSN1LOGP allows a total of 50 statements for a given job.

System action: Processing is terminated.

User response: Resubmit the job using no more than 50 statements.

Severity: 8 (error)

DSN1111E   LIMIT OF 80 TOKENS EXCEEDED

Explanation: DSN1LOGP allows a total of 80 keywords and corresponding value specifications. A keyword with its value is considered two tokens.

System action: Processing is terminated.

User response: Resubmit the job using no more than 80 keywords.

Severity: 8 (error)

DSN1112E   TOKEN xx..xx EXCEEDS 64 HEXADECIMAL CHARACTERS

Explanation: The SYSIN input contains the character string 'xx..xx'. This string is invalid.

System action: Processing is terminated.

User response: Resubmit the job with the specified token being no more than 64 hexadecimal characters in length.

Severity: 8 (error)

DSN1113E   **VALUE FOR KEYWORD xxxxxxxx INVALID**

Explanation: The SYSIN input contains the keyword 'xxxxxxxx'. The value specified for that keyword is not of the form '(value)'.

System action: Processing is terminated.

User response: Resubmit the job with the correct form of the keyword. The problem might be that the closing parenthesis ')' is missing or that an extra blank exists in the value.

Severity: 8 (error)

DSN1114E   **RBASTART MULTIPLY DEFINED**

Explanation: The RBASTART parameter can be specified only once for a given job.

System action: Processing is terminated.

User response: Resubmit the job using a single RBASTART value.

Severity: 8 (error)

DSN1115E   *csect-name* **KEYWORD** *keyword* **ALLOWS ONLY** *number* **HEX CHARACTERS**

Explanation: The value given for *keyword* is too long.

System action: Processing is terminated.

User response: Resubmit the job providing no more than the maximum number of characters allowed for the keyword.

Severity: 8 (error)

DSN1116E   *csect-name keyword* **IS USED MORE THAN ONCE.**

Explanation: The *keyword* can be used only once in a single job.

System action: Processing is terminated.

User response: Resubmit the job providing a single keyword.

Severity: 8 (error)

DSN1117E   *csect-name keyword* **SPECIFIED WITH NO VALUE OR INCORRECT VALUE**

Explanation: The keyword must be followed by a correct value.

System action: Processing is terminated.

User response: Resubmit the job providing a correct value for the keyword.

Severity: 8 (error)

DSN1118E   **RBAEND > 12 HEX CHARACTERS**

Explanation: The maximum size of a log RBA in DB2 is 6 bytes (hence, 12 characters).

System action: Processing is terminated.

User response: Resubmit the job providing an RBAEND value that is no more than 12 hexadecimal characters.

Severity: 8 (error)

DSN1119E   **DBID MULTIPLY DEFINED**

Explanation: The DBID keyword can be specified only once for a given job.

System action: Processing is terminated.

User response: Resubmit the job providing a single DBID value.

Severity: 8 (error)

DSN1120E   DBID SPECIFIED WITH NO VALUE

Explanation: The DBID keyword specifies a hexadecimal database identifier and must be followed by a value.

System action: Processing is terminated.

User response: Resubmit the job providing a DBID (value).

Severity: 8 (error)

DSN1121E   DBID > 4 HEX CHARACTERS

Explanation: A DBID is 2 bytes in length (hence, 4 characters maximum).

System action: Processing is terminated.

User response: Resubmit the job providing a DBID which is no longer than 4 hexadecimal characters.

Severity: 8 (error)

DSN1122E   OBID MULTIPLY DEFINED

Explanation: The OBID keyword can be specified only once for a given job.

System action: Processing is terminated.

User response: Resubmit the job providing a single OBID value.

Severity: 8 (error)

DSN1123E   OBID SPECIFIED WITH NO VALUE

Explanation: The OBID keyword specifies a hexadecimal database object identifier. OBID must be followed by a value.

System action: Processing is terminated.

User response: Resubmit the job providing an OBID (value).

Severity: 8 (error)

DSN1124E   OBID > 4 HEX CHARACTERS

Explanation: An OBID is 2 bytes in length (hence, 4 characters maximum).

System action: Processing is terminated.

User response: Resubmit the job providing an OBID which is no longer than 4 hexadecimal characters.

Severity: 8 (error)

DSN1125E   PAGE SPECIFIED WITH NO VALUE

Explanation: The PAGE keyword specifies a hexadecimal page number. PAGE must be followed by a value.

System action: Processing is terminated.

User response: Resubmit the job providing a PAGE (value).

Severity: 8 (error)

DSN1126E   PAGE > 6 HEX CHARACTERS

Explanation: A page number is 3 bytes in length (hence, 6 characters maximum).

System action: Processing is terminated.

User response: Resubmit the job providing a PAGE value which is no longer than 6 hexadecimal characters.

DSN1127E • DSN1132E

Severity: 8 (error)

DSN1127E   KEYWORD xxxxxxxx UNKNOWN

Explanation: DSN1LOGP does not recognize the keyword 'xxxxxxxx'.

System action: Processing is terminated.

User response: Check to make sure all keywords are valid and resubmit the job.

Severity: 8 (error)

DSN1128E   *csect-name keyword* **KEYWORD SPECIFIED WITHOUT** *keyword*

Explanation: The identified keywords are associated with each other. You cannot specify one keyword without the other keyword.

System action: Processing is terminated.

User response: Resubmit the job providing the corresponding correct keywords.

Severity: 8 (error)

DSN1129E   PAGE WITHOUT DBID OBID

Explanation: DB2 pages exist in objects called 'page sets'. When you specify a page, you must also identify the page set by using the appropriate DBID and OBID specifications.

System action: Processing is terminated.

User response: Resubmit the job providing the associated DBID (value) and OBID (value) that correspond to the given page.

Severity: 8 (error)

DSN1130E   OBID WITHOUT DBID

Explanation: When you specify an OBID, you must also identify the database that contains that OBID by specifying the DBID of the database.

System action: Processing is terminated.

User response: Resubmit the job providing both an OBID (value) and a DBID (value).

Severity: 8 (error)

DSN1131E   OFFSET/VALUE PAIR NUMBER xx CONTAINS INVALID HEX DATA

Explanation: A hexadecimal specification is required. The value for offset/value pair number 'xx' contains an invalid hexadecimal character.

System action: Processing is terminated.

User response: Resubmit the job providing a correct hexadecimal value specification.

Severity: 8 (error)

DSN1132E   URID SPECIFIED WITH NO VALUE

Explanation: The URID keyword specifies the hexadecimal unit of recovery identifier. URID must be followed by a value.

System action: Processing is terminated.

User response: Resubmit the job providing an associated value for the URID parameter.

Severity: 8 (error)

DSN1133E   **URID > 12 HEX CHARACTERS**

Explanation: The maximum size of a URID in DB2 is 6 bytes (hence, 12 characters).

System action: Processing is terminated.

User response: Resubmit the job providing a valid URID value of no more than 12 hexadecimal characters.

Severity: 8 (error)

DSN1134E   *csect-name* **LIMIT OF** *number keyword***S IS EXCEEDED**

Explanation: The *keyword* was specified more than the maximum number of times allowed in any given job.

System action: Processing is terminated.

User response: Resubmit the job specifying this keyword no more than the maximum number of times.

Severity: 8 (error)

DSN1135E   **SUMMARY MULTIPLY DEFINED**

Explanation: The SUMMARY option can be specified only once.

System action: Processing is terminated.

User response: Resubmit the job using a single SUMMARY specification.

Severity: 8 (error)

DSN1136E   **SUMMARY SPECIFIED WITH NO VALUE OR INCORRECT VALUE**

Explanation: The permitted values for summary are YES, NO, ONLY, or FILTER.

System action: Processing is terminated.

User response: Resubmit the job providing a valid parameter for the SUMMARY option.

Severity: 8 (error)

DSN1137I   *csect-name* **FIRST PAGE SET CONTROL RECORD AFTER RESTART LOG** *seq nnnnnnnnnnnn*

Explanation: This message shows the log sequence number of a record that serves as an implicit indication that a restart occurred just prior to this point.

seq   RBA or LRSN

nnnnnnnnnnnn
        The log sequence number

System action: Processing continues.

Severity: 0 (informational)

DSN1138E   *csect-name keyword* **AND** *keyword* **CANNOT BOTH BE SPECIFIED**

Explanation: You cannot specify both keywords in one job.

System action: Processing is terminated.

User response: Resubmit the job specifying only one of the keywords.

Severity: 8 (error)

DSN1139E   RID SPECIFIED WITH NO VALUE

Explanation: The RID keyword specifies a 4-byte hexadecimal number, with the first 3 bytes representing the page number and the last byte representing the record ID. A value must be associated with the RID.

System action: Processing is terminated.

User response: Resubmit the job providing a RID (value).

Severity: 8 (error)

DSN1140E   RID > 8 HEX CHARACTERS

Explanation: The maximum size of a RID in DB2 is 4 bytes (hence, 8 characters).

System action: Processing is terminated.

User response: Resubmit the job providing a valid RID value of no more than 8 hexadecimal characters.

Severity: 8 (error)

DSN1141E   RID WITHOUT DBID OBID

Explanation: The DBID and OBID associated with the given RID value must be specified.

System action: Processing is terminated.

User response: Resubmit the job providing the associated DBID (value) and OBID (value) that correspond to the given RID value.

Severity: 8 (error)

DSN1142E   DATAONLY MULTIPLY DEFINED

Explanation: The DATAONLY option limits the log records extracted in the detail report to those that represent data modifications. This option can be specified only once.

System action: Processing is terminated.

User response: Resubmit the job specifying DATAONLY once.

Severity: 8 (error)

DSN1143E   DATAONLY SPECIFIED WITH NO VALUE OR INCORRECT VALUE

Explanation: The DATAONLY keyword must be specified with either a YES or NO value. Any other value (including no value) is incorrect.

System action: Processing is terminated.

User response: Resubmit the job specifying the correct value for the DATAONLY parameter.

Severity: 8 (error)

DSN1144E   RBASTART SPECIFIED WITH NO VALUE

Explanation: The RBASTART keyword specifies the hexadecimal log RBA from which to begin extraction. RBASTART must be followed by a value.

System action: Processing is terminated.

User response: Resubmit the job providing an associated value for the RBASTART parameter.

Severity: 8 (error)

**DSN1146E   CURRENT END CHECKPOINT TIMESTAMP TIMESTAMP=yy.ddd hh.mm.ss LOG RBA=xxxxxx IS
OUT OF SEQUENCE. THE GREATEST TIMESTAMP SEEN SO FAR IS TIMESTAMP=yy.ddd
hh.mm.ss LOG RBA=xxxxxx Multiple processors may be writing to the same log.**

Explanation: This message indicates that the current log record has a timestamp that is less than the previous
timestamp processed. This may be a potential problem.

System action: Processing continues.

User response: Examine current log to determine whether multiple processors are writing to the same log. (Data
may be being overwritten.) This may lead to data inconsistencies.

Severity: 4 (warning)

DSN1147E   SYSCOPY MULTIPLY DEFINED

Explanation: The SYSCOPY option can only be specified once.

System action: Processing is terminated.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For
information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis
Guide and Reference.

User response: Resubmit the job using a single SYSCOPY specification.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
  time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 8

DSN1148E   SYSCOPY SPECIFIED WITH NO VALUE OR INCORRECT VALUE

Explanation: The SYSCOPY keyword must be specified with either a YES or NO value. Any other value (including
no value) is incorrect.

System action: Processing is terminated.

System programmer response: If you suspect an error in DB2, you might need to report the problem. For
information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis
Guide and Reference.

User response: Resubmit the job specifying the correct value for the SYSCOPY parameter.

Problem determination: Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
  time spanning the failure.
- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

Severity: 8

DSN1150I   SUMMARY OF COMPLETED EVENTS

Explanation: This message heads the summary of completed units of recovery (URs) and checkpoints. If you
suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the
problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

System action: Processing continues.

Severity: 0 (informational)

DSN1151I

DSN1151I	*csect-name mbn* **UR CONNID=***cc* **CORRID=***bb* **AUTHID=***aa* **PLAN=***pp* **START DATE=***yy.ddd* **TIME=***hh:mm:ss* **DISP=***xx* **INFO=***ii* **STARTRBA=***ss* **ENDRBA=***ee* **STARTLRSN=***rr* **ENDLRSN=***nn* **NID=***nid* **LUWID=***luwid* **COORDINATOR=***name* **PARTICIPANTS: (***list of participants***)**

Explanation: This message describes a unit of recovery (UR) that terminated for data sharing member *mbn*.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

cc   Connection ID (for example, BATCH).

bb   Correlation ID (for example, JOB xxx).

aa   Authorization ID executing the UR.

pp   Plan name used by the UR.

          DSN1LOGP returns a plan name of SYSTEM in this field if the UR was generated as a result of a system event (such as the first update of a table space). In such a case, no bound plan exists for the UR described.

yy.ddd   Starting date of the UR.

hh:mm:ss
          Starting time of the UR.

xx   Disposition of the UR. Values reported are:
          • Inflight
          • In-abort
          • In-commit
          • Indoubt
          • Committed
          • Aborted
          • P-Abort

          If the UR was bypassed by a conditional restart, DSN1LOGP indicates this by preceding the disposition value with the word BYPASSED.

ii   The status of the data is one of the following:
          • COMPLETE, indicating that all page sets modified by this UR were identified.
          • PARTIAL, indicating that the list of page sets modified by this UR is incomplete. DSN1LOGP indicates PARTIAL status if all records associated with a UR are not available, and no checkpoint is found before the UR's completion.

ss   RBA of the first log record associated with the UR (that is, the URID).

ee   RBA of the last log record associated with the UR.

rr   LRSN of the first log record associated with the UR.

nn   LRSN of the last log record associated with the UR

nid   The identifier assigned by IMS or CICS for this UR. This is blank if the UR is a result of TSO, batch, or remote access.

luwid   The logical unit of work identifier associated with the UR

name   The name of the remote coordinator associated with the UR. This is blank if the coordinator is the local DB2, IMS, or CICS.

list of participants
          The names of all remote participants that might have updated data in this UR.

This message is followed by messages that identify the page sets modified by the UR. For each page set, a single message is printed in the following form:

```
DATABASE=dbid=dbname
PAGE SET=obid=psname
```

Where:

dbid   Internal database identifier

dbname Database name corresponding to the *dbid*

obid   Internal page set identifier

psname Page set name corresponding to the OBID

The *dbname* and *psname* fields might be absent if a full checkpoint was unavailable to DSN1LOGP or was caused by overlapped processing of DB2 during a checkpoint.

If no data was modified by the UR (for example, certain URs might be created merely to track information that is valid only for the life of the application), the phrase NO DATA MODIFIED is printed.

System action: Processing continues.

Severity: 0 (informational)

DSN1152I   *csect-name mbn* **UR CONNID=** ** **CORRID=** ** **AUTHID=** ** **PLAN=** ** **START DATE=** ** **TIME=** **
DISP=*xx* **INFO=***ii* **STARTRBA=***ss* **STARTLRSN=** ** **ENDRBA=***ee* **ENDLRSN=***nn***NID=***nid*
LUWID=*luwid* **COORDINATOR=***name* **PARTICIPANTS: (***list of participants***)**

Explanation: This message describes a unit of recovery (UR) for data sharing member *mbn*. The UR identifying information (**) described in message DSN1151I is unavailable.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

The remaining information (DISP, INFO, START, END, NID, LUWID, COORDINATOR and PARTICIPANTS) appears just as it does in message DSN1151I. If the UR did not complete, the END RBA or LRSN values are not printed.

This message is followed by messages that identify the page sets modified by the UR. Details of that are in the explanation of message DSN1151I.

System action: Processing continues.

Severity: 0 (informational)

DSN1153I   *csect-name* **CHECKPOINT** *mbn* **STARTRBA=***ss* **ENDRBA=***ee* **STARTLRSN=***rr* **ENDLRSN=***nn*
DATE=*yy.ddd* **TIME=***hh:mm:ss*

Explanation: This message describes a complete checkpoint on the log starting and ending RBA or LRSN for data sharing member *mbn*. If the information is available, DSN1LOGP also returns the date and time that the checkpoint started.

When this message follows message DSN1157I RESTART SUMMARY, it identifies the checkpoint that would be used at restart. If no checkpoint is available, message DSN1158I is printed instead.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

System action: Processing continues.

Severity: 0 (informational)

DSN1154I   *csect-name* **RESTART** *mbn* **AT RBA=***xx* **LRSN=***ll* **DATE=***yy.ddd* **TIME=***hh:mm:ss*

Explanation: A normal restart occurred at log RBA *xx* or LRSN *ll*.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

DSN1LOGP also returns the date and time of that restart.

System action: Processing continues.

Severity: 0 (informational)

DSN1155I   *csect-name* **CONDITIONAL RESTART** *mbn* **AT RBA=***xx* **LRSN=***ll* **DATE=***yy.ddd* **TIME=***hh:mm:ss*
               STARTRBA=*ss* **ENDRBA=***ee* **FORWARD=***ff* **BACKOUT=***bb*

Explanation: A conditional restart occurred at the log RBA *xx* and LRSN *ll*. DSN1LOGP also returns the date and time of that restart.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

ss   Relative byte address of the beginning of the restart

ee   Relative byte address of the conclusion of the restart

ff   Whether forward log recovery phase was to be bypassed (YES | NO)

bb   Whether backward log recovery phase was to be bypassed (YES | NO)

System action: Processing continues.

Severity: 0 (informational)

DSN1156I   **ALL URS COMPLETE**

Explanation: There are no URs outstanding for restart.

System action: Processing continues.

Severity: 0 (informational)

DSN1157I   **RESTART SUMMARY**

Explanation: This message heads the summary of the description of work to be performed at restart. Restart information that follows is based on the scope of the log scanned. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

System action: Processing continues.

Severity: 0 (informational)

DSN1158I   **NO CHECKPOINT AVAILABLE — RESTART SUMMARY INCOMPLETE**

Explanation: No checkpoint is available within the scope of the log scanned. The information following this message includes:
- URs that have not completed
- Page sets modified by these URs
- Page sets with writes pending.

The information cannot be considered complete.

System action: Processing continues.

Severity: 0 (informational)

DSN1159I   **NO DATABASE WRITES PENDING**

Explanation: All database I/O has been completed. If the Log Extractor can find no complete checkpoint, it will return an additional message '(BASED ON INCOMPLETE LOG INFORMATION)', to warn you that there may in fact be pending database writes.

System action: Processing continues.

Severity: 0 (informational)

DSN1160I   **DATABASE WRITES PENDING:**

Explanation: Database writes are pending at restart. Each page set with outstanding I/O is identified by a separate message following this one:

DATABASE=dbid=dbname
PAGESET=obid=psname START=ss

Where:

dbid   Internal database identifier

dbname
        Database name corresponding to the dbid

obid   Internal page set identifier

psname
        Page set name corresponding to the obid

        Note: *dbname* and *psname* may be absent if a full checkpoint was unavailable to the Log Extractor or was caused by overlapped processing of DB2 during a checkpoint.

ss   Log RBA of the earliest log record required to complete outstanding writes for this page set. If the value of ss is returned as **, this information is unavailable. Such will be the case if the Log Extractor is unable to locate a full checkpoint. In this event, DSN1LOGP will return an additional message (LIST MAY BE INCOMPLETE), to alert you to the possibility that the list of page sets with outstanding writes may be incomplete.

System action: Processing continues.

Severity: 0 (informational)

DSN1162I   *csect-name mbn***UR CONNID=***cc* **CORRID=***bb* **AUTHID=***aa* **PLAN=***pp* **START DATE=***yy.ddd*
              TIME=*hh:mm:ss* **DISP=***xx* **INFO=***ii* **STARTRBA=***ss* **STARTLRSN=***ll* **NID=***nid***LUWID=***luwid*
              COORDINATOR=*name* **PARTICIPANTS: (***list of participants***)**

Explanation: This message describes a unit of recovery (UR) that has not terminated. The information returned is the same as for message DSN1151I, except that the END field is not included in the message because the UR did not terminate.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

System action: Processing continues.

Severity: 0 (informational)

DSN1163E   **NO CHECKPOINT AVAILABLE DUE TO LOG ERROR -- RESTART SUMMARY INCOMPLETE**

Explanation: A log error was encountered. DSN1LOGP marked any checkpoints encountered before the log error as invalid. There were no complete checkpoints following the log error in the specified log range. The information following this message includes:

- URs that have not completed
- Page set modified by these URs
- Page sets with writes pending.

This information cannot be considered complete.

System action: Processing continues.

Severity: 0 (informational)

DSN1164E   INVALID xxxx FOUND AT RBA = yyyyyyyyyyyy

Explanation: While processing the SUMMARY option, an invalid checkpoint record (either URE or RURE identified by 'xxxx') was encountered in the log.

System action: Processing continues.

User response: If the checkpoint record identified in the message is used to restart DB2, the restart will fail because DB2 will not be able to process the unit of recovery presented by the invalid URE or RURE. DB2 must be restarted from a previous checkpoint to correctly process all units of recovery. The print log map (DSNJU004) utility can be used to list the checkpoint records in the BSDS, and the CHKPTRBA parameter of the change log inventory (DSNJU003) utility can be used to direct the DB2 restart to use a particular checkpoint.

Severity: 0 (informational)

DSN1170E   SUBTYPE MULTIPLY DEFINED

Explanation: The SUBTYPE parameter can be specified only once for any given job.

System action: Processing terminates.

User response: Provide a single SUBTYPE value and resubmit the job.

Severity: 12 (severe error)

DSN1171E   SUBTYPE SPECIFIED WITH NO VALUE

Explanation: The SUBTYPE parameter specifies a decimal set subtype. The SUBTYPE must be followed by a value, i.e. SUBTYPE(value).

System action: Processing terminates.

User response: Provide a SUBTYPE value and resubmit the job.

Severity: 12 (severe error)

DSN1172E   SUBTYPE > 2 DECIMAL CHARACTERS

Explanation: The maximum size of a SUBTYPE is 2 decimal characters.

System action: Processing terminates.

User response: Provide a SUBTYPE value with a maximum of 2 decimal characters and resubmit the job.

Severity: 12 (severe error)

DSN1173E   *csect-name* INVALID VALUE *invalid-value* SPECIFIED FOR KEYWORD *keyword-name*

Explanation: The specified keyword value is not valid.

csect-name
> The name of the control section that issued the message.

invalid-value
> The invalid value.

keyword-name
> The keyword that contained an invalid value.

System action: Processing terminates.

User response: Provide a valid value and resubmit the job.

Severity: 12 (severe error)

Related concepts:

DB2 online utilities (DB2 Utilities)

DB2 stand-alone utilities (DB2 Utilities)

DSN1174E   OFFSET SPECIFIED WITH NO SUBTYPE

Explanation: The given OFFSET has no corresponding SUBTYPE.

System action: Processing terminates.

User response: Provide a valid SUBTYPE for the given OFFSET and resubmit the job.

Severity: 12 (severe error)

DSN1175E   OFFSET SPECIFIED WITH NO VALUE OR INCORRECT VALUE

Explanation: The given OFFSET has no corresponding VALUE given or an incorrect VALUE is specified, such as specifying another keyword in place of a value.

System action: Processing terminates.

User response: Provide a valid VALUE for the given OFFSET and resubmit the job.

Severity: 12 (severe error)

DSN1176E   OFFSET > 8 HEX CHARACTERS

Explanation: The maximum size for an OFFSET is 4 bytes (hence, 8 characters).

System action: Processing terminates.

User response: Provide an OFFSET with a maximum of 8 characters and resubmit the job.

Severity: 12 (severe error)

DSN1177E   LIMIT OF 10 OFFSET, VALUE PAIRS EXCEEDED

Explanation: A maximum of 10 OFFSET, VALUE pairs can be specified for a given log record SUBTYPE.

System action: Processing terminates.

User response: Provide a maximum of 10 OFFSET, VALUE pairs for a given SUBTYPE and resubmit the job.

Severity: 12 (severe error)

DSN1178E   OFFSET SPECIFIED WITH NO CORRESPONDING VALUE

Explanation: The given OFFSET has no corresponding VALUE keyword specified.

System action: Processing terminates.

User response: Provide an associated VALUE (value) for the given OFFSET and resubmit the job.

Severity: 12 (severe error)

DSN1179E   VALUE SPECIFIED WITH NO CORRESPONDING OFFSET OR SUBTYPE PARAMETER

Explanation: The given VALUE has either no corresponding OFFSET or SUBTYPE parameter specified.

System action: Processing terminates.

User response: Provide an associated OFFSET or SUBTYPE for the given VALUE and resubmit the job.

Severity: 12 (severe error)

DSN1180E   VALUE SPECIFIED WITH NO VALUE OR INCORRECT VALUE

Explanation: The given VALUE has no corresponding value given, an incorrect value is specified, such as specifying a keyword in place of a value, or an uneven number of characters were specified.

System action: Processing terminates.

User response: Provide a valid value for the given VALUE and resubmit the job.

Severity: 12 (severe error)

DSN1181E   VALUE SPECIFIED WITH NO CORRESPONDING OFFSET

Explanation: The given VALUE has no corresponding OFFSET specified.

System action: Processing terminates.

User response: Provide a valid OFFSET for the given VALUE and resubmit the job.

Severity: 12 (severe error)

DSN1182E   LIMIT OF 10 LUWIDs EXCEEDED

Explanation: A maximum of 10 LUWID keywords can be specified in any given job.

System action: DSN1LOGP processing is terminated.

User response: Resubmit the job providing no more than 10 LUWID keywords.

Severity: 8 (error)

DSN1183E   LUWID SPECIFIED WITH INCORRECT VALUE OR NO VALUE

Explanation: The LUWID consists of an LU network name, an LUW instance number, and a commit sequence number. The LU network name consists of a 1 to 8 character network ID, a period, and a 1 to 8 character network LU name. The LUW instance number consists of a period followed by 12 hexadecimal characters. The LUW sequence number consists of a period followed by 4 hexadecimal characters.

System action: DSN1LOGP processing is terminated.

User response: Resubmit the job providing a valid LUWID value.

Severity: 8 (error)

DSN1184E   FILTER KEYWORD MULTIPLY DEFINED

Explanation: The FILTER keyword was specified more than once.

System action: DSN1LOGP processing is terminated.

User response: Correct the input to provide only one FILTER keyword and resubmit the job.

Severity: 8 (error)

DSN1185E   FILTER KEYWORD SPECIFIED WITHOUT SUMMARY KEYWORD

Explanation: The FILTER keyword was specified without an associated SUMMARY keyword.

System action: DSN1LOGP processing is terminated.

User response: Correct the input to include the SUMMARY keyword and resubmit the job.

Severity: 8 (error)

| **DSN1186E**   *csect-name* **KEYWORD** *keyword* **MUST EITHER INPUT 12 OR 20 CHARACTERS**

| **Explanation:** LRSN values with the specified *keyword* must be a value of 12 characters or 20 characters in length.

| **System action:** Processing terminates.

| **User response:** Correct the input on the control statement for the specified keyword and resubmit the job.

| **Severity:** 8 (error)

| **DSN1187E**   *csect-name* **VALUES FOR KEYWORDS** *keyword-1* **AND** *keyword-2* **MUST BE THE SAME LENGTH**

| **Explanation:** The DSN1LOGP stand-alone utility issues this message when a pair of keywords do not have the same
| input length. LRSN values for the specified keyword must be of the same format. The input values must either
| specify a 6-byte log format (12 characters) or a 10-byte log format (20 characters).

| **System action:** Processing terminates.

| **User response:** Correct the input on the control statement for the specified keyword and resubmit the job.

| **Severity:** 8 (error)

DSN1196I   *database spacenam dbid obid #pg-regressions*

Explanation: This message contains the following summary information about page regression errors:

database
     The name of the database that has page regression. An asterisk would be printed if the name is unknown.

spacenam
     The name of the pageset that has page regression. An asterisk would be printed if the pageset name is
     unknown.

dbid   Hexadecimal DBID of the pageset that has page regression.

obid   Hexadecimal OBID of the database that has page regression.

 # pg-regressions
     The total number of regressed pages for the pageset.

DSN1197I   **TOTAL NUMBER OF PAGES CHECKED FOR REGRESSION =** *page-count*

Explanation: This message lists the total number of unique pages that DSN1LOGP has checked for page regression
errors. *page-count* is displayed in decimal format.

DSN1198I   ********NO REGRESSED PAGES DETECTED********

Explanation: No page regression errors were detected within the RBA/LRSN range that was specified.

DSN1210E   **DSNJSLR ERROR RETCODE=***reason-code* **REASON CODE =***return-code*

Explanation: The stand-alone log read function detected an error.

reason-code
     A numeric value that indicates the reason for the failure.

return-code
     A numeric value that indicates the reason for the failure.

System action: Processing is terminated.

Severity: 8 (error)

Related concepts:

▶ DB2 reason codes (DB2 Codes)

Related tasks:

▶ Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

▶ Registers and return codes (DB2 Administration Guide)

DSN1211E   DSNJSLR RETURNED BAD LOG RBA

Explanation: One of the three problems listed below exists.

1. The recovery log data set is damaged.
2. You identified a data set that isn't a recovery log data set.
3. The stand-alone log read service has a problem.

System action: Processing continues.

User response: Determine if the problem is your error by dumping the data set and determining if it is a log data set.

Problem determination: A common error is to specify the first data set on an archive tape (the 'Bxxxxxxx' data set) as a log data set; it is actually a BSDS.

Severity: 0 (informational)

DSN1212I   *csect-name* FIRST LOG *seq* ENCOUNTERED *nnnnnnnnnnnn*

Explanation: This message identifies the sequence number of the first log record that was read.

seq   RBA or LRSN

System action: Processing continues.

Severity: 0 (informational)

DSN1213I   *csect-name* LAST LOG *seq* ENCOUNTERED *nnnnnnnnnnnn*

Explanation: This message identifies the sequence number of the last log record that was read.

seq   RBA or LRSN

System action: Processing continues.

Severity: 0 (informational)

DSN1214I   NUMBER OF LOG RECORDS READ nnnnnnnnnnnnnnnnn

Explanation: This identifies the number (in decimal) of 'logical' log records read during DSN1LOGP processing.

System action: Processing continues.

Severity: 0 (informational)

DSN1215I   NO LOG RECORDS WERE READ

Explanation: DSN1LOGP read no log records.

System action: Processing continues. Possible explanations:

- An error prevents DSN1LOGP from continuing, so no log records have yet been processed. If this is the case, an error message should precede this message.
- You specified the active log data sets or archive log data sets out of RBA sequence.
- You specified a STARTRBA that is greater than any RBA in the active and/or archive data sets available.

Severity: 0 (informational)

DSN1216E *csect-name* **DSNJSLR ERROR RETCODE =** *xxxxxxxx* **REASON CODE =** *yyyyyyyy seq* = *nnnnnnnnnnnn*

Explanation: The stand-alone log read function returned an error indicating that either the log has an error in one of the control intervals (CI) or a data set containing the requested sequence number cannot be located. The sequence number shown in the message indicates where the error was detected.

seq   RBA or LRSN

The sequence number points:
- To the start of the CI, if there is a problem with the log control interval definition (LCID), or with any of the general control information within a CI.
- To the log record in the CI, if there is a problem with a log record header (LRH).

If this is the first log record read during this execution of DSN1LOGP, and if there is a problem with the LCID, the sequence number is all zeros.

Before returning any records, DSN1LOGP checks the control information (LCID) at the end of a CI, and analyzes the LRH to ensure that all records are properly chained together within the CI. If an error is detected while performing this process, DSN1LOGP issues this message before dumping the entire CI. It will not format individual records within the CI, but will, if possible, continue processing by reading the next CI.

System action: Processing continues.

Severity: 0 (informational)

Related concepts:

DB2 reason codes (DB2 Codes)

Related tasks:

Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

Registers and return codes (DB2 Administration Guide)

DSN1LOGP (DB2 Utilities)

DSN1217E *csect-name mbn* **DSNJSLR WARNING RETCODE =** *xxxxxxxx* **REASON CODE =** *yyyyyyyy* **CURRENT LOG** *seq* = *nnnnnnnnnnnn*

Explanation: A gap in the log sequence range was encountered for member *mba*. The CURRENT LOG sequence number shows the log record for the member following the gap, and is formatted following this message.

Previous log record can be found with same member ID.

mbn   MEMBER=*membername*, where *membername* is the member of a data sharing group. This is printed only for a data sharing record.

seq   RBA or LRSN

System action: Processing continues.

Severity: 0 (informational)

Related concepts:

DB2 reason codes (DB2 Codes)

Related tasks:

Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

Registers and return codes (DB2 Administration Guide)

DSN1218I   NUMBER OF LOG ERROR MESSAGES nnnnnnnnnnnnnnnn

Explanation: DSN1LOGP distinguishes three classes of errors:

1. Code problems existing in the DB2 or system code used for DSN1LOGP. In the event of a SYSABEND dump, a user abend code of U0153 will be issued, and DSN1LOGP will be terminated.

2. Incorrect invocation of DSN1LOGP caused, perhaps, by your having used an incorrect keyword or missed a DD statement. Under these circumstances, DSN1LOGP will issue appropriate error messages, and the program will be terminated.

3. Failure by a given log CI under the scrutiny of DSN1LOGP. Such scrutiny is performed before any of the records within the CI are processed. Failure is an indication of logical damage, and error messages are issued by the Log Extractor. The CI or failing log record is printed, and DSN1LOGP continues to the next CI or log record.

   The count 'nnnnnnnnnnnnnnnn' provided summarizes the number (in decimal) of errors DSN1LOGP detected while accessing the log.

System action: Processing continues.

Severity: 0 (informational)

DSN1219E   SYSSUMRY DD STATEMENT MISSING

Explanation: You requested the SUMMARY option, but failed to include the SYSSUMRY DD statement in your JCL invocation.

System action: Processing terminates.

User response: Resubmit the job with a SYSSUMRY DD statement included in the JCL invocation.

Severity: 8 (error)

DSN1220E   ARCHIVE LOG TRUNCATED AT xxxxxxxx - INVALID LOG RECORDS WERE READ

Explanation: At a restart of DB2, an archive log was truncated. This archive log data set could not be physically altered to reflect this truncation, and invalid log records therefore still exist. DSN1LOGP has already reported this information in the summary report, and cannot retract it. Nor can it disregard the invalid log information already read in order adequately to summarize what has occurred. Therefore, all information up to this point in the log will be summarized, and a new summary report initiated. Consequently, the same UR may be reported twice with different dispositions and different page sets modified. To avoid this condition, use the BSDS DD statement instead of the ARCHIVE DD statement.

System action: Processing continues.

Severity: 0 (informational)

DSN1221E   DSNJSLR ERROR RETCODE=*return-code*REASON CODE=*reason-code* VSAM RETURN
               CODE=*vsam-return-code* ERROR CODE=*error-code*

Explanation: The stand-alone log read function resulted in a VSAM error.

The *vsam-return-code* and *error-code* values identify the nature of the VSAM error. Refer to the appropriate VSAM publication for an explanation of these codes.

System action: Processing continues.

Severity: 0 (informational)

Related tasks:

▶ Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

▶ Registers and return codes (DB2 Administration Guide)

DSN1222E   **DSNJSLR ERROR RETCODE=***return-code* **REASON CODE=***reason-code* **DYNAMIC ALLOCATION INFORMATION CODE=***dynamic-allocation-code* **ERROR CODE=***error-code*

Explanation: An error occurred while dynamically allocating a log data set. The *dynamic-allocation-code* and e*error-code* values were returned by the dynamic allocation SVC and identify the nature of the error. Refer to the appropriate MVS publication for an explanation of these codes.

System action: Processing terminates.

Severity: 12 (severe error)

Related tasks:

➡ Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

➡ Registers and return codes (DB2 Administration Guide)

DSN1223E   **DSNJSLR ERROR RETCODE=***return-code***REASON CODE=***reason-code* **RDJFCB RETURN CODE=***rdjfcb-return-code*

Explanation: An error occurred while trying to read the job file control block.

The RDJFCB return code (*rdjfcb-return-code*) identifies the nature of the error. Refer to the appropriate MVS publication for an explanation of these codes.

System action: Processing continues.

Severity: 0 (informational)

Related tasks:

➡ Reading log records with OPEN, GET, and CLOSE (DB2 Administration Guide)

Related reference:

➡ Registers and return codes (DB2 Administration Guide)

DSN1224I   *csect-name* **SPECIFIED LOG type** *nnnnnnnnnnnn* **COULD NOT BE LOCATED FOR MEMBER** *mbn*

Explanation: The specified start or end log record address could not be found in the available log data sets. This error might indicate that a specific log record could not be formatted.

Type   RBASTART/RBAEND/LRSNSTART/LRSNEND

nnnnnnnnnnnn
          Specified start or end log RBA/LRSN

mbn   Member name. For non-data sharing, this value is blank

System action: Processing continues.

User response: Check that the specified start log record address (RBA or LRSN) is required for the determination of the problem. Either correct the log range to the specification that you want, or provide the missing log data sets either in the ACTIVE*n* or ARCHIVE DD statements.

A common problem is to specify an excessively large range to express that all available log records should be formatted. If all available log records should be formatted, simply omit the low and high limits to suppress this message.

Problem determination: Contact IBM Software Support for assistance.

Severity: 4 (warning)

DSN1300E   *csect-name* **RBA RANGE CANNOT BE USED TO PRINT LOG RECORDS FROM MULTIPLE DB2 SYSTEMS**

Explanation: RBASTART and RBAEND cannot be used to define the range when selecting log records from multiple DB2 systems. LRSNSTART and LRSNEND should be used instead.

System action: Processing terminates.

User response: Resubmit the job with either RBA values only or LRSN values only.

DSN1303E   *csect-name* **INCOMPATIBLE DD STATEMENTS WERE SPECIFIED FOR LOG PROCESSING**

Explanation: Either MxxACTn or MxxARCHV was erroneously specified in conjunction with MxxBSDS to locate the log data sets of a member of a DB2 data sharing group for a DSN1LOGP job.

System action: Processing terminates.

User response: Correct the DD statement and resubmit the job.

DSN1309E   *csect-name* **SPECIFIED END RANGE IS LESS THAN START RANGE**

Explanation: The value specified for ENDLRSN or ENDRBA is less than the value of STARTLRSN or STARTRBA for a DSN1LOGP job.

System action: Processing terminates.

User response: Resubmit the job with correct start and end values.

DSN1401I   **DSNTSMFD ERROR: UNABLE TO ALLOCATE STORAGE FOR IO AREA**

Explanation: DSNTSMFD was unable to allocate dynamic storage required for processing input and output.

System action: DSNTSMFD terminates.

Programmer response: Increase the region size of the job or job step that executes DSNTSMFD. If the problem occurs during peak system load, you might need to run DSNTSMFD at a different time.

Severity: 12 (severe error)

Related reference:

⇨ DSNTSMFD (DB2 Utilities)

DSN1402I   **DSNTSMFD ERROR:** *action-type* **ERROR FOR DD** *dd-name*

Explanation: DSNTSMFD was unable to allocate dynamic storage required for processing input and output.

action-type
> The action that was attempted: OPEN, WRITE, or CLOSE. Typical causes for this condition include, but are not limited to:

> **OPEN**

>> • Data set has invalid attributes (such as record format or logical record length)
>> • Data set is protected or otherwise unavailable
>> • Job step does not contain a DD statement for the file that is identified in the message

> **WRITE**

>> • Data set has invalid attributes (such as record format or logical record length)
>> • Data set is protected or otherwise unavailable

dd-name
> Identifier of the file that was being opened or closed.

System action: DSNTSMFD terminates.

Programmer response: Add a DD statement for *dd-name* if one does not already exist in the job step that executes DSNTSMFD. Otherwise, verify the attributes of the data set allocated to that DD. Ensure that the data set is not in use by another agent and that it is available for write access by the user ID used to execute DSNTSMFD. Check the system log for evidence of system or device errors.

Severity: 12 (severe error)

Related reference:

➲ DSNTSMFD (DB2 Utilities)

DSN1409I   **DSNTSMFD ERROR: A SEVERE ERROR WAS ENCOUNTERED; UNABLE TO CONTINUE**

Explanation: DSNTSMFD encountered an unrecoverable error, as noted by one or more preceding messages.

System action: DSNTSMFD terminates.

User response: Look for other messages that precede this one, and take the indicated actions.

Severity: 12 (severe error)

Related reference:

➲ DSNTSMFD (DB2 Utilities)

DSN1410I   **DSNTSMFD ERROR: UNABLE TO ALLOCATE MEMORY**

Explanation: The z/OS compression service CSRCESRV is not available.

System action: DSNTSMFD continues execution, but does not decompress any compressed records. Data remains in the compressed form.

User response: No action is required.

To decompress the data, run DSNTSMFD on a system with compression services available.

Severity: 4 (warning)

DSN1411I   **DSNTSMFD ERROR: UNABLE TO DECOMPRESS, RC=***return-code*

Explanation: DSNTSMFD encountered an error while decompressing a record.

return-code
>   A numeric value that indicates the reason for the failure. The return code is issued by the compression service.

System action: The record is not decompressed. Individual instances of message DSN1411I are issued for each record that cannot be decompressed.

User response: Use the return code provided by the z/OS compression service CSRCESRV to resolve the decompression error.

Severity: 8 (error)

Related reference:

➲ CSRCESRV - Compress and expand data (z/OS MVS Programming: Assembler Services Reference, Volume 1)

DSN1700E   **VALUE FOR KEYWORD xxxxxxxx IS INVALID**

Explanation: The SDMPIN input contains the 'xxxxxxxx'. The value specified for that keyword is invalid.

System action: Processing is terminated.

User response: Provide a correct keyword value and resubmit the job.

Severity: 8 (error)

DSN1701E   **START TRACE MISSING**

Explanation: The START TRACE command is required. The first keyword record of the SDMPIN input does not contain a START TRACE command.

System action: Processing is terminated.

User response: Resubmit the job with START TRACE as the first keyword record in the SDMPIN input.

Severity: 8 (error)

DSN1702E   KEYWORD xxxxxxxx MULTIPLY DEFINED

Explanation: Keyword 'xxxxxxxx' was specified multiple times in the SDMPIN input. All keywords are allowed only once.

System action: Processing is terminated.

User response: Specify the keyword only once. Resubmit the job.

Severity: 8 (error)

DSN1703E   KEYWORD xxxxxxxx UNKNOWN

Explanation: DSN1SDMP does not recognize the keyword 'xxxxxxxx'.

System action: Processing is terminated.

User response: Make sure that all keywords are valid and then resubmit the job.

Severity: 8 (error)

DSN1704E   UNABLE TO OBTAIN RETURN-AREA STORAGE

Explanation: An error resulted when DSN1SDMP attempted to obtain storage for an instrumentation facility interface (IFI) return area.

System action: Processing is terminated.

User response: When you resubmit the job either:
- Specify a larger region size in the job control language.
- Reduce the value of BUFSIZE on START TRACE.

Severity: 8 (error)

DSN1705E   SYNTAX ERROR, LINE = ll, COLUMN = cc

Explanation: A syntax error was encountered in the SDMPIN input. The location of the error is given by:

ll   The line number within the input stream that is in error.

cc   The column number.

System action: Processing is terminated.

User response: Correct the syntax error and then resubmit the error.

Severity: 8 (error)

DSN1706E   INVALID ABEND CODE SPECIFIED

Explanation: An invalid abend code was specified on the ACTION keyword.

System action: Processing is terminated.

User response: Allowable abend codes are 00E60100 through 00E60199. Specify an allowable abend code and then resubmit the job.

Severity: 8 (error)

DSN1707E   LIMIT OF 50 STATEMENTS EXCEEDED

Explanation: DSN1SDMP allows a maximum of 50 statements for a given job.

System action: Processing is terminated.

User response: Use no more than 50 statements when you resubmit the job.

Severity: 8 (error)

DSN1708E   TOO MANY SELECT CRITERIA VALUES

Explanation: DSN1SDMP allows a maximum of eight SELECT criteria values.

System action: Processing is terminated.

User response: Use no more than eight SELECT criteria values and then resubmit the job.

Severity: 8 (error)

DSN1709E   INPUT/OUTPUT ERROR, DDNAME=xxxxxxxx

Explanation: An error resulted when DSN1SDMP attempted to do I/O. The ddname of the data set to which I/O failed is 'xxxxxxxx'

System action: Processing is terminated.

User response: Ensure that the 'xxxxxxxx' DD statement in the job control language is valid and resubmit the job.

Severity: 8 (error)

DSN1710E   UNABLE TO OPEN SDMPTRAC

Explanation: An error resulted when DSN1SDMP attempted to open the SDMPTRAC data set.

System action: Processing is terminated.

User response: Ensure that the SDMPTRAC DD statement in the job control language is valid and resubmit the job.

Severity: 8 (error)

DSN1711E   START TRACE ERROR, REASON=xxxxxxxx

Explanation: DB2 did not process the submitted START TRACE command. The reason code 'xxxxxxxx' is returned from the instrumentation facility interface (IFI).

System action: Processing is terminated.

User response: Check the return area that follows message DSN1712I. Possible problems are:
- The syntax of the START TRACE command is invalid. Correct the command and resubmit the job. Do not include the subsystem recognition character before the command text.
- The user does not have authority to issue trace commands. Obtain trace authority and resubmit the job.

Severity: 8 (error)

DSN1712I   START TRACE RETURN-AREA FOLLOWS:

Explanation: DSN1SDMP issued the START TRACE command and the return area resulting from the instrumentation facility interface (IFI) COMMAND call follows this message. The return area contains DB2 command response messages.

System action: Processing continues.

Severity: 0 (informational)

DSN1713I   READA ERROR, REASON=xxxxxxxx

Explanation: An error resulted when DSN1SDMP issued a instrumentation facility interface (IFI) READA call in an attempt to obtain DB2 trace records. Reason code 'xxxxxxxx' is the IFI reason code returned in IFCARC2.

System action: DSN1SDMP writes any records that were returned and processing continues.

User response: Possible reason codes are:

00E60802
> The DSN1SDMP return area was not large enough to hold the returned trace records. This condition should not occur because the DSN1SDMP return area is the same size as the requested OPn monitor buffer.

00E60803
> No trace data was available when the READA request was made. This condition occurs when the trace started by DSN1SDMP is stopped from the operator console or by another application program.

Severity: 0 (informational)

DSN1714I   TRACE RECORDS WERE LOST

Explanation: DB2 trace records were lost and are not contained in the SDMPTRAC data set.

System action: Processing continues.

User response: Possible reasons for lost trace records are:
- The OP buffer is not large enough. Specify a larger BUFSIZE on the START TRACE command and then resubmit the job.
- The trace data is too voluminous and buffer is overflowing before the data can be read. Specify a more granular selectivity on the START TRACE command and then resubmit the job.

Severity: 0 (informational)

DSN1715E   UNABLE TO OBTAIN WORK SPACE

Explanation: An error resulted when DSN1SDMP attempted to obtain work area storage.

System action: Processing is terminated.

User response: Specify a larger region size in the job control language and then resubmit the job.

Severity: 8 (error)

DSN1716E   UNABLE TO OPEN SDMPIN

Explanation: An error resulted when DSN1SDMP attempted to open the SDMPIN data set.

System action: Processing is terminated.

User response: Ensure that the SDMPIN DD statement in the job control language is valid and resubmit the job.

Severity: 8 (error)

DSN1717I   STOP TRACE RETURN-AREA FOLLOWS:

Explanation: DSN1SDMP issued the STOP TRACE command and the return area resulting from the instrumentation facility interface (IFI) COMMAND call follows this message. The return area contains DB2 command response messages.

System action: Processing continues.

Severity: 0 (informational)

DSN1718E   STOP TRACE ERROR, REASON=xxxxxxxx

Explanation: DSN1SDMP attempted to submit a DB2 STOP TRACE command at the end of its processing; however, the command failed. The reason code returned from the instrumentation facility interface (IFI) is 'xxxxxxxx'.

System action: Processing is terminated.

User response: Check the return area that follows message DSN1717I. One possible problem is:
- The trace has already been stopped from the operator's console or by another application program.

Severity: 8 (error)

DSN1719I   ACTIONS ARE COMPLETE

Explanation: All requested IFC selective dump actions finished. All trace records have been written to the specified destination(s). If an abend action was requested, all abends have taken place.

System action: DSN1SDMP stops the trace that it started and completes processing.

Severity: 0 (informational)

DSN1720I   READA ERROR, REASON=xxxxxxxx

Explanation: An error resulted when DSN1SDMP issued a instrumentation facility interface (IFI) READA call in an attempt to obtain DB2 trace records. Reason code 'xxxxxxxx' is the IFI reason code returned in IFCARC2.

System action: Processing is terminated.

User response: One possible reason code is:

00E60824
> The user does not have correct monitor authority. Obtain the correct monitor authority (MONITOR1 or MONITOR2) and resubmit the job.

Severity: 8 (error)

DSN1800I   START OF DSN1CHKR FOR JOB *job-name step-name*

Explanation: This message indicates that the DSN1CHKR utility started processing.

job-name
> The name of the job that requested DSN1CHKR.

step-name
> The name of the job step that requested DSN1CHKR.

System action: DSN1CHKR starts processing.

User response: No action is required.

Severity: 0

DSN1806I   IDMAP ENTRY entry-number IN PAGE page-number IS IN ERROR

Explanation: This message indicates that the IDMAP entry 'entry-number' in page 'page-number' is not valid. Valid entries are from X'0014' to X'0Fhh' (depending on the number of IDMAP entries in the page).

System action: The page is marked for output if the FORMAT/DUMP option is in effect. Processing continues.

Problem determination: Contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1807I   CHILD POINTER chd-rid IN ERROR IN PARENT RECORD par-rid IN PAGE pppppp

Explanation: This message indicates that the parent record with record ID 'par-rid' in page 'pppppp' has a child pointer with an invalid record ID of 'chd-rid'. If this message is preceded by DSN1835I, then the child pointer is pointing to a page beyond the high used RBA of the data set. Either the child pointer is invalid or the data set has been truncated.

System action: The page is marked for output if the FORMAT/DUMP option is in effect. Processing continues.

Problem determination: Contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1808I   OVERFLOW RECORD IN ERROR - OVERFLOW RID = ovf-rid, POINTER RID = ptr-rid

Explanation: This message indicates that the record at record id 'ovf-rid', pointed to by the pointer record at record id 'ptr-rid', is not an overflow record.

System action: The page is marked for output if the FORMAT/DUMP option is in effect. Processing continues.

Problem determination: Contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1809I   OVERFLOW RECORD IN ERROR - OVERFLOW RID = ovf-rid, HAS NO POINTER RECORD

Explanation: This message indicates no pointer record points to the overflow record with record id 'ovf-rid'.

System action: The page is marked for output if the FORMAT/DUMP option is in effect. Processing continues.

Problem determination: Contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1810I   INPUT DATA SET INVALID, NOT A CHECKABLE SYSTEM TABLESPACE

Explanation: This message indicates that the data set defined by SYSUT1 is not a checkable table space and therefore cannot be analyzed.

System action: Processing terminates.

System programmer response: Correct the SYSUT1 DD statement and rerun the job.

Problem determination: Checkable system table spaces must be referenced by a hashed entry or contain links. Single-record table spaces referenced by an index are not checkable.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSN1811I   ANCHOR POINT PAGE page IS OUT OF RANGE - VALUE IS LESS THAN 2 OR GREATER THAN 'E'X

Explanation: The ANCHOR function has an incorrect page value for the page-slot argument. Anchor points can be found only on pages 2 through 14 (E in hexadecimal). Any anchor point page specified beyond that range is invalid.

System action: Processing continues with the next set of parameters.

Severity: 4 (warning)

DSN1812I   ORPHAN ID = orph-id ID ENTRY = entry-val FOUND IN PAGE page-number

Explanation: This message indicates that the record with id = 'orph-id' and id entry = 'entry-val' in page 'page-number' is an orphan.

System action: Processing continues.

Problem determination: Contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1813I   UNABLE TO DUMP PAGE page-number

Explanation: An error was encountered while retrieving the page and therefore page 'page-number' could not be dumped.

System action: Processing continues.

Problem determination: Examine previous diagnostic and system messages, correct problem, and rerun job.

Severity: 4 (warning)

DSN1814I   **UNEXPECTED END OF DATA AT PAGE page-number**

Explanation: The END-OF-DATA condition was reached before expected at page 'page-number'.

System action: Processing terminates.

Problem determination: Examine previous diagnostic and system messages, correct problem and rerun job.

Severity: 8 (error)

DSN1815I   **DSN1CHKR COMPLETED SUCCESSFULLY,** *pg-cnt* **PAGES PROCESSED**

Explanation: The DSN1CHKR utility successfully completed processing; all checks are completed.

pg-cnt   The number of pages that were read during check processing. This number is not the number of pages in the table space.

System action: Processing terminates.

User response: No action is required.

Severity: 0

DSN1816I   **DSN1CHKR TERMINATED WITH ERRORS,** *pg-cnt* **PAGES PROCESSED**

Explanation: This message indicates that the DSN1CHKR utility did not successfully complete processing.

pg-cnt   The number of pages that were read during check processing. This number is not the number of pages in the table space.

System action: Processing terminates.

User response: Examine previous diagnostic messages. If possible, correct any problems and rerun the job.

Severity: 4 (warning) or 8 (error)

DSN1817I   **ARGUMENT** *arg-val* **OF THE MAP COMMAND IS INVALID**

Explanation: This message indicates that the argument '*arg-val*' of the MAP command is invalid.

System action: Processing terminates.

Problem determination: Check the syntax of the command for invalid arguments.

Examples of invalid arguments are:
- The offset value of the ANCHOR or HASH option is neither zero (0) nor four (4).
- When the ANCHOR and RID option has been specified, the page number specified is less than two, which would be a nondata page.

Severity: 8 (error)

Related reference:

➡ Syntax and options of the DSN1CHKR control statement (DB2 Utilities)

DSN1823I   **INVALID SYNTAX,** *expected-character* **EXPECTED**

Explanation: This message indicates that an unexpected character was found while parsing the parmlist and that *expected-character* was the expected character.

System action: Processing terminates.

Problem determination: Correct the syntax and reissue the command.

Severity: 8 (error)

Related reference:

➧ The DSN command and its subcommands (DB2 Commands)

Related information:

➧ DB2 and related commands (DB2 Commands)

DSN1824I   INVALID HEX DATA: hex-data

Explanation: This message indicates that the input value 'hex-data' is not a valid hexadecimal number.

System action: Processing terminates.

Problem determination: Correct input value and rerun job.

Severity: 8 (error)

DSN1825I   INPUT VALUE EITHER TOO LONG OR OMITTED: in-val

Explanation: This message indicates that the input value 'in-val' is either greater than 8 hexadecimal digits or was totally omitted.

System action: Processing terminates.

Problem determination: Correct input value and rerun job.

Severity: 8 (error)

DSN1830I   INVALID NUMBERS OF VALUES SPECIFIED FOR *func-name*

Explanation: This message indicates that either:

• No value was specified.
• An odd number of values was specified for a function that requires an even number of values.

System action: Processing terminates.

Problem determination: Check the syntax of the parameter list, correct the parameter list, and rerun the job.

Severity: 8 (error)

Related concepts:

➧ DB2 online utilities (DB2 Utilities)

➧ DB2 stand-alone utilities (DB2 Utilities)

DSN1831I   END OF DATE REACHED, PAGE = page-number IS OUT OF RANGE

Explanation: This message indicates that page 'page-no', specified as a parameter of the MAP option, caused the END-OF-DATA condition to be reached. The page number is invalid for the table space being mapped.

System action: Processing continues with the next set of parameters.

Problem determination: Correct the page parameter of the MAP option and rerun the job.

Severity: 4 (warning)

DSN1833I   ANCHOR POINT IS NOT IN USE

Explanation: The anchor point to be mapped has a value of 0, which means that it is not in use. Therefore, no links are checked.

System action: Processing continues with the next set of parameters.

Severity: 4 (warning)

DSN1834I   **GETMAIN FOR BUFFERS FAILED. NO MORE AVAILABLE STORAGE IN ADDRESS SPACE.**

Explanation: The GETMAIN for page buffers failed. There is no more virtual storage available in the address space. The table space being checked is too large to be processed with the current system configuration.

System action: Processing terminates.

Problem determination: Make changes to the system configuration, if possible, to allow more virtual storage to be available in the address space.

Severity: 8 (error)

DSN1835I   **THE REQUESTED PAGE page-no IS BEYOND THE END OF THE TABLESPACE.**

Explanation: A request for a page number from an anchor point or a child pointer in a parent record is pointing beyond the high used RBA of the data set. Either the rid is in error or the data set has been truncated.

System action: The page is marked for output if the FORMAT/DUMP option is in effect. Processing continues.

Problem determination: Examine previous diagnostic and system messages, correct problem if possible and rerun the job. Otherwise, contact your IBM Support Center for assistance.

Severity: 4 (warning)

DSN1930I   *csect-name* **THE INPUT PARAMETER** *xxxxxxxx* **CONFLICTS WITH THE HEADER PAGE FIELD HPG***yyyy***. THE INPUT PARAMETER WILL BE USED**

Explanation: The specified parameter does not agree with the value of the related field in the header page. The utility will continue the execution using the specified input parameter.

System action: The utility continues

User response: Confirm that the input parameter has the correct value.

Severity: 0 (informational)

DSN1931I   *csect-name* **THE INPUT PARAMETER** *xxxxxxxx* **CONFLICTS WITH THE HEADER PAGE FIELD HPG***yyyy***. THE HEADER PAGE VALUE WILL BE USED**

Explanation: The specified parameter does not agree with the value of the related field in the header page. The utility will continue the execution using the header page value.

System action: The utility continues

User response: Confirm that the input parameter has the correct value.

Severity: 0 (informational)

DSN1937I   *csect-name* **TXT-DECK** *deck-name* **BUILT** *record-count* **RECORDS WRITTEN**

Explanation: A text deck has been generated from the dictionary and written to the data set that is specified by the data definition statement DSN1DICT.

csect-name
        The name of the control section that issued the message.

deck-name
        The name of the new text deck. This value matches the value of the invocation parameter EXTNDICT.

record-count
        The number of records that were written.

System action: Processing terminates.

User response: No action is required.

Severity: 0

DSN1938I   *csect-name* **ERROR ON DDNAME/DATASET FOR EXTERNALIZED DICTIONARY DSN1DICT**

Explanation: An error was encountered during an attempt to write to the data set that is specified with the data definition statement DSN1DICT.

System action: Processing terminates.

User response: Verify that the name of the correct data definition statement has been supplied in the JCL stream and that the data set attributes are correct. Review other messages in the job output. Correct the cause of the problem, and resubmit the job.

Severity: 8 (error)

DSN1939I   *csect-name* **AN INVALID CI SIZE HAS BEEN DETECTED**

Explanation: The VSAM Control Interval (CI) size for the output data set is invalid. The CI size must be 4K, or it must match the page size.

System action: Processing terminates.

System programmer response: Redefine the output VSAM data set so that the CI size is either 4K or equal to the page size of the object that is being processed.

Severity: 8 (error)

DSN1940I   *csect-name* **COMPRESSION REPORT**

Explanation: DSN1COMP completed successfully. This report contains the compression savings for every supported page size.

The compression report returns the following statistics:
- the number of leaf pages processed
- the number of keys processed
- the number of RIDS processed
- the KB of key data processed
- the KB of compressed keys processed

An evaluation report follows. The evaluation report contains estimates of compression savings for every supported page size. The report format is as follows:
- the index page size entered
- the estimated reduction in index leaf page space
- the original index's leaf page space
- the amount of unused bufferpool space

Example output:

```
8  K Page Buffer Size yields a
51 % Reduction in Index Leaf Page Space
   The Resulting Index would have approximately
49 % of the original index's Leaf Page Space
   No Bufferpool Space would be unused
```

System action: The utility completed successfully.

Severity: 0 (informational)

Related reference:

➥ DSN1COMP (DB2 Utilities)

➥ DSN1COMP output (DB2 Utilities)

DSN1941I   *csect-name* **COMPRESSION ESTIMATE WAS NOT CALCULATED**

Explanation: There is insufficient data to build a compression dictionary. Either the input data set is empty, or the number of rows in the input data set was insufficient to build the dictionary.

System action: Processing terminated.

User response: Increase the number of rows in the input data set.

Severity: 8 (error)

DSN1942I   *csect-name* **ERROR WHEN BUILDING THE COMPRESSION DICTIONARY**

Explanation: An internal error was encountered while building the dictionary.

System action: Processing terminated.

System programmer response: Determine the failing environment. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Severity: 8 (error)

DSN1943I   *csect-name* **DICTIONARY PAGE ENCOUNTERED**

Explanation: A dictionary page was encountered while processing the input data set. This indicates that the data is in compressed format.

System action: Processing terminated.

User response: Run DSN1COMP on a data set containing data that is not in a compressed format.

Severity: 8 (error)

DSN1944I   *csect-name* **INPUT PARAMETERS**

Explanation: The input parameters used by DSN1COMP are either user-supplied or default values.

System action: Processing continues.

Severity: 0 (informational)

DSN1945I   *csect-name* **DICTIONARY TREE**

Explanation: The compression dictionary tree follows. Each node is formatted in the SYSPRINT data set.

System action: Processing continues.

Severity: 0 (informational)

DSN1946I   *csect-name* **IMAGE COPY CONTAINING MULTIPLE PARTITIONS**

Explanation: DSN1COMP will build a compression report given sufficient data in the first partition only, of an Image Copy containing multiple partitions.

System action: Processing terminated.

User response: Run DSN1COMP on an image copy containing a single partition.

Severity: 8 (error)

DSN1947I   DSN1947I *csect-name* **COMPRESSION UNAVAILABLE FOR INDEX SPACES**

Explanation: Index data was provided as input to DSN1COMP. Compression is supported only for table space data.

System action: Processing terminated.

User response: Provide DSN1COMP with data containing table space data.

Severity: 8 (error)

DSN1948I *csect-name* **UNABLE TO RESET DICTIONARY VERSION**

Explanation: DSN1COPY cannot reset the dictionary version for one of the reasons below.

- The dictionary version was not found.
- The input data set is an incremental or an inline image copy.

System action: Processing continues.

User response: DSN1COPY was not able to reset the dictionary version. Run the REORG utility with KEEPDICTIONARY=NO to reset this value.

Severity: 4 (warning)

Related concepts:

➜ Data compression (DB2 Utilities)

Related reference:

➜ DSN1COPY (DB2 Utilities)

➜ REORG TABLESPACE (DB2 Utilities)

DSN1950I **DSN1COPY HAS SUPPRESSED VSAM OPEN VERIFY, ACBERRFLG=acb-error-code.open-option**

Explanation: DSN1COPY suppresses the VSAM VERIFY done by the VSAM OPEN for the SYSUT1 data set. The VSAM VERIFY is not needed for the VSAM data sets used by DB2. For DB2 VSAM data sets, the data set hi-used RBA in the ICF catalog is always current.

System action: Processing continues.

Severity: 0 (informational)

DSN1951I **ODD NUMBER OF HEX CHARACTERS SPECIFIED IN VALUE PARM.**

Explanation: The VALUE parameter contained an odd number of hexadecimal characters. Only even numbers of hexadecimal characters are allowed.

System action: Processing terminates.

User response: Correct the number of hexadecimal digits specified for the VALUE parameter and resubmit the job.

Severity: 8 (error)

Related reference:

➜ Syntax and options of the DSN1COPY control statement (DB2 Utilities)

DSN1952I **UNEXPECTED END OF INPUT DATA - CHECK FOR INVALID SPECIFICATION OF THE 32K PARAMETER.**

Explanation: End of data occurred for the image copy data set specified in SYSUT1 before the first page could be read. Either the wrong input data set was specified or 32KB was specified even though the input data set contains 4KB pages.

System action: Processing terminates.

User response: Remove the 32KB parameter if the input data set contains 4KB pages, change the name of the input data set to specify a valid image copy, or remove the FULLCOPY or INCRCOPY parameter.

Problem determination: Refer to DSN1COPY documentation for parameter requirements.

Severity: 8 (error)

DSN1953I   STARTING PAGE NUMBER REQUIRES AN INPUT DATA SET NAME OF data-set-name

Explanation: The starting page number of the page range to be printed is inconsistent with the input data set specified.

System action: Processing terminates.

User response: Ensure that NUMPARTS, the 32KB parameter (or its default of 4KB), and the starting page number are correct. If it is correct, change the input data set name as indicated by the message.

Problem determination: Refer to DSN1COPY or DSN1PRNT documentation for parameter requirements.

Severity: 8 (error)

DSN1954I   csect-name PARAMETER PAGESIZE OR LOB IS MISSING OR INCORRECTLY SPECIFIED.

Explanation: This message is issued in the following situations:
- The program tried to determine the page size from the first page of the data set, but the first page is not a header page, or
- The first page is a LOB header page and the LOB parameter is missing.

System action: Processing terminates.

User response: If the input data set is not a LOB table space, specify the PAGESIZE parameter. If the input data set is a LOB table space, specify the LOB or PAGESIZE parameter or both, as appropriate.

Severity: 8 (error)

DSN1955I   SYSUT1 DD CARD IS MISSING

Explanation: No SYSUT1 DD statement was provided.

System action: Processing terminates.

User response: Rerun the job with a SYSUT1 DD statement.

Severity: 8 (error)

DSN1956I   *csect-name* SYNTAX ERROR: *parm2* IS SPECIFIED WITHOUT *parm1*

Explanation: The keyword *parm1* must be specified with *parm2*.

System action: Processing terminates.

User response: Resubmit the job specifying the correct keyword.

Severity: 8 (error)

DSN1957I   *csect-name* COMPRESSED RECORDS CANNOT BE EXPANDED DUE TO INCORRECT INPUT
                DATA SET SPECIFICATION.

Explanation: DSN1PRNT issues this message when the EXPAND option is specified. The input data set might be an incremental image copy data set.

System action: Processing terminates.

User response: Resubmit the job specifying the correct input data set.

Severity: 8 (error)

DSN1958I   INVALID DATA SET NAME.

Explanation: A DB2 table space was given an incorrect name.

System action: Processing terminates.

User response: Correct the output data set name if FULLCOPY or INCRCOPY were specified. If a page range is being printed, correct the input data set name.

Severity: 8 (error)

Related concepts:

➦ Managing your own data sets (DB2 Administration Guide)

Related tasks:

➦ Defining data sets (DB2 Administration Guide)

Related reference:

➦ Data set naming conventions (DB2 Administration Guide)

DSN1959I   **BOTH FULLCOPY AND INCRCOPY PARAMETERS CANNOT BE SPECIFIED.**

Explanation: DSN1COPY cannot apply to both full images copies and incremental image copies in the same execution. The full image copies are applied first. Each incremental image copy must then be applied in separate steps, starting with the oldest incremental copy.

System action: Processing will terminate.

User response: Omit either FULLCOPY or INCRCOPY.

Severity: 8 (error)

DSN1960I   *csect-name input/output* **DATA SET ALLOCATION ERROR. ERROR FIELD IS ERRFIELD** *value1***. INFORMATION FIELD IS INFFIELD** *value2***.**

Explanation: The allocation of an input or output data set failed. The ERRFIELD *value1* contains the error number; the INFFIELD *value2* contains the information number. See SVC99 error codes in *z/OS MVS Diagnosis: Reference* or *z/OS MVS Programming: Authorized Assembler Services Guide* for help in determining the cause of the error.

System action: Processing terminates.

User response: Correct the cause of the allocation error and resubmit the job.

Severity: 8 (error)

DSN1961I   **PIECE NUMBER value IS INVALID.**

Explanation: A page was retrieved that has a piece number greater than NUMPARTS for partitioned table spaces or greater than the default of 32 for linear table spaces. 'DPIECNUM value' is the piece number for the retrieved page. For page range printing, this message is issued if the starting page number has an invalid piece number as determined from calculations involving NUMPARTS and the page size.

System action: Processing will terminate.

User response: Look in SYSIBM.SYSTABLESPACE for the correct number of partitions for this page. Correct NUMPARTS or 32KB if they were specified incorrectly, resubmit the job with a valid image copy, or correct the starting page number for page range printing.

Severity: 8 (error)

DSN1962I   **OUTPUT DATA SETS MUST BE VSAM WHEN IMAGE COPY DATA SETS ARE USED AS INPUT TO DSN1COPY.**

Explanation: If image copy data sets are used as input to DSN1COPY, the output data sets must be VSAM. QSAM data sets cannot be used for output.

System action: Processing will terminate.

User response: Change the output data set to VSAM, or remove the FULLCOPY and NUMPARTS parameters.

Severity: 8 (error)

DSN1963I   ZERO PAGES ARE INVALID WHEN IMAGE COPY DATA SETS ARE USED AS INPUT TO STAND ALONE UTILITIES.

Explanation: A page in an IMAGE COPY data set has a zero header. Pages with zero headers are illegal if image copy data sets are used as input to stand-alone utilities.

System action: Processing continues.

User response: If the image copy data set name was invalid, look in SYSIBM.SYSCOPY for valid names. Resubmit the valid IMAGE COPY data set to the stand-alone utility. If the IMAGE COPY data set had a valid name, use the REPAIR utility to replace the zero pages with valid data pages.

Severity: 4 (warning)

DSN1964I   NO VALUE HAS BEEN SPECIFIED FOR THE NUMBER OF PARTITIONS.

Explanation: No value was given for the number of partitions. This value should follow the 'NUMPARTS' parameter.

System action: Processing will terminate.

User response: Look in SYSIBM.SYSTABLESPACE for the correct number of partitions for this page. Specify the correct number of partitions and resubmit the job. If the page is not partitioned, then either omit the NUMPARTS parameter or specify NUMPARTS as zero.

Severity: 8 (error)

DSN1965I   THE PARTITION NUMBER VALUE IS INVALID.

Explanation: The partition number value following the 'NUMPARTS' parameter is not between 1 and 64.

System action: Processing will terminate.

User response: Look in SYSIBM.SYSTABLESPACE for the correct number of partitions for this page.

Severity: 8 (error)

DSN1966I   OUTPUT DATA SET *odsname* NOT FOUND

Explanation: DB2 cannot find the output data set *odsname* because it has not yet been defined.

If DSN1COPY terminates and issues this message, DSN1COPY attempts to determine the correct target data set or data sets that affect the data set number. (The data set number is the last qualifier in the data set name. For example, the data set number in the data set DSNCAT.DSNDBD.TESTDB.TS01.I0001.A002 is A002.)

When DSN1COPY issues this message, *odsname* might be different than the output data set defined by SYSUT2. SYSUT1 can contain data that should be copied to data sets other than the one specified by SYSUT2. DSN1COPY does not define data sets, so the output data set must be defined before DSN1COPY is run.

System action: Processing is terminated.

User response: Use access method services to define *odsname* and any other VSAM data sets that have not yet been defined.

Severity: 8 (error)

DSN1967I   THE OBID ARRAYS ARE FULL

Explanation: More than 1000 records of OBIDs were specified in the SYSXLAT file.

System action: Processing stops.

Problem determination: DSN1COPY cannot be used for table spaces containing more than 1000 tables.

Severity: 8 (error)

DSN1968I   OBIDS SHOULD NOT BE DUPLICATED

Explanation: There should not be any repeated source OBIDs or target OBIDs within the SYSXLAT file.

System action: Processing is terminated.

User response: Change the SYSXLAT file.

Severity: 8 (error)

DSN1969I   SYSXLAT FILE CAN NOT BE OPENED

Explanation: The QSAM OPEN for the SYSXLAT file failed because no DD statement was provided.

System action: Processing is terminated.

System programmer response: Provide a SYSXLAT DD statement if OBID translate is to be done. Otherwise, remove the SYSXLAT parameter and rerun.

User response: Notify system programmer.

Problem determination: Follow standard QSAM diagnostic procedures.

Severity: 8 (error)

DSN1970I   DATA PAGE CONTAINING OBID xxxx, HAS NO MATCHING VALUE IN THE SYSXLAT FILE

Explanation: One of the OBID records of the SYSXLAT file contains a source OBID value that does not match the OBID value of the source system.

System action: Processing is terminated.

User response: Provide that record of the SYSXLAT file which contains an unmatched OBID value with an OBID that matches the OBID value of the source system.

This error can also be caused when an OBIDXLAT is attempted on an incremental image copy of a compressed segmented table space.

Severity: 8 (error)

Related concepts:

The effects of not specifying the OBIDXLAT option (DB2 Utilities)

Related reference:

Syntax and options of the DSN1COPY control statement (DB2 Utilities)

DSN1971I   DBID xxxx NOT FOUND IN THE SYSXLAT FILE. REPLACE SOURCE DBID xxxx OF THE
               SYSXLAT FILE WITH IT.

Explanation: The first record of the SYSXLAT file has the incorrect source DBID value.

System action: Processing is terminated.

User response: Provide the first record of the SYSXLAT file with a source DBID value which matches the DBID value of the source system.

Severity: 8 (error)

DSN1972I   PSID xxxx NOT FOUND IN THE SYSXLAT FILE REPLACE SOURCE DBID xxxx OF THE SYSXLAT
               FILE WITH IT

Explanation: The second record of the SYSXLAT file has the incorrect source PSID value.

System action: Processing is terminated.

User response: Provide the second record of the SYSXLAT file with the correct PSID value.

Severity: 8 (error)

DSN1973I   SOURCE DBID IS MISSING

Explanation: The first record of the SYSXLAT file contains no DBID value.

System action: Processing is terminated.

User response: Provide the first record of the SYSXLAT file with a pair of source and target DBID values which match the DBID values of the source and target systems.

Severity: 8 (error)

DSN1974I   TARGET DBID IS MISSING

Explanation: The first record of the SYSXLAT file contains no target DBID value.

System action: Processing is terminated.

User response: Provide the first record of the SYSXLAT file with a target DBID value which matches the DBID value of the target system.

Severity: 8 (error)

DSN1975I   SOURCE PSID IS MISSING

Explanation: The second record of the SYSXLAT file contains no PSID value.

System action: Processing is terminated.

User response: Provide the second record of the SYSXLAT file with a pair of source and target PSID values which match the PSID values of the source and target systems.

Severity: 8 (error)

DSN1976I   TARGET PSID IS MISSING

Explanation: The target PSID is missing from the second record of the SYSXLAT file.

System action: Processing is terminated.

User response: Provide the second record of the SYSXLAT file with a target PSID value which matches the PSID values of the target system.

Severity: 8 (error)

DSN1977I   SOURCE OBID IS MISSING

Explanation: There is a record in the SYSXLAT file with no OBID value.

System action: Processing is terminated.

User response: Correct the SYSXLAT file so that all the records following the first and second records will contain a pair of source and target OBID values.

Severity: 8 (error)

DSN1978I   TARGET OBID IS MISSING

Explanation: There is a record in the SYSXLAT file with no target OBID value specified.

System action: Processing is terminated.

User response: Provide the SYSXLAT file with an OBID value that matches the OBID value of the target system.

Severity: 8 (error)

DSN1979I   *csect-name* **SOURCE & TARGET VALUES MUST BE WITHIN THE RANGE OF 1 TO 65535.**

Explanation: The SOURCE and TARGET values of the SYSXLAT file must be between 1 and 65535.

System action: Processing is terminated.

User response: Provide the SYSXLAT file with values that are between 1 and 65535.

Severity: 8 (error)

DSN1980I   **BROKEN PAGE BIT SET, WILL RESET AND RETRY BROKEN PAGE TEST**

Explanation: This message indicates that a broken page was detected. If message DSN1990I is issued following this message, determine the cause of the broken page. If message DSN1990I is not issued, reset the broken page bit with the REPAIR utility to correct the problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

System action: Processing continues.

Operator response: Notify the system programmer.

System programmer response: Use either the RECOVER or REPAIR utility to correct the broken page. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination should have been done when the dump associated with the setting of the broken page bit was requested.

Severity: 4 (warning)

DSN1981I   **INVALID PARAMETER FIELD: bad-parameter**

Explanation: An invalid input parameter was detected on the EXEC statement. The parameter is either not a valid keyword or it has invalid subparameters associated with it.

Examples of invalid subparameters are:
- The NUMPARTS subparameter was other than '(nn)'. 'nn' is a decimal number.
- The NUMPARTS subparameter was missing either a left or right parenthesis.
- The PRINT subparameter was missing either a left or right parenthesis.
- The PRINT subparameter specified a digit in a page number which was not hexadecimal.
- The PRINT subparameter contained a starting page number that is larger than the ending page number.
- The PRINT subparameter contains a page number larger than six hexadecimal digits.

System action: Processing terminates.

User response: Correct the indicated parameter on the EXEC DD statement, and resubmit the job.

Problem determination: Refer to DSN1COPY, DSN1PRNT, or DSN1COMP documentation for parameter requirements.

Severity: 8 (error)

DSN1982I   **NO BLANKS ARE ALLOWED WITHIN THE PARAMETER LIST**

Explanation: There is a space or spaces between one or more parameters in the utility list.

System action: Processing is terminated.

User response: Check the parameter list and make sure that there are no spaces between any of the parameters.

Severity: 8 (error)

DSN1983I   *csect-name* **DICTIONARY IS INVALID, REASON= X'***nnnn***'**

Explanation: The dictionary found in the data set is invalid. A DB2 internal error occurred during the validation of the compression dictionary.

System action: Processing terminates.

System programmer response: Determine the failing environment. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Verify that the correct data set is specified. You can delete the EXPAND option and resubmit the job.

Problem determination: The REASON identifies the nature of the error. The REASON is used for diagnostic purposes by the IBM Support Center.

Severity: 8 (error)

DSN1984I   **UNEXPECTED PAGE NUMBER, EXPECTING:** *expected-page-number* **FOUND:** *page-number-found*

Explanation: This message indicates that an unexpected page number was encountered. This message might be received if the input data set is an image copy but you did not specify the FULLCOPY or INCRCOPY parameters. The page numbers displayed in hexadecimal format are the page numbers that are expected and found. After this message is issued five hundred times, it will stop being issued.

System action: Processing continues.

System programmer response: If running DSN1COPY use DSN1PRNT to dump the table space or index space. Otherwise, analyze the DSN1PRNT output. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

User response: Notify the system programmer.

Severity: 4 (warning) for DSN1PRNT or 8 (error) for DSN1COPY

DSN1985I   **ZERO PAGES ENCOUNTERED. FIRST PAGE = first-zero-page-number, LAST PAGE = last-zero-page-number**

Explanation: One or more pages zero pages were encountered with valid data pages after them. A zero page is a page of all zeros or a page with only the page-synchronization bit on in the first and last byte of the page. The 'first zero page number' and the 'last zero page number' displayed in hexadecimal are the pages where zeros were first encountered and last encountered. In most cases, this is not an error condition. This is a normal condition for segmented table spaces and also for index spaces created with a FREEPAGE parameter not equal to zero. In addition, it is also normal for nonsegmented table spaces to contain zero pages after certain update scenarios terminated abnormally and data rollback was required.

System action: Processing continues.

System programmer response: If the zero page is the result of segmenting, no action is necessary. Similarly, if the zero page is a valid zero page that was introduced by DB2, no action is necessary as this page is free space and is used by DB2 when needed. If the zero page is a valid zero page and you want to remove it from the table space, run the REORG utility.

To help determine if the zero page was validly introduced by DB2, run the CHECK INDEX utility. This assumes that at least one index is defined on the affected table space.

If the zero page is determined to be invalid, you need to run the RECOVER utility.

User response: Notify the system programmer.

Severity: 4 (warning)

DSN1986I **NEITHER FULLCOPY NOR INCRCOPY PARAMETERS MAY BE SPECIFIED WITH THE SEGMENT PARAMETER.**

Explanation: The parameters SEGMENT, FULLCOPY, and INCRCOPY describe the type of input that SYSUT1 references. SEGMENT is only relevant when a table space or index space is the input. FULLCOPY and INCRCOPY are only relevant when a image copy is the input. If the input data set is a segmented table space, then the FULLCOPY or INCRCOPY parameters should not be specified. If the input data set is a full or incremental image copy, then SEGMENT should not be specified.

System action: Processing will terminate.

User response: Omit the incorrect parameter.

Severity: 8 (error)

DSN1987I *csect-name* **INDEX PAGE CONTAINING OBID** *id* **HAS NO MATCHING VALUE IN THE SYSXLAT FILE.**

Explanation: This message is issued when the OBID on the page does not find a matching value in the input source OBID list. This message can also be issued when the input does not have enough OBID values.

System action: Processing terminates with a dump of the page on which the OBID does not find a match in the source list.

User response: Resubmit the job with correct input OBID values.

Severity: 8 (error)

DSN1988I *csect-name* **EXPAND KEYWORD IS SPECIFIED BUT DICTIONARY IS NOT FOUND**

Explanation: DSN1PRNT issues this message when the EXPAND option is specified. The dictionary was not found in the data set. An incorrect data set piece is specified or an incorrect data set number is specified for a multi-piece linear data set.

System action: Processing terminates.

User response: Resubmit the job specifying the correct keyword or the correct data set name.

Severity: 8 (error)

DSN1989I *csect-name* **IS PROCESSED WITH THE FOLLOWING OPTIONS:**

Explanation: The indicated stand-alone utility is processing with the indicated options. These options were either specified or supplied by default.

csect-name
        The name of the control section that issued the message.

The options include:
```
check/32K/fullcopy/incrcopy/segment/
numparts=n/format/expand/
print/value/obidxlat/reset/no check/4K/
no imagecopy/nonsegment/numparts=0/
no print/no value/no obidxlat/
no format/no reset/no expand
```

System action: Processing continues.

User response: Verify that the parameters are correct.

Severity: 0

Related concepts:

⮕ DB2 stand-alone utilities (DB2 Utilities)

DSN1990I   *csect-name* **BROKEN PAGE. PAGE NUMBER =** *broken-page-number*. **REASON:** *comment*, **SERVICE INFO: CODE =** *c1, d1 = v1 ... dn = vn*

Explanation: During page checking, the page was detected as broken for the reason given in the message. The page number is displayed in hexadecimal format.

System action: Processing continues.

Operator response: Notify the system programmer.

System programmer response: Use either the RECOVER or REPAIR utility to correct the broken page.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Service information is used for diagnostic purposes by the IBM Support Center.

Severity: 4 (warning)

DSN1991I   **UNCLUSTERED DATA DETECTED. RID:** *record-ID* **TABLE:** *table-name* **INDEX KEY:** *key-name*

Explanation: The arrangement of data in one of the following DB2 catalog table spaces is not optimal for minimizing locking contention:
- SYSDBASE
- SYSPLAN
- SYSVIEWS

Data is not broken. Only the arrangement is affected.

record-ID
        The record ID in the format 'xxxxxxxx'X for the duplicate record found.

table-name
        The name of the table for which data is unclustered.

key-name
        The index key that identifies the row that is unclustered.

Normally, data on these pages is arranged to minimize locking contention. DSN1COPY found that the arrangement of data no longer minimizes locking contention for the specified objects. Therefore, an increase in the number of deadlocks and timeouts may be seen on SYSDBASE, SYSPLAN, and SYSVIEWS.

System action: Processing continues.

Administrator response: Run the REORG TABLESPACE utility on the relevant catalog table space to rearrange the data.

DSN1992I   **VSAM PUT ERROR, RPLERREG =** *return-code*, **RPLERRCD =** *reason-code*

Explanation: VSAM PUT failed with the indicated error code.

System action: Processing terminates.

System programmer response: Follow standard VSAM diagnostic procedures. See *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the standard VSAM procedures, the RPL fields, and the VSAM control blocks.

User response: Notify system programmer.

Severity: 8 (error)

DSN1993I   **DSN1COPY/DSN1PRNT/DSN1COMP TERMINATED,** *page-count* **PAGES PROCESSED**

Explanation: This message indicates that a stand-alone utility has not successfully completed processing. The page count is displayed in decimal format.

System action: Processing terminates.

System programmer response: Correct the previous error and resubmit the job.

User response: Examine the previous error message of the application program.

Severity: 8 (error)

DSN1994I   **DSN1COPY/DSN1PRNT/DSN1COMP COMPLETED SUCCESSFULLY,** *page-count* **PAGES PROCESSED**

Explanation: This message indicates that a stand-alone utility successfully completed processing. The page count is displayed in decimal format.

System action: Processing terminates.

Severity: 0 (informational)

DSN1995I   **VSAM GET ERROR, RPLERREG =** *return-code*, **RPLERRCD =** *reason-code*

Explanation: VSAM GET failed with the indicated error code.

System action: Processing terminates.

System programmer response: Follow standard VSAM diagnostic procedures. See *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the standard VSAM procedures, the RPL fields, and the VSAM control blocks.

User response: Notify system programmer.

Severity: 8 (error)

DSN1996I   **VSAM OPEN ERROR, ACBERRFLG = ACB** *error-code*. **OPEN** *option*

Explanation: VSAM OPEN failed with the indicated 'ACB error-code'. The OPEN option was either 'INPUT' or 'OUTPUT'.

System action: Processing terminates if error code is greater than or equal to 128; otherwise, processing continues normally.

System programmer response: Follow standard VSAM diagnostic procedure. See *z/OS DFSMS: Macro Instructions for Data Sets* for more information on the standard VSAM procedures and the VSAM control blocks.

User response: Notify system programmer.

Severity: 8 (error) if error is greater than or equal to 128; otherwise, severity code is 4.

DSN1997I   **OUTPUT DSNAME =** *data-set-name*, *data-set-type*

Explanation: The indicated 'data set name' is used for output. 'data set type' is either VSAM or SEQ (for sequential).

Severity: 0 (informational)

DSN1998I   **INPUT DSNAME =** *data-set-name*, *data-set-type*

Explanation: The indicated 'data-set-name' is used for input. 'data-set-type' is either VSAM or SEQ (indicating sequential).

Severity: 0 (informational)

DSN1999I   **START OF DSN1COPY/DSN1PRNT/DSN1COMP FOR JOB** *job-name step-name*

Explanation: This message indicates that a stand-alone utility started processing.

System action: The stand-alone utility starts processing.

Severity: 0 (informational).

Chapter 23. DSN3 messages

Message identifiers that begin with DSN3 communicate information about the subsystem support subcomponent.

DSN3001I   *csect-name* **ABNORMAL DISCONNECT FROM SUBSYSTEM INTERFACE**

Explanation: This message indicates that 'csect-name' detected that an online routine was still supporting SSI calls (IEFSSREQ) even though DB2 was nearly done terminating or no longer executing. This occurs with 'csect-name' DSN3RS0Xwhen the service address space has reached end-of-memory and neither normal termination nor online error recovery routines have successfully completed termination of the DB2 subsystem. This occurs with 'csect-name' DSN3SSTM when this condition is discovered during online termination.

System action: The connection is terminated. All IEFSSREQ requests are handled by the DB2 ERLY code until DB2 is restarted.

Problem determination: An SVC dump is requested. This SVC dump and associated SYS1.LOGREC entries provide information to determine which resource managers had not removed their support of particular SSI function codes.

For dumps requested by DSN3RS0X,all pertinent control blocks and stack storage are dumped. The primary control block of interest is the SSVT for DB2.

For DSN3SSTM, the DB2 'standard dump' facility is used to dump the information necessary for problem diagnosis. The primary control blocks of interest are the 'active' CONN blocks.

DSN3002I   **RECOVERY INCOMPLETE FOR connection-name**

Explanation: At least one DB2 unit of recovery (UR), related to an identified subsystem ('connection-name'), is still 'indoubt' after restart synchronization has taken place. ('Indoubt' URs are those for which commit has been voted by DB2 but which have not been acknowledged by 'connection-name'.)

This message may appear if the 'connection-name' subsystem has begun to do new work before having resolved all indoubt URs. The 'connection-name' subsystem may be in the process of resolving the indoubt URs.

System action: Resources held (locked) by these 'indoubt' URs are unavailable to any other work units until their status is resolved.

System programmer response: The system programmer and/or system administrator must determine the correct recovery action to resolve the 'indoubt' situations. This involves either 'ensure-commit' or 'backout' decisions for all 'indoubt' URs.

The -DISPLAY THREAD command should be used to see the URs still indoubt. It may show that all indoubt URs have already been resolved. The -RECOVER INDOUBT command should be used, either from an MVS console or through an attachment facility, to resolve the indoubt URs and to release the resources they hold.

Problem determination: This error is probably caused by a cold start after an abnormal termination of either subsystem or by offline alterations of the logs of either subsystem.

DSN3004I   **DESCRIPTOR GET FAILURE RC=X xxxxxxxx, REASON=X xxxxxxxx**

Explanation: The parameter list required to create the DB2 agents under which IEFSSREQ calls are processed could not be obtained from the subsystem support subcomponent directory of internal parameters (DSN3DIR1). The RC indicates the return code received from the system parameter manager and REASON indicates the reason code (from register 0).

The return code is always 4 when this message is presented, indicating that the IEFSSREQ descriptor was not found in the subsystem support subcomponent directory. This is a system error and results in termination of the DB2 subsystem. The reason code is always 0.

System action: Subsystem initialization is discontinued. Termination occurs.

System programmer response: Repair the master directory (DSNZPARM) and/or the DB2 program libraries, and restart the DB2 subsystem.

Problem determination: If the problem persists, use any dumps and the console log to diagnose the problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSN3006I   xxxx SSI FUNCTION WAS ALREADY ACTIVE WHEN ACTIVATE WAS ATTEMPTED

Explanation: An initialization sequence error has occurred. The subsystem support for SSI services resource manager (SSSS) has discovered during its initialization that another resource manager ('xxxx') has already activated the EOM or FEOT SSI call by calling DSN3AC0X. If this occurs, the error is a serious one.

System action: Subsystem initialization is discontinued. DB2 termination occurs.

System programmer response: Ensure that all maintenance has been applied to the DB2 program libraries, and then restart the DB2 subsystem.

Problem determination: If the problem persists, use any dumps and the console log to diagnose the problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSN3007I   xxxx SSI FUNCTION WAS ALREADY INACTIVE WHEN DEACTIVATE WAS ATTEMPTED

Explanation: A termination sequence error has occurred. The subsystem support for SSI services resource manager (SSSS) has discovered during its termination that another resource manager ('xxxx') has already deactivated the EOM or FEOT SSI call by calling DSN3AC0X. If this occurs, the error is a serious one.

System action: Termination continues.

System programmer response: Ensure that all maintenance has been applied to the DB2 program libraries.

Problem determination: If the problem persists, use any dumps and the console log to diagnose the problem. If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

DSN3008I   *csect-name* ABNORMAL DISCONNECT FOR PROGRAM REQUEST HANDLER(S)

Explanation: This message indicates that 'csect-name' has discovered that one or more resource managers are still supporting application program calls through their program request handler, even though DB2 was almost done terminating or was no longer executing. This occurs when the system services address space has gone to end of memory and neither normal termination nor online error recovery routines have successfully completed termination of the DB2 subsystem.

System action: The program request handler (PRH) connection is terminated. All application program support requests are rejected with an indication that DB2 is not up.

System programmer response: See the Problem Determination section of this message.

Problem determination: An SVC dump is requested. This SVC dump and associated SYS1.LOGREC entries provide information to determine which resource managers were still supporting application program support calls. All pertinent control blocks and stack storage are dumped. The primary control block of interest is the program request handler block (PRHB).

DSN3010I   *csect-name* PARTICIPANT RRS IS COLD STARTING AND HAS LOST ITS LOG. OTHER
             RECOVERABLE RESOURCE MANAGERS MIGHT HAVE INDOUBT UNITS OF RECOVERY.

Explanation: DB2 has coordinator responsibility for indoubt units of recovery and RRS has a participant role in the unit of recovery. RRS has cold started and has lost knowledge of indoubt units of recovery. There might be damage at RRS Recoverable Resource Managers that also participated in the unit of recovery.

System action: Processing continues.

System programmer response: Contact the database administrator.

Database Administrator Action: DB2 is the coordinator for one or more indoubt units of recovery at other RRS recoverable resource managers. RRS has lost all knowledge of these units of recovery. The participant RRS

Recoverable Resource Managers might still have indoubt units of recovery. DB2 has no knowledge of other RRS Recoverable Resource Manager except for the RRS unit of recover ID (URID) that identifies such a unit of recovery. Subsequent DSN3017I, DSN3018I or DSN3019I messages describe the status of each unit of recovery and contain the RRS URID for each.

DSN3011I   *csect-name* **COORDINATOR RRS IS COLD STARTING AND HAS LOST ITS LOG. INDOUBT DB2 THREADS NEED MANUAL RESOLUTION.**

Explanation: DB2 has participant responsibility for indoubt threads. RRS, the commit coordinator, has informed DB2 that it lost all knowledge of DB2 indoubt threads. The indoubt threads at this DB2 must be manually resolved with the RECOVER INDOUBT command.

System action: Processing continues.

System programmer response: Contact the database administrator.

Database Administrator Action: DB2 is a participant with one or more indoubt threads. The coordinator is RRS. RRS has lost knowledge of the indoubt DB2 threads. A list of indoubt threads where RRS is the coordinator can be displayed using the DISPLAY THREAD command for indoubt type threads by specifying RRSAF as the connection name. A DSNV408I message with coordinator = RRSAF identifies a thread of this type. The DSNV449I message in the DISPLAY THREAD report that follows contains the RRS unit of recovery ID (URID) that identifies the RRS unit of recovery in the NID field.

The heuristic decision (that is, to commit or abort the logical unit of work) should be coordinated with any other participant RRS Recoverable Resource Managers. The existence of other participants might not be easy to determine. The information might be available in the MVS/RRS recovery log even though information has been lost.

At this DB2, all indoubt threads coordinated by the RRS must be resolved with the RECOVER INDOUBT command. Locked data remains unavailable until resolution. Threads that were already resolved with this command are discarded. Threads not yet resolved are discarded after resolution with the command.

The commit or abort decision provided using the RECOVER INDOUBT command for a logical unit of work are propagated to all downstream participants, if any.

DSN3012I   *csect-name* **DB2 WAS RESTARTED ON THE WRONG SYSTEM AND CANNOT CONNECT TO RRS. THERE ARE UNRESOLVED UNITS OF RECOVERY WHERE DB2 IS THE COORDINATOR.**

Explanation: DB2 is the commit coordinator for one or more indoubt units of recovery. DB2 is unable to connect to RRS to resolve these indoubt units of recovery.

System action: Processing continues.

Operator response: Use the RRS IPCS interface to determine the correct system and restart DB2 on that system.

DSN3013I   *csect-name* **DB2 WAS RESTARTED ON THE WRONG SYSTEM AND CANNOT CONNECT TO RRS. THERE ARE UNRESOLVED UNITS OF RECOVERY WHERE DB2 IS A PARTICIPANT.**

Explanation: DB2 has one or more indoubt threads and is unable to connect to RRS to resolve these indoubt units of recovery.

System action: Processing continues.

Operator response: Use the RRS IPCS interface to determine the correct system and restart DB2 on that system.

DSN3014I   *csect-name* **DB2 HAS NO KNOWLEDGE OF RRS URID=***rrs_urid*. **THE RRS UR IS INDOUBT. THE DB2 URID RECORDED BY RRS=***db2_urid*.

Explanation: DB2 is restarting with RRS where DB2 is a participant and RRS is the coordinator. RRS has a unit of recovery that DB2 should be a participant in, but DB2 has no knowledge of the RRS unit of recovery. The RRS unit of recovery ID is 'rrs_urid'. RRS has recorded the DB2 unit of recovery ID as 'db2_urid'.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: This message might indicate a DB2 or RRS problem, or it might be produced normally because one of the following prior actions:

- A DB2 conditional restart was performed that resulted in the loss of part or all of the DB2 log. This conditional restart might have happened at any time in the past.
- The DB2 RECOVER INDOUBT command was used to resolve the DB2 unit of recovery with URID 'db2_urid'.

If either of these occurred, the message can be ignored. If neither occurred, there might be a DB2 or an RRS problem. If the 'db2_urid' appears to be a valid log RBA, use the DSN1LOGP utility with the SUMMARY option and URID options using the 'db2-urid' value. If DSN1LOGP finds the 'db2-urid', the disposition will indicate whether the DB2 UR was committed or aborted (rolled back). If possible, use the RRS ISPF interface to commit or abort the RRS URID to avoid heuristic damage.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of DB2 log output.
- DB2 log output, including any DB2 recovery log archive tapes.
- Copy of the log for this period, so that additional DSN1LOGP reports can be produced for problem determination.

.

DSN3015I   *csect-name* **DB2 HAS NO KNOWLEDGE OF RRS URID=***rrs_urid***. THE RRS UR IS INDOUBT. THE DB2 URID RECORDED BY RRS=***db2_urid***.**

Explanation: DB2 is restarting with RRS where DB2 is the coordinator and RRS is the participant. RRS has a unit of recovery that DB2 should be the coordinator of, but DB2 has no knowledge of the RRS unit of recovery. The RRS unit of recovery ID is 'rrs_urid'. RRS has recorded the DB2 unit of recovery ID as 'db2_urid'.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: This message might indicate a DB2 or RRS problem, or it might be produced normally because a DB2 conditional restart was performed that resulted in the loss of part or all of the DB2 log. This conditional restart might have happened at any time in the past.

If the 'db2_urid' appears to be a valid log RBA, use the DSN1LOGP utility with the SUMMARY option and URID options using the 'db2-urid' value. If DSN1LOGP finds the 'db2-urid', the disposition will indicate whether the DB2 UR was committed or aborted (rolled back). Use the RRS ISPF interface to commit or abort the RRS URID to avoid heuristic damage.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of DB2 log output.
- DB2 log output, including any DB2 recovery log archive tapes.
- Copy of the log for this period, so that additional DSN1LOGP reports can be produced for problem determination.

.

DSN3016I   *csect-name* **RRS HAS LOST DATA FROM ITS LOG.**

Explanation: DB2 is restarting with RRS and RRS has lost some portion of its log.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: DB2 might not be able to resolve indoubt Units of Recovery successfully with RRS because of the loss of RRS log data. This message might be followed by one or more of the following messages: DSN3017I, DSN3018I, DSN3019I, DSN3020I, DSN3021I, DSN3022I and DSN3023I.

DSN3017I   *csect-name* **RRS HAS NO KNOWLEDGE OF DB2 URID=**db2_urid**. THE DB2 UR IS INDOUBT. THE RRS URID RECORDED BY DB2=**rrs_urid**.**

Explanation: DB2 is restarting with RRS where DB2 is the coordinator and RRS is a participant. DB2 has a unit of recovery that RRS should be a participant in, but RRS has no knowledge of the DB2 unit of recovery. The DB2 unit of recovery ID is 'db2_urid'. DB2 has recorded the RRS unit of recovery ID as 'rrs_urid'.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: This message might indicate a DB2 or RRS problem, or it might be produced normally because of one of the following:

- An RRS cold start was performed that resulted in the loss of the RRS log. In this case, message DSN3010I should have been produced.
- RRS might have had a partial loss of log data during restart. In this case, message DSN3016I should have been produced.
- The RRS ISPF interface might have been used to heuristically commit or roll back the UR.

If any of the above are true, the message can be ignored. If not, there might be a DB2 or an RRS problem.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of DB2 log output.
- DB2 log output, including any DB2 recovery log archive tapes.
- Copy of the log for this period, so that additional DSN1LOGP reports can be produced for problem determination.
.

DSN3018I   *csect-name* **RRS HAS NO KNOWLEDGE OF DB2 URID=**db2_urid**. THE DB2 UR WAS COMMITTED. THE RRS URID RECORDED BY DB2=**rrs_urid**.**

Explanation: DB2 is restarting with RRS where DB2 is the coordinator and RRS is a participant. DB2 has a unit of recovery that RRS should be a participant in, but RRS has no knowledge of the DB2 unit of recovery. The DB2 unit of recovery ID is 'db2_urid'. DB2 has recorded the RRS unit of recovery ID as 'rrs_urid'.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: This message might indicate a DB2 or RRS problem, or it might be produced normally because of one of the following:

- An RRS cold start was performed that resulted in the loss of the RRS log. In this case, message DSN3010I should have been produced.
- RRS might have had a partial loss of log data during restart. In this case, message DSN3016I should have been produced.
- The RRS ISPF interface might have been used to heuristically commit or roll back the UR.

If any of the above are true, the message can be ignored. If not, there might be a DB2 or an RRS problem.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of DB2 log output.
- DB2 log output, including any DB2 recovery log archive tapes.
- Copy of the log for this period, so that additional DSN1LOGP reports can be produced for problem determination.

.

DSN3019I   *csect-name* **RRS HAS NO KNOWLEDGE OF DB2 URID=***db2_urid***. THE DB2 UR WAS ROLLED BACK. THE RRS URID RECORDED BY DB2=***rrs_urid***.**

Explanation: DB2 is restarting with RRS where DB2 is the coordinator and RRS is a participant. DB2 has a unit of recovery that RRS should be a participant in, but RRS has no knowledge of the DB2 unit of recovery. The DB2 unit of recovery ID is 'db2_urid'. DB2 has recorded the RRS unit of recovery ID as 'rrs_urid'.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: This message might indicate a DB2 or RRS problem, or it might be produced normally because of one of the following:

- An RRS cold start was performed that resulted in the loss of the RRS log. In this case, message DSN3010I should have been produced.
- RRS might have had a partial loss of log data during restart. In this case, message DSN3016I should have been produced.
- The RRS ISPF interface might have been used to heuristically commit or roll back the UR.

If any of the above are true, the message can be ignored. If not, there might be a DB2 or an RRS problem.

Problem determination: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Collect the following diagnostic items:

- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of DB2 log output.
- DB2 log output, including any DB2 recovery log archive tapes.
- Copy of the log for this period, so that additional DSN1LOGP reports can be produced for problem determination.

.

DSN3020I   *csect-name* **POSSIBLE HEURISTIC DAMAGE. RRS HAS COMMITTED THE UR WITH RRS URID=***rrs_urid***. THE DB2 UR WITH DB2 URID=***db2_urid* **IS INDOUBT.**

Explanation: DB2 is restarting with RRS and has a unit of recovery (UR) DB2 URID=db2_urid that is indoubt. DB2 is the commit coordinator for the UR and RRS is a participant. RRS has committed the same UR. The RRS URID=rrs_urid.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: If possible, ensure that the DB2 UR is resolved to commit. If this is done, there will be no heuristic damage. If the DB2 UR is resolved to abort, heuristic damage will have occurred since RRS committed the same UR.

DSN3021I   *csect-name* **POSSIBLE HEURISTIC DAMAGE. RRS HAS ROLLED BACK THE UR WITH RRS. URID=***rrs_urid***. THE DB2 UR WITH DB2 URID=***db2_urid* **IS INDOUBT.**

Explanation: DB2 is restarting with RRS and has a unit of recovery (UR) DB2 URID=db2_urid that is indoubt. DB2 is the commit coordinator for the UR and RRS is a participant. RRS has rolled back the same UR. The RRS URID=rrs_urid.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: If possible, ensure that the DB2 UR is resolved to abort. If this is done, there will be no heuristic damage. If the DB2 UR is resolved to commit, heuristic damage will have occurred since RRS rolled back the same UR.

DSN3022I   *csect-name* **HEURISTIC DAMAGE HAS OCCURRED. RRS HAS COMMITTED THE UR WITH RRS URID=***rrs_urid***, AND DB2 HAS ROLLED BACK THE SAME UR. THE DB2 URID=***db2_urid*

Explanation: DB2 is restarting with RRS and has a unit of recovery (UR) with DB2 URID=db2_urid that was rolled back. DB2 is the commit coordinator for the UR and RRS is a participant. RRS has committed the same UR. The RRS URID=rrs_urid.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: Heuristic damage has occurred since DB2 rolled back the UR and RRS has committed the same UR. Correct the damage. This is not an easy task. Knowledge of the application is required to determine which Recoverable Resource Managers are involved in the UR. Since the time of the heuristic action, the data that was updated might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The RRS ISPF functions can provide assistance in determining the set of resource managers involved and what action they took.
- The DB2 DSN1LOGP utility provides a summary report that identifies the table spaces modified by the unit of recovery.

Contact your IBM Support Center for further assistance.

DSN3023I   *csect-name* **HEURISTIC DAMAGE HAS OCCURRED. RRS HAS ROLLED BACK THE UR WITH RRS URID=***rrs_urid***, AND DB2 HAS COMMITTED THE SAME UR. THE DB2 URID=***db2_urid*

Explanation: DB2 is restarting with RRS and has a unit of recovery (UR) with DB2 URID=db2_urid that was committed. DB2 is the commit coordinator for the UR and RRS is a participant. RRS has rolled back the same UR. The RRS URID=rrs_urid.

System action: Restart with RRS continues.

Operator response: Notify the database administrator.

Database Administrator Action: Heuristic damage has occurred since DB2 committed the UR and RRS has rolled back the same UR. Correct the damage. This is not an easy task. Knowledge of the application is required to determine which Recoverable Resource Managers are involved in the UR. Since the time of the heuristic action, the data that was updated might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The RRS ISPF functions can provide assistance in determining the set of resource managers involved and what action they took.
- The DB2 DSN1LOGP utility provides a summary report that identifies the table spaces modified by the unit of recovery.

Contact your IBM Support Center for further assistance.

DSN3024I	*csect_name* **HEURISTIC DAMAGE HAS OCCURRED. COORDINATOR DB2 HAS COMMITTED THE UR WITH DB2 URID=***db2_urid***, AND RRS HAS INDICATED THAT THE OUTCOME WAS MIXED. THE RRS URID=***rrs_urid***.**

Explanation: DB2 is the commit coordinator and RRS is a participant. DB2 committed the UR with URID = db2_urid and RRS has indicated that one or more Recoverable Resource Managers that participated in the RRS UR with URID = rrs_urid has rolled back the same UR.

System action: Processing continues.

Operator response: Notify the database administrator.

Database Administrator Action: Heuristic damage has occurred since DB2 committed the UR and at least one RRS Recoverable Resource manager has rolled back the same UR. Correct the damage. This is not an easy task. Knowledge of the application is required to determine which Recoverable Resource Managers are involved in the UR. Since the time of the heuristic action, the data that was updated might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The RRS ISPF functions can provide assistance in determining the set of resource managers involved and what action they took.
- The DB2 DSN1LOGP utility provides a summary report that identifies the table spaces modified by the unit of recovery.

Contact your IBM Support Center for further assistance.

DSN3025I	*csect-name* **HEURISTIC DAMAGE HAS OCCURRED. COORDINATOR DB2 HAS ROLLED BACK THE UR WITH DB2 URID=***db2_urid***, AND RRS HAS INDICATED THE OUTCOME WAS MIXED. THE RRS URID=***rrs_urid***.**

Explanation: DB2 is the commit coordinator and RRS is a participant. DB2 rolled back the UR with URID = db2_urid and RRS has indicated that one or more Recoverable Resource Managers that participated in the RRS UR with URID = rrs_urid has committed the same UR.

System action: Processing continues.

Operator response: Notify the database administrator.

Database Administrator Action: Heuristic damage has occurred since DB2 rolled back the UR and at least one RRS Recoverable Resource manager has committed the same UR. Correct the damage. This is not an easy task. Knowledge of the application is required to determine which Recoverable Resource Managers are involved in the UR. Since the time of the heuristic action, the data that was updated might have been used by many applications - read and/or write. Correcting the damage might involve reversing the effects of these applications as well. Tools available are:

- The RRS ISPF functions can provide assistance in determining the set of resource managers involved and what action they took.
- The DB2 DSN1LOGP utility provides a summary report that identifies the table spaces modified by the unit of recovery.

Contact your IBM Support Center for further assistance.

DSN3026I	*csect-name* **POSSIBLE HEURISTIC DAMAGE. COORDINATOR DB2 HAS COMMITTED THE UR WITH DB2 URID=***db2_urid***, AND RRS HAS INDICATED THAT THE OUTCOME IS PENDING. THE RRS URID=***rrs_urid***.**

Explanation: DB2 is the commit coordinator for the DB2 UR with URID=db2_urid and RRS is a participant. RRS has also committed the UR but has indicated that at least one RRS Recoverable Resource manager has not completed commit processing for the RRS UR with URID=rrs_urid.

System action: Processing continues.

Operator response: Notify the database administrator.

Database Administrator Action: If possible, ensure that the RRS Recoverable Resource Manager commits the RRS UR. If this is done, there will be no heuristic damage. If the RRS UR is rolled back by the RRS Recoverable Resource

manager, heuristic damage will have occurred since DB2 committed the same UR.

DSN3027I *csect-name* **POSSIBLE HEURISTIC DAMAGE. COORDINATOR DB2 HAS ROLLED BACK THE UR WITH DB2 URID=***db2_urid***, AND RRS HAS INDICATED THAT THE OUTCOME IS PENDING. THE RRS URID=***rrs_urid***.**

Explanation: DB2 is the commit coordinator for the DB2 UR with URID=db2_urid and RRS is a participant. RRS has also rolled back the UR but has indicated that at least one RRS Recoverable Resource manager has not completed roll back processing for the RRS UR with URID=rrs_urid.

System action: Processing continues.

Operator response: Notify the database administrator.

Database Administrator Action: If possible, ensure that the RRS Recoverable Resource Manager rolls back the RRS UR. If this is done, there will be no heuristic damage. If the RRS UR is committed by the RRS Recoverable Resource manager, heuristic damage will have occurred since DB2 rolled back the same UR.

DSN3028I *csect-name action-1* **CANNOT BE PERFORMED ON A UR BECAUSE THE UR IS IN A GLOBAL TRANSACTION, AND THE STATE OF THE GLOBAL TRANSACTION IS** *action-2***.** *action-2* **IS PERFORMED. URID =** *rba* **CONNECTION NAME =** *connid* **CORRELATION ID =** *corrid* **LUWID =** *logical-unit-of-work-ID* **=** *token*

Explanation: A -RECOVER INDOUBT command was issued for a Unit of Recovery (UR) that was part of a global transaction. The recover action is inconsistent with the state of the global transaction. The requested recover action *action-1* is being changed to *action-2* to preserve the integrity of the global transaction. URID is the beginning RBA of the UR. The combination of Connection Name, Correlation ID, and LUWID identify the thread associated with the UR.

System action: The indoubt Unit of Recovery is resolved to *action-2*.

DSN3029I *csect-name* **RRS ATTACH PROCESSING IS AVAILABLE.**

Explanation: RRSAF has finished synchronizing with z/OS RRS and is now available to process application calls.

System action: Processing continues.

Operator response: Applications that require RRSAF services can be started.

DSN3030I *csect-name* **RRS ATTACH WAITING FOR Z/OS RRS INITIALIZATION, RRS CALL=***rrs-function***, RRS RETURN CODE=X'***rrs-return***'**

Explanation: The DB2 Resource Recovery Services Attachment Facility (RRSAF) could not establish a connection to z/OS RRS because RRS is not available.

csect-name
> The name of the control section that issued the message.

rrs-function
> The RRS request that is failing.

rrs-return
> The z/OS RRS return code issued by the failing request.

System action: RRSAF functions, stored procedures (except for native SQL procedures), and user-defined functions are not available on this DB2 subsystem.

Operator response: Start z/OS RRS. When z/OS RRS is started, DB2 RRSAF should automatically connect and resync with z/OS RRS, and RRSAF functions, stored procedures, and user defined functions become available.

If it will not start, notify the system programmer, and provide the information in this message.

System programmer response: Use the z/OS RRS return code provided in the message to determine why RRS has not started.

Related reference:

⇨ z/OS MVS Programming: Resource Recovery

DSN3031I   *csect-name* **RRS ATTACH INITIALIZATION FAILED, RRS CALL=***rrs-function***, RRS RETURN CODE=X'***rrs-return***'**

Explanation: The DB2 Resource Recovery Services Attachment Facility (RRSAF) could not establish a connection to z/OS RRS because an error code was returned by an RRS function.

csect-name
> The name of the control section that issued the message.

rrsaf-function
> The RRS request that is failing.

rrs-return
> The z/OS RRS return code issued by the failing request.

System action: A record is written to SYS1.LOGREC, and an SVC dump is generated for DB2 reason code 00F30095.

RRSAF functions, stored procedures (except for native SQL procedures), and user-defined functions are not available on this DB2 subsystem.

Operator response: Notify the system programmer, and provide the information in this message.

System programmer response: Use the z/OS RRS return code provided in the message to determine why RRS has not started. If the problem persists, contact IBM Software Support.

When z/OS RRS is started, DB2 RRSAF should automatically connect and resync with z/OS RRS, and RRSAF functions, stored procedures, and user defined functions become available.

Related reference:

⇨ z/OS MVS Programming: Resource Recovery

Related information:

⇨ 00F30095 (DB2 Codes)

DSN3100I   *src csect-name* **- SUBSYSTEM** *ssnm* **READY FOR START COMMAND**

Explanation: This message indicates that a DB2 subsystem is ready to be started.

src   The DB2 command prefix.

csect-name
> The name of the control section that issued the message.

ssnm   The DB2 subsystem name.

This message occurs during MVS IPL processing. Message DSN3104I is also issued during this processing.

If AUTOSTART=YES is specified in DSNZPARM and this message is used as a trigger for automation to restart the DB2, a DELAY must be programmed into the automation to allow time for the IRLM to process the termination, and issue the STOP IRLM. Another approach would be to monitor for both the DB2 termination and the IRLM termination in combination, before restarting DB2. If neither of these actions are done, DB2 could identify to the existing IRLM and will NOT STOP automatically from that point onward.

Operator response: Issue the -START command when you are ready to start DB2.

DSN3101I   *csect-name* **INVALID EARLY PROCESSING MODULE NAME IN PARMLIB RECORD.** *ssnm* **IS NOT AVAILABLE**

Explanation: CSECT DSN3UR00 has discovered that a parameter passed to DB2 subsystem initialization is in error. Either the Early processing load module name is not between 1 and 8 characters long, or there are syntax errors in the parameter string. The entire parameter string must be between 3 and 10 characters long, and the next to the last

character must be a ','. The parameters were obtained from the parameter fields of a record defining a DB2 subsystem in member IEFSSNxx of SYS1.PARMLIB.

The subsystem recognition character is not inserted in this message as it is not yet known. 'ssnm' is the name of the DB2 subsystem undergoing IPL-time initialization.

System action: The DB2 subsystem with the indicated name is not available for this IPL of MVS.

Operator response: Notify the system programmer.

System programmer response: Correct the parameter fields in the record of SYS1.PARMLIB member IEFSSNxx, and re-IPL MVS.

Related tasks:

▶ Registering the command prefixes, member group attachment name, and subgroup attachment name (DB2 Installation and Migration)

Related reference:

▶ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3102I　*csect-name* **INVALID SUBSYSTEM RECOGNITION CHARACTER IN PARMLIB RECORD.** *ssnm* **IS NOT AVAILABLE**

Explanation: CSECT DSN3UR00 detected that the subsystem recognition character is invalid; it must be nonblank. The subsystem recognition character is the last field of the parameter string passed to DB2 IPL initialization. It is obtained from a record for the named subsystem in member IEFSSNxx of SYS1.PARMLIB.

The subsystem recognition character is not inserted in this message as it cannot be determined. 'ssnm' is the name of the DB2 subsystem undergoing IPL-time initialization.

System action: The DB2 subsystem with the indicated name is not available for this IPL of MVS.

Operator response: Notify the system programmer. Correct the parameter fields in the SYS1.PARMLIB record of member IEFSSNxx, and re-IPL MVS.

Related tasks:

▶ Registering the command prefixes, member group attachment name, and subgroup attachment name (DB2 Installation and Migration)

Related reference:

▶ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3103I　*csect-name* **- UNABLE TO LOAD EARLY PROCESSING MODULE SPECIFIED IN PARMLIB RECORD.** *ssnm* **IS NOT AVAILABLE**

Explanation: CSECT DSN3UR00 is unable to load the early processing load module named in the first parameter passed to DB2 subsystem IPL initialization. 'ssnm' is the name of the DB2 subsystem undergoing IPL-time initialization.

Either the load module name was specified incorrectly or the load module does not reside in a library included in the MVS linklist.

System action: DSN3UR00 abends with code X'04E' and reason code X'00F30103'. The DB2 subsystem with the indicated name is not available for this IPL of MVS.

Operator response: Notify the system programmer.

System programmer response: Identify the cause of the error. Either correct the early processing load module or take the necessary steps to include the module in a link-listed library. Re-IPL MVS. Refer to the appropriate MVS publication for MVS linklist requirements.

Problem determination: An entry is made to SYS1.LOGREC. No SVC dump is taken.

DSN3104I • DSN3107I

Related tasks:

➦ Registering the command prefixes, member group attachment name, and subgroup attachment name (DB2 Installation and Migration)

Related reference:

➦ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3104I   *csect-name* **- TERMINATION COMPLETE**

Explanation: This message indicates that a DB2 subsystem has terminated. The actual MVS termination of the DB2 address space(s) may have completed earlier. This message is presented for every termination, normal or abnormal.

Operator response: Issue a DISPLAY ACTIVE command to see if MVS has completed termination processing of the address space(s).

DSN3105I   *csect-name* **- UNABLE TO LOAD EARLY PROCESSING MODULE 'DSN3EPX'** *ssnm* **IS NOT AVAILABLE**

Explanation: CSECT DSN3UR00 is unable to load the early processing module DSN3EPX required for MVS. The name of the DB2 subsystem undergoing IPL-time initialization is 'ssnm'. Either the load module has been inadvertently deleted, or does not reside in a library included in the MVS linklist.

System action: DSN3UR00 abends with code X'04E' and reason code X'00F30105'. DB2 subsystem 'ssnm' is not available for this IPL of MVS.

Operator response: Notify the system programmer.

System programmer response: Identify the cause of the error. Either correct the early processing load module or take the necessary steps to include the module in a link-listed library. Re-IPL MVS.

If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Problem determination: An entry is made in SYS1.LOGREC. No SVC dump is taken. Refer to MVS linklist requirements in *MVS/ESA JES2 Initialization and Tuning Guide*.

Collect the following diagnostic items:
- Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of time spanning the failure.
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSN3106I   *csect-name* **SUBSYSTEM STOPPED. THIS COMMAND NOT PROCESSED:** *command-text*

Explanation: A command has been received which cannot be processed due to one of the following:
- The DB2 subsystem has not been started.
- The command was queued for processing while DB2 was starting, but startup terminated with an error.
- DB2 terminated before the command could be processed.

System action: The command is not processed.

Operator response: Start the DB2 subsystem, then reenter the command.

DSN3107I   *csect-name* **COMMAND REJECTED. REQUESTER NOT AUTHORIZED**

Explanation: CSECT DSN3EC0X has discovered that the source of the command was a console that does not have 'system' authority.

System action: The command is not processed. This message is sent to the console that entered the command.

Operator response: Enter the command from another console that has 'system' authority.

System programmer response: Verify that this console should be used for entering DB2 commands. If so, authorize it for 'system' commands by using MVS services.

DSN3108I   *csect-name* **COMMAND REJECTED. COMMAND FACILITY PATH UNAVAILABLE**

Explanation: CSECT DSN3EC0X has discovered that the path from MVS consoles to the DB2 command processor is unavailable. It may still be possible to enter commands through one of the attachment facilities.

System action: The command is not processed. This message is delivered to the console that entered the command.

System programmer response: The console command facility is available again the next time DB2 is started. If it is necessary to regain this facility immediately, DB2 should be canceled and restarted.

DSN3109I   *csect-name* **- UNABLE TO OBTAIN SUBSYSTEM AFFINITY TABLE INDEX FOR SUBSYSTEM** *ssnm*. **IEFSSREQ RC =** *nn*

Explanation: CSECT DSN3UR00 was unable to obtain a subsystem affinity table index for the named subsystem. MVS did not recognize the named subsystem name as a known subsystem. If this message is issued, a serious error has occurred in MVS or DB2.

In the message, 'nn' is the return code from the IEFSSREQ MVS service. 'ssnm' is the name of the DB2 subsystem undergoing IPL-time initialization.

System action: DSN3UR00 abends with completion code X'04E' and reason code X'00F30104'. The DB2 subsystem with the indicated name is not available for this IPL of MVS.

Operator response: Notify the system programmer.

System programmer response: Try to re-IPL. If the problem persists, follow standard DB2 problem reporting procedures.

Problem determination: A record is written to SYS1.LOGREC. No SVC dump is taken. Return codes from IEFSSREQ are documented under control block SSOB in the appropriate MVS publication.

DSN3111I   *csect-name* **LOAD MODULE** *module-name* **DOES NOT HAVE AMODE(31) ATTRIBUTE.** *ssnm* **IS NOT AVAILABLE**

Explanation: The DB2 program which establishes DB2 as an MVS subsystem during Master Scheduler initialization has determined that either its own load module or a load module loaded by it does not have the linkage editor attributes AMODE(31) and RMODE(ANY). Most DB2 load modules must be capable of running in the 31-bit addressing mode.

In the message, 'csect-name' is the name of the program detecting the error, 'module-name' is the name of the first load module found with an invalid AMODE attribute, and 'ssnm' is the name of the DB2 subsystem undergoing IPL-time initialization.

System action: Program DSN3UR00 abends with a code of X'04E' and a reason code of X'00F30111'. The DB2 subsystem named 'ssnm' is not initialized. The subsystem will not be available until the error is corrected and MVS is re-IPLed.

Operator response: Notify the system programmer of the failure.

System programmer response: Verify that all of the following were done:
- The installation and maintenance activities against DB2 were executed using SMP.
- The JCLIN for SMP included the correct AMODE and RMODE control statements.
- The appropriate Linkage Editor provided with the Data Facility Product (DFP) for the version of MVS being used was invoked by SMP for the link-edit steps.

Correct and rerun the procedure or JCL which caused the error. If necessary, REJECT all SMP APPLY steps which were executed incorrectly and rerun the rejected steps. Request a re-IPL of MVS.

Problem determination: A record is written to SYS1.LOGREC. No SVC dump is taken.

The MVS service aid AMBLIST provides Linkage Editor module attributes in the summary section of the LISTLOAD function output.

DSN3112I   *csect-name* **UNABLE TO REGISTER COMMAND PREFIX, CPF RETURN CODE =** *xxyy*. *ssnm* **IS NOT AVAILABLE**

Explanation: DB2 is unable to register the installation-specified command prefix with MVS. The command prefix is specified in the subsystem definition parameter in SYS1.PARMLIB(IEFSSN*nn*). *xx* is the hexadecimal return code, and *yy* is the hexadecimal reason code from the command prefix facility (CPF) of MVS.

System action: The DB2 subsystem, *ssnm*, is not available.

System programmer response: Use the information listed for REQUEST=DEFINE in CPF macro return and reason codes (MVS Programming Assembler Services Reference) to identify the cause of the error and determine what to do. Find the specific combination of return code (*xx*) and reason code (*yy*) that identifies the meaning of the error and the suggested action.

Ensure that the specified command prefix is not already registered, or is not a substring or superstring of an already registered command prefix. Use the MVS DISPLAY OPDATA command, if necessary, to see command prefixes registered with MVS.

Related reference:

⮕ z/OS OPDATA command (MVS System Commands)

DSN3113I   *csect-name* **INVALID SUBSYSTEM DEFINITION PARAMETER IN PARMLIB RECORD.** *ssnm* **IS NOT AVAILABLE.**

Explanation: DB2 detected an error in the subsystem definition parameter, which is obtained from the IEFSSNxx member of SYS1.PARMLIB. Some possible causes for this error are:

• The early processing load module name is not between 1 and 8 characters long.
• The command prefix is not between 1 and 8 characters long.
• The command prefix scope is not 1 character long.
• The command prefix scope is not X, S, or M.
• The group attach name is not between 1 and 4 characters long.

The command prefix is not inserted into this message because it is unknown. The name of the subsystem attempting to initialize is *ssnm*.

System action: The DB2 subsystem is not available.

System programmer response: Correct the subsystem definition parameter and re-IPL MVS.

Related tasks:

⮕ Registering the command prefixes, member group attachment name, and subgroup attachment name (DB2 Installation and Migration)

Related reference:

⮕ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3114I   *csect-name* **GROUP ATTACH NAME IS NOT ALLOWED WITH THIS RELEASE OF MVS.** *ssnm* **IS NOT AVAILABLE**

Explanation: DB2 found an installation-specified group attachment name specified in the subsystem definition parameter, IEFSSNxx, but DB2 group attachment is not supported with this release of MVS. DB2's group attachment feature requires MVS/ESA SP4.2.2 or a later release. The DB2 subsystem *ssnm* is not available.

System programmer response: Delete the group attachment name in the subsystem definition parameter and re-IPL.

DSN3115I   *csect-name* **UNABLE TO REGISTER GROUP ATTACH NAME, NAME/TOKEN RC =** *xx*. *ssnm* **IS NOT AVAILABLE**

Explanation: DB2 is unable to register the installation-specified group attach name with MVS. The group attach name is specified in the subsystem definition parameter. *xx* is the hexadecimal return code from the MVS Name/Token service. Subsystem *ssnm* is not available.

System action: This DB2 subsystem is not available.

System programmer response: Refer to MVS documentation about the Name/Token macro service to determine the reason for the *xx* return codes.

DSN3116I   *csect-name* **THE VERSION OF THE DB2 EARLY CODE IS INCOMPATIBLE WITH THIS VERSION OF MVS.** *ssnm* **IS NOT AVAILABLE**

Explanation: This version of DB2 early code requires MVS/ESA 4.2.0. Subsystem *ssnm* is not available.

System programmer response: The version of the DB2 early code that was loaded from the LINKLIST library is not compatible with the release of MVS that you are using. Change the concatenation order of the LINKLIST library so that a compatible release of DB2 early code is used.

DSN3117I   *xxxx* **MAINTENANCE LEVELS**

Explanation: The -REFRESH DB2, EARLY command completed successfully.

DSN3118I   *csect-name* **DSN3UR00 - GROUP ATTACH NAME** *group-name* **WAS ALREADY DEFINED AS A SUBGROUP ATTACH**

Explanation: The IEFSSN*xx* subsystem parameter specifies a group name that has already been used as a subgroup name.

csect-name
>   The name of the control section that issued the message.

group-name
>   The group attachment name.

System action: The DB2 subsystem is not available.

System programmer response: Change the subgroup name in the subsystem parameter, and re-IPL z/OS.

Related concepts:

➡ Group attachments and subgroup attachments (DB2 Data Sharing Planning and Administration)

Related reference:

➡ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3119I   *csect-name* **DSN3UR00 - SUBGROUP ATTACH** *subgroup-name* **DOES NOT BELONG TO GROUP ATTACH** *group-name*

Explanation: The IEFSSN*xx* subsystem parameter specifies a subgroup attachment name that does not belong to the specified group.

csect-name
>   The name of the control section that issued the message.

subgroup-name
>   The subgroup attachment name.

group-name
>   The group attachment name.

A subgroup attachment name can belong to only one group.

System action: The DB2 subsystem is not available.

System programmer response: Change the subgroup name in the subsystem parameter, and re-IPL z/OS.

Related concepts:

⬕ Group attachments and subgroup attachments (DB2 Data Sharing Planning and Administration)

Related reference:

⬕ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3120I   *csect-name* **DSN3UR00 -** *group-name* **CANNOT BE DEFINED AS BOTH THE GROUP ATTACH NAME AND THE SUBGROUP ATTACH NAME**

Explanation: The same name was specified for the group and subgroup attachment names. In compliance with subgroup attachment rules, these names must be unique.

csect-name
         The name of the control section that issued the message.

group-name
         The group attachment name.

System action: The DB2 subsystem is not available.

System programmer response: Change either the group or the subgroup name, and re-IPL z/OS.

Related concepts:

⬕ Group attachments and subgroup attachments (DB2 Data Sharing Planning and Administration)

Related reference:

⬕ Define group or member panel: DSNTIPK (DB2 Installation and Migration)

DSN3121I   *csect-name* **DSN3UR00 - SUBGROUP ATTACH** *subgroup-name* **WAS ALREADY DEFINED AS A GROUP ATTACH**

Explanation: The IEFSSN*xx* subsystem parameter specifies a subgroup name that has already been used as a group name.

csect-name
         The name of the control section that issued the message.

subgroup-name
         The subgroup attachment name.

System action: The DB2 subsystem is not available.

System programmer response: Change either the group name or the subgroup name, and re-IPL z/OS.

Related concepts:

⬕ Group attachments and subgroup attachments (DB2 Data Sharing Planning and Administration)

Related reference:

⬕ MVS PARMLIB updates panel: DSNTIPM (DB2 Installation and Migration)

DSN3201I   **ABNORMAL EOT IN PROGRESS FOR USER=***xxxxxxxx* **CONNECTION-ID=***yyyyyyyy*
                 CORRELATION-ID=*zzzzzzzzzzzzz* **JOBNAME=***aaaaaaaa* **ASID=***9999* **TCB=***99999999*

Explanation: Abnormal termination started for the agent with the values shown for USER, CONNECTION-ID, CORRELATION-ID, JOBNAME, ASID and TCB address. These values are the last known set of identifiers for the terminating agent.

The abnormal termination could be the result of a failure in the allied agent's address space or the result of an operator cancel command.

The value for the USER and/or CORRELATION-ID might be blank. If the terminating agent is connected to DB2 at the IDENTIFY level and was not previously signed on, or was not previously authorized to a plan at the time of the failure, then no CORRELATION-ID is associated with the agent. Otherwise, the values for the USER, CONNECTION-ID, CORRELATION-ID, JOBNAME and ASID are the last values established to DB2 for this connection. They might represent the current activity of the agent. The TCB value is the address of the TCB that is

terminating. Previous DB2 work by this agent might have completed successfully. Correlation IDs beginning with '0nn', where nn is a number from 01 to 28, identify system agents.

System action: The agent was previously queued to a System Services service task for termination processing. This message indicates that the agent was taken from the queue for processing. Any uncommitted database changes are rolled back.

System programmer response: See the Problem Determination section for this message. The MVS CANCEL and FORCE commands have no effect and should not be issued. DB2 should not be canceled. If an extensive rollback is in progress, the subsequent DB2 restart might take a very long time because of additional log activity.

User response: Your job or TSO session cannot be canceled. The PA1 key has no effect in this circumstance. To avoid long waits when your application program is canceled or terminates abnormally, issue frequent commits.

Problem determination: You can detect a deferred termination condition for a task by examining several indicators. Some or all of the following might be present:

- The allied address space might be swapped out and appear to be in a never-ending WAIT condition.
- The MVS commands CANCEL and FORCE appear to have no effect.
- The allied task holds an MVS-shared ENQ on resource SYSZDSN3.ERLYOLRHerly-block-address.
- During abnormal termination of the agent associated with the failing task, but before this message was written, the task's connection appears on the -DISPLAY THREAD output with a QD status. See message DSNV401I for the definition of this status code.
- This message, DSN3201I, is written to the MVS console after the agent is removed from the service task work queue when termination processing begins.
- During abnormal termination of the agent associated with the failing task, the task's connection appears on the -DISPLAY THREAD output with a D status. See message DSNV401I for the definition of this status code.

DSN3458I   *csect-name* **MODULE** *modname* **MUST HAVE THE AMODE (31) ATTRIBUTE**

Explanation: An authorization exit load module was not link-edited with the 31-bit addressing mode (AMODE) attribute. Either load module DSN3@ATH or DSN3@SGN is specified by 'modname'.

System action: Subsystem startup is terminated.

System programmer response: Relink-edit the authorization exit module specifying the attributes AMODE(31), RMODE(ANY).

Problem determination: Check the load module attributes used for the link-edit.

DSN3580I   **CONNECTION FOR [EOM | FEOT | IDEN] GAVE RC=X xxxxxxxx, REASON=X xxxxxxxx**

Explanation: A nonzero return code has been returned to DSN3AMI2 from the SSSS connect to subsystem interface call. The variables in the message indicate which SSI call is involved and the actual return and reason codes associated with it.

System action: The current task (TCB) is abended with a system completion code of X'04E' and with an abend reason code of '00F30580' The DB2 subsystem terminates.

Operator response: Notify the system programmer.

System programmer response: Restart the DB2 subsystem.

Problem determination: The trace table and stack storage associated with the EB in register 6 must be examined to locate the error.

Collect the following diagnostic items:

- Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
- Listing of SYS1.LOGREC data set, obtained by executing IFCEREP1.

DSN3581I   *csect-name* **SAF** *service* **FAILED WITH SAF RETURN CODE =** *saf-retc*, **RACF RETURN CODE =** *racf-retc*, **AND RACF REASON CODE =** *racf-reac*. **DB2 REASON CODE =** *db2-reac*

Explanation: DB2 invoked the specified SAF (Security Authorization Facility) service which failed with the following codes:

Variable
> **Definition**

csect   The CSECT name of the DB2 module that issued the message.

service   The name of the service that failed.

saf-retc   The SAF return code.

racf-retc
> The RACF return code.

db2-retc
> The DB2-assigned reason code for the failure.

> DB2 associates the failure with a DB2 reason code which may appear in other messages associated with this failure.

System action: The call fails. Other messages associated with this feature may follow.

Operator response: Notify the system programmer.

System programmer response: The action to take depends on the service being used.

- R_ticketserv

  DB2 requests this service to parse a Kerberos ticket received from a remote client. Refer to OS/390 Security Server (RACF) Callable services for an explanation of the error.

  DB2 uses the SAF R_ticketserv service to parse a Kerberos ticket that was received from a remote client. Refer to *OS/390 SecureWay Security Server RACF Callable Services* for an explanation of the error. A SAF and RACF return code of 8 and 16 indicate a failure from the Network Authentication Privacy Service (Kerberos) when parsing the ticket in which case the RACF reason code contains a status code from the Network Authentication Privacy Service. These status codes are documented in the *OS/390 SecureWaySecurity Server Network Authentication and Privacy Service Administration* manual. Additional information for some of the more common status codes and how they relate to DB2 follows:

  96C73A07
  > One possibility is that you must associate the ssnmDIST started task user ID with a principal name and the client system must use this principal name in the Kerberos target principal.

  > **Note:** DB2 Connect™ client systems must catalog a DB2 for OS/390 database with "AUTHENTICATION KERBEROS TARGET PRINCIPAL principal_name". Assume that SYSDSP is the user ID associated with the ssnmDIST started task. A principal, say SYSDSPp, must be associated to the SYSDSP user ID by a RACF ALTUSER command. Further, the SYSDSPp principal name must appear in the target principal name of the DB2 Connect database configuration for the DB2 for OS/390 database.

- R_ticketserv(principal name)

  DB2 uses the SAF R_usermap service to determine the user ID associated with a Kerberos ticket received from a remote client. Refer to *OS/390 SecureWay Security Server RACF Callable Services* for an explanation of the error. The principal name obtained from the ticket is included in parenthesis to assist the user in determining the cause of the problem. Either the client system is using an invalid principal name, or a principal name to user ID association has not been defined to RACF.

  Principals authenticated on the local OS/390 realm must have a mapping that associates the local principal name to a local RACF user ID. This association can be made by using the RACF ADDUSER or ALTUSER commands with the KERBNAME parameter.

  Principals authenticated on a foreign realm must have a mapping that associates the foreign fully qualified principal name to a local RACF user ID. This association can be made by using the RACF RDEFINE or RALTER commands with the KERBLINK parameter.

- R_usermap

  DB2 requests this service to determine the user ID associated with a Kerberos ticket received from a remote client. Refer to *OS/390 Security Server (RACF) Callable services* for an explanation of the error.

User response: Notify the system programmer.

Chapter 24. DSN5 messages

Message identifiers that begin with DSN5 communicate information about DB2.

DSN5001I   ERROR OCCURRED WHILE CONTACTING OMNIFIND SERVER *server* FOR *function*. ON
COLLECTION *collection*.

Explanation: A failure that requires action from the system administrator occurred while contacting an OmniFind®
server.

System action: If the function is SYSPROC.SYSTS_DROP, the information in DB2 for the collection is deleted. If the
function is SYSPROC.SYSTS_UPDATE, takeover of the collection by another IBM OmniFind server of
SYSIBMTS.SYSTEXTSERVERS is attempted.

Programmer response: If the function is SYSPROC.SYSTS_DROP, then *collection* must be deleted at the OF server
using OF administration APIs. If the function is SEARCH (Contains or Score), check if the IBM OmniFind server has
a persistent problem and call SYSPROC.SYSTS_TAKEOVER. If the function is SYSPROC.SYSTS_TAKEOVER, then
takeover was not possible or failed; therefore check the IBM OmniFind server. If takeover was not possible because
no additional servers are registered in SYSIBMTS. SYSTEXTSERVERS (message parameter *server* empty), add an
additional server and check and correct the problem on the existing server or servers.

Chapter 25. DSN7 messages

Message identifiers that begin with DSN7 communicate information about the group manager.

DSN7000I   *csect-name* **REQUIRED CSECT DSN6GRP COULD NOT BE FOUND IN SYSTEM PARAMETERS LOAD MODULE.**

Explanation: The required CSECT DSN6GRP was not found in the system parameters load module.

System action: The DB2 subsystem terminates with reason code 00F70001.

Operator response: Restart DB2 with the correct system parameters load module.

System programmer response: Reassemble the system parameters load module and include the DSN6GRP macro.

DSN7003I   *csect-name* **SCA DELETED BECAUSE OF SUBSYSTEM PARAMETER SETTING**

Explanation: The DB2 shared communications area (SCA) has been deleted because the DEL_CFSTRUCT_ON_RESTART subsystem parameter indicated that DB2 restart should attempt to delete it.

csect-name
> The name of the control section that issued the message.

System action: The SCA is deleted and rebuilt.

User response: No action is required.

Related reference:

➥ DEL CF STRUCTS field (DEL_CFSTRUCTS_ON_RESTART subsystem parameter) (DB2 Installation and Migration)

| **DSN7100I**   *csect-name*

| **Explanation:** This message displays output from the DISPLAY GROUP command.

| *csect-name*
| > The name of the control section that issued the message.

| The output from the command consists of one or more of the following sections, in the indicated order:
| • Output heading
| • Data-sharing members
| • New-function mode enablement
| • Subgroup attachment
| • Shared communications area
| • LOCK1 structure
| • SPT01 table space information
| • End of output

| **Output heading:** The name of the control section is followed by heading information for the output:

| **GROUP (***text***)**
| > The name of the data-sharing group.

| **CATALOG LEVEL (***text***)**
| > A string of three numeric characters that indicates the highest release with which any DB2 subsystem in the group is started:
| > • Version
| > • Release

DSN7100I

- Modification level

MODE (*text***)**
    A 2- or 3-character indicator of the current mode of the system:

C*n*   Conversion mode. All members in the group are on the same DB2 release.

C*n*****   Conversion mode. The * indicates that at one time the data-sharing group was in enabling-new-function mode, enabling-new-function mode*, or new-function mode.

E*n*   Enabling-new-function mode.

E*n*****   Enabling-new-function mode. The * indicates that at one time the data-sharing group was in new-function mode.

N   New-function mode.

X   Coexistence mode. Members in the data-sharing group are on different DB2 releases, or a member has performed a fallback from a higher release.

PROTOCOL LEVEL (*num***)**
    A single numeric character that indicates the current locking protocol that is used in the data-sharing group.

GROUP ATTACH NAME (*text***)**
    The group attach name for the data-sharing group.

Data-sharing members: The output heading is followed by a section that contains information about the data-sharing members:

DB2 MEMBER *text*
    The name of the DB2 data-sharing member.

ID *num*
    The internally assigned ID for each data-sharing member.

SUBSYS *text*
    The subsystem name of the data-sharing member.

CMDPREF *text*
    The command prefix of the data-sharing member.

STATUS *text*
    The state of the data-sharing member. Expected values are:

    ACTIVE
        The member is up and running.

    AI   The member is active with more conditions. Indoubt or postponed abort units of recovery are unresolved, so retained locks are held.

    DEACT
        The member is deactivated.

    FAILED
        The member terminated abnormally.

    QUIESCED
        The member is down normally.

    QI   The member is down with indoubt units of recovery.

    QR   The member is down with resynchronization responsibility.

    QC   The member is down with group buffer pool castout responsibility.

    QCR, QIC, QIR, or QICR
        Possible combinations of the quiesced states (QI, QR, and QC):
- QCR is the equivalent of QC and QR.
- QIC is the equivalent of QI and QC.
- QIR is the equivalent of QI and QR.
- QICR is the equivalent of QC, QI, and QR.

| **DB2 LVL** *num*

A string of three numeric characters that identifies the DB2 product version:

- DB2 version
- DB2 release
- DB2 modification level

| **SYSTEM NAME** *text*

The name of the z/OS system on which the data-sharing member runs.

| **IRLM SUBSYS** *text*

The accompanied IRLM's subsystem name for this DB2 member.

| **IRLMPROC** *text*

The accompanied IRLM's startup procedure name.

| **New-function mode enablement:** If the DISPLAY GROUP command is issued with the DETAIL option, and the DB2 catalog is in enabling-new-function mode (indicated by either E*n* or E*n** in the output heading), the following additional information is provided:

| **TABLE SPACE** *text*

The name of the table space.

| **ENABLED FOR NEW FUNCTION** *text*

An indicator of the readiness of the table space for new-function mode. Expected values are:

| **YES**   The table space was created in Unicode or it has been converted to Unicode.

| **NO**   The table space is not enabled for new-function mode. Table spaces are converted to Unicode during enabling-new-function mode.

| **Subgroup attachment:** If the DISPLAY GROUP command is issued with the DETAIL option, subgroup attachment information is provided:

| **GROUP ATTACH** *text*

The name of the group attachment group for the data-sharing member that is printing the output.

| **MEMBERS OF SUBGROUP ATTACH** *text*

The name of the subgroup attachment group, followed by one or more lines that list the members of the subgroup. Each line contains up to eight members.

If more than 105 subgroup attachment groups are found, the following message is displayed:

```
MAXIMUM NUMBER OF SUBGROUP ATTACH GROUPS
DISPLAYED
```

In this case, no additional groups are reported.

| **Shared communications area:** The output provides the following information about the shared communications area (SCA):

| **STRUCTURE SIZE** *num*

The size of the shared communications area (SCA), measured in kilobytes (KB).

| **STATUS** *tx*

A 2-character indicator of the status of the shared communications area. Possible values are:

| **AC**   The SCA is active.

| **CC**   The SCA rebuild connect function completed.

| **CS**   The SCA rebuild connect function started.

| **DS**   The SCA loss of connectivity, delayed action started.

| **DC**   The SCA loss of connectivity, delayed action completed.

| **EC**   The SCA rebuild stop complete function completed.

| **ES**   The SCA rebuild stop complete function started.

| **IN**   The SCA is in initialization phase.

DSN7100I

	LC	The SCA rebuild cleanup function completed.
	LS	The SCA rebuild cleanup function started.
	OC	The SCA rebuild connect complete function completed.
	OS	The SCA rebuild connect complete function started.
	PC	The SCA rebuild process complete function completed.
	PS	The SCA rebuild process complete function started.
	QC	The SCA rebuild quiesce function completed.
	QS	The SCA rebuild quiesce function started.
	RC	The SCA rebuild resource manager invocation completed.
	RS	The SCA rebuild resource manager invocation started.
	SC	The SCA rebuild stop function completed.
	SS	The SCA rebuild stop function started.
	TE	The SCA is in termination phase.
	UC	The SCA rebuild resource manager invocation completed.
	US	The SCA rebuild resource manager invocation started.
	VC	The SCA volatility change function completed.
	VS	The SCA volatility change function started.

SCA IN USE *num*
> Either the percentage of the number of entries that are in use in the SCA or the percentage of the number of elements that are in use in the SCA, whichever is larger. If the SCA structure is very large or if only a small number of SCA entries are in use, the percentage that is in use might be less than 1%. In this case, the percentage is indicated as:

```
SCA IN USE: < num %
```

LOCK1 structure: The output provides the following information about the LOCK1 structure:

STRUCTURE SIZE *num*
> The size of the LOCK1 coupling facility structure, measured in kilobytes (KB).

LOCK ENTRIES *num*
> The maximum number of lock table entries in the LOCK1 coupling facility structure.

LIST ENTRIES *num*
> The maximum number of modify lock list entries in the LOCK1 coupling facility structure.

LIST ENTRIES IN USE *num*
> The number of modify lock list entries in the LOCK1 coupling facility structure that are currently in use. These entries are sometimes referred to as *record data*.

SPT01 table space information: The output contains information about the size of the inline length of the SPT01 column SPTSEC_DATA:

SPT01 INLINE LENGTH *num*
> The current size of the inline length of the SPTSEC_DATA column.

End of output: The output normally ends with the following message:

```
*** END DISPLAY OF GROUP(text)
```

GROUP
> The name of the group that the preceding lines of the output refer to.

If the output exceeds the maximum length, the following message marks the end of the output instead:

```
DISPLAY TERMINATED DUE TO INSUFFICIENT MESSAGE
    SPACE.
```

System action: Processing continues.

| **User response:** No action is required.

| **Related reference**:

| ⬈ -DISPLAY GROUP (DB2) (DB2 Commands)

DSN7114I   **XCF OR STRUCTURE INFORMATION NOT AVAILABLE. MVS** *nnnnnnnn* **RETURN CODE=***xxxxxxxx***, MVS** *nnnnnnnn* **REASON CODE=***yyyyyyyy***.**

Explanation: An error occurred during the query of the cross-system coupling facility (XCF) or CF structure information.

nnnnnnnn could be IXCQUERY or IXLMG.

System action: The DB2 subsystem displays the group, but some XCF or coupling facility structure information is missing. Refer to *MVS/ESA Programming: Sysplex Services Reference* for an explanation of IXCQUERY or IXLMG return and reason codes.

DSN7401I   *csect-name* **INCONSISTENT DATA SHARING PARAMETER DETECTED,** *parameter-name* **IN DB2 SYSTEM PARAMETERS (***sysparm-parm***) IS NOT EQUAL TO PARAMETER IN THE DB2 START-UP PROCEDURE (***start-parm***).**

Explanation: At least one data sharing parameter in the member's system parameters is not equal to the data sharing parameter value passed on the DB2 start-up procedure.

System action: The DB2 subsystem terminates with reason code 00F70003.

Operator response: Restart DB2 with the correct system parameters load module or inform the system programmer to change the DB2 start-up procedure.

System programmer response: Reassemble the system parameters load module. Make sure that the group name and the member name match the values in the DB2 start-up procedure, or correct the parameters on the DB2 start-up procedure.

DSN7403I   *csect-name* **MEMBER(***member-name***) IS ALREADY ACTIVE ON SYSTEM(***mvs-system-name***).**

Explanation: The started DB2 subsystem is already active on another MVS system.

System action: The DB2 subsystem terminates with reason code 00F70003.

Operator response: If you want the DB2 subsystem to run on this MVS system, stop the DB2 subsystem on the other MVS. After the DB2 subsystem stops, restart the DB2 on this MVS.

DSN7404I   *csect-name* **INVALID MEMBER(***member-name***) WAS FOUND IN DB2 XCF GROUP(***db2-group-name***). MEMBER HAS BEEN DELETED.**

Explanation: The invalid member was found in a DB2 data sharing group, which is a particular type of cross-system coupling facility (XCF) group. The invalid name *member-name* was found in data sharing group *db2-group-name* and was deleted. *member-name* might represent an unauthorized program, other than DB2 for MVS/ESA, that attempted to join the data sharing group.

System action: The DB2 subsystem continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

Operator response: Report the occurrence of this message to your system programmer.

System programmer response: Investigate which unauthorized program joined the DB2 group and find its owner. This could indicate a security violation.

DSN7405I   *csect-name* **MEMBER(***member-name***) UNABLE TO JOIN DATA SHARING GROUP(***group-name***), MVS IXCJOIN RETURN CODE=***xxxxxxxx***, MVS IXCJOIN REASON CODE=***yyyyyyyy***.**

Explanation: An unexpected error occurred when the data sharing DB2 member attempted to join the MVS cross-system coupling facility (XCF) data sharing group.

System action: The DB2 subsystem terminates with reason code 00F70408.

Operator response: Issue an MVS 'DISPLAY XCF,GROUP' command to ensure that XCF is functional; then try to restart the DB2 subsystem.

System programmer response: Refer to *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the IXCJOIN return and reason codes.

DSN7406I *csect-name* **UNABLE TO QUERY DATA SHARING GROUP INFORMATION, MVS IXCQUERY RETURN CODE=***xxxxxxxx*, **MVS IXCQUERY REASON CODE=***yyyyyyyy*.

Explanation: An unexpected error occurred during the execution of the MVS IXCQUERY macro.

System action: The DB2 subsystem terminates with reason code 00F70406.

Operator response: Issue an MVS DISPLAY XCF,GROUP command to ensure that XCF is functional. Then try to restart the DB2 subsystem again.

System programmer response: Refer to *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the IXCQUERY return and reason codes.

DSN7407I *csect-name* **AUTOMATIC RESTART MANAGER** *function* **FAILED. MVS IXCARM RETURN CODE =** *xxxxxxxx*, **MVS IXCARM REASON CODE =** *yyyyyyyy*.

Explanation: DB2 encountered an error using the specified MVS automatic restart manager (ARM) function.

System action: DB2 continues. However, if DB2 abnormally terminates, the MVS ARM cannot automatically restart DB2.

System programmer response: Refer to *MVS/ESA Programming: Sysplex Services Reference* for the function, return code, and reason code from the IXCARM request.

DSN7408I *csect-name* **ARM EVENT EXIT FAILED. DB2 SUBSYSTEM** *ssnm* **NOT RESTARTED. MVS IEFSSI RETURN CODE =** *xxxxxxxx*, **MVS IEFSSI REASON CODE =** *yyyyyyyy*.

Explanation: The DB2 ARM event exit encountered an error while attempting to define the DB2 subsystem to MVS.

System action: The automatic restart of the DB2 subsystem is stopped.

System programmer response: Refer to *MVS/ESA Programming: Assembler Services Reference* for return and reason codes from the IEFSSI request. If the IEFSSI return code and reason code are 0, look in the console log for previous messages that indicate why the definition of the DB2 subsystem failed. You can manually start DB2 by using the -START DB2 command on any system where the DB2 subsystem is defined.

DSN7409I *csect-name* **MEMBER(***member-name***) UNABLE TO QUIESCE FROM DATA SHARING GROUP(***group-name***), MVS IXCQUIES RETURN CODE=***xxxxxxxx*, **MVS IXCQUIES REASON CODE=***yyyyyyyy*.

Explanation: An unexpected error occurred when the data sharing DB2 member attempted to place itself in an MVS cross-system coupling facility (XCF) quiesced state.

System action: The DB2 subsystem termination continues.

Operator response: Inform your system programmer.

System programmer response: Refer to *MVS/ESA Programming: Sysplex Services Reference* for an explanation of the IXCQUIES return and reason codes.

DSN7410I *csect-name* **MEMBER** *member-name* **IS ATTEMPTING TO START DB2 IN A DATA SHARING GROUP WITH AN INCOMPATIBLE SCA FORMAT.**

Explanation: A DB2 member is attempting to start at a code level that is not compatible with the current catalog level of the data sharing group. The formats of the SCA (shared communications area) for these two levels are not compatible.

csect-name
    The name of the control section that issued the message.

| *member-name*
| The name of the DB2 member.

| This message can occur when the starting DB2 member is at a release level for which coexistence with the data
| sharing group is not supported.

| **System action:** DB2 terminates with reason code 00F70403.

| **Operator response:** Verify the following items:
| • Coexistence is supported between the two DB2 releases that you are trying to run.
| • The DB2 startup procedure references the correct set of load libraries.

| **Problem determination:** Collect the following diagnostic items:
| • Console output from the system on which the job was run, and a listing of the SYSLOG data set for the period of
| time spanning the failure.
| • Dynamic dump, taken to SYS1.DUMPxx data set, by DB2 (04E and 04F abends).
| • Contents of the BSDS (both copies if dual BSDS in use) obtained by using the DB2 print log map (DSNJU004)
| utility.
| **Related concepts**:

| Mixed releases in a data sharing group (DB2 Installation and Migration)
| **Related tasks**:

| Determining the release of a data sharing group in coexistence (DB2 Installation and Migration)

| Collecting data (Collecting data)
| **Related information**:

| 00F70403 (DB2 Codes)

DSN7501A   *csect-name* **SCA STRUCTURE** *sca-structure-name* **CONNECTIVITY FAILURE.**

Explanation: The specified DB2 member lost connectivity to the shared communication area (SCA) structure.

System action: The specified DB2 disconnects from the SCA and terminates with reason code 00F70600.

Operator response: Notify the system programmer.

System programmer response: Find out why the link from this central processor complex (CPC) to the SCA
structure was lost. After fixing it, restart DB2.

DSN7502I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **FAILURE, ATTEMPT TO REBUILD IS IN
PROGRESS.**

Explanation: The specified DB2 member detected the shared communications area (SCA) structure failure.

System action: DB2 tries to rebuild the SCA structure on a coupling facility in the preference list for the coupling
facility.

Operator response: Wait for message DSN7503I or DSN7504A.

DSN7503I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **REBUILD SUCCESSFUL.**

Explanation: This member successfully connected to a shared communications area (SCA) structure on the same or
an alternate coupling facility and completed restoration of its information in the shared communications area (SCA).

System action: DB2 processing continues.

DSN7504I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **REBUILD UNSUCCESSFUL. REASON CODE = *xx*.**

Explanation: This member has not successfully connected to a shared communication area (SCA) structure on an
alternate coupling facility and completed restoration of its information in the shared communications area (SCA).
This message follows message DSN7502I.

04   An operator requested to stop the rebuild process.

08   A DB2 resource manager requested to stop the rebuild process.

12   The new SCA structure failed during the rebuild process.

16   A DB2 member lost connectivity to the new SCA in the rebuild process.

20   An MVS service used during the rebuild process failed. A message was issued from the failed MVS service.

System action: This DB2 member terminates with reason code 00F70601. Other members can continue processing.

System programmer response: See the termination code for more details.

DSN7505A   *csect-name* **THERE IS A SHORTAGE OF FREE STORAGE IN SCA STRUCTURE** *sca-structure-name*

Explanation: The amount of free storage in the shared communications area (SCA) coupling facility structure is low. The SCA is 75 percent full. Because SCA storage space is required by some DB2 commands and processes, the availability of those processes might be affected. Completely running out of SCA storage can cause DB2 subsystem outages.

System action: DB2 processing continues.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

This message might not be automatically deleted in a pre SP5.2.1-JES3 environment.

Operator response: If the message remains highlighted after the error situation is cleared up, manually delete the message from the console.

System programmer response: You can monitor the amount of SCA storage in use with the DISPLAY GROUP command.

You can free SCA storage by removing exception conditions. Another thing you can do is issue START DATABASE commands with the SPACENAM option or use the RECOVER utility to remove pages from the logical page list (LPL).

If you don't do anything to reclaim space, message DSN7512A appears when the SCA is 90 percent full.

DSN7506A   *csect-name* **CONNECT TO THE SCA STRUCTURE** *sca-structure-name* **FAILED. MVS IXLCONN RETURN CODE=***xxxxxxxx***, MVS IXLCONN REASON CODE=***yyyyyyyy***.**

Explanation: DB2 cannot connect to the shared communications area (SCA) coupling facility structure.

System action: DB2 startup terminates with reason code 00F70602.

System programmer response: DB2 uses the MVS IXLCONN service to connect to the SCA structure. Refer to *MVS/ESA Programming: Sysplex Services Reference* for return and reason codes from the IXLCONN request.

DSN7507I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **IS ALLOCATED IN A VOLATILE STRUCTURE.**

Explanation: The shared communications area (SCA) structure is volatile.

System action: DB2 processing continues.

System programmer response: One DB2 issues this message whenever the structure has the volatile attribute. DB2 detects this either on the first connection to the SCA or on the attribute changes later. DB2 always requests a nonvolatile SCA. If this is not what you want, change the MVS CFRM administrative policy and request a rebuild of the SCA with the MVS SETXCF START,REBUILD command.

DSN7508I   *csect-name* **ACCESS TO THE SCA STRUCTURE** *sca-structure-name* **FAILED. MVS IXLLIST RETURN CODE = ***xxxxxxxx***, MVS IXLLIST REASON CODE = ***yyyyyyyy***.**

Explanation: DB2 encountered an error in accessing the shared communications area (SCA) structure.

System action: DB2 terminates with reason code 00F70603.

System programmer response: DB2 uses the MVS IXLLIST service to access the SCA structure. Refer to *MVS/ESA Programming: Sysplex Services Reference* for return and reason codes from the IXLLIST request.

DSN7509I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **IS ALLOCATED IN A NONVOLATILE STRUCTURE.**

Explanation: The SCA structure is now contained in a nonvolatile coupling facility structure.

System action: DB2 processing continues.

DSN7510I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **REBUILD REQUEST RECEIVED, BUT SUBSYSTEM IS NOT READY TO PARTICIPATE.**

Explanation: The DB2 subsystem received a request from MVS to rebuild the shared communications area (SCA), but DB2 is not ready to participate in this process because DB2 startup has not yet completed.

System action: DB2 terminates with reason code 00F70604. The SCA rebuild process might continue for other DB2s in the data sharing group.

Operator response: Wait until the SCA rebuild process is finished and then start the failed DB2 again.

DSN7511I   *csect-name* **RESPONSE TO MVS REBUILD COMPLETE OF SCA STRUCTURE** *sca-structure-name* **FAILED. MVS IXLREBLD RETURN CODE =** *xxxxxxxx*, **MVS IXLREBLD REASON CODE =** *yyyyyyyy*.

Explanation: DB2 encountered an error during the rebuild of the shared communications area (SCA) structure.

System action: DB2 stops.

System programmer response: DB2 uses the MVS IXLREBLD service during the SCA structure rebuild. Refer to *MVS/ESA Programming: Sysplex Services Reference* for return and reason codes from the IXLREBLD request.

DSN7512A   *csect-name* **THERE IS A CRITICAL SHORTAGE OF FREE STORAGE IN SCA STRUCTURE** *sca-structure-name*

Explanation: The amount of free storage in the shared communications area (SCA) coupling facility structure is critically low. The SCA is 90 percent full.

System action: DB2 processing continues. However, DB2 limits use of SCA to critical information. Some utilities or commands that cannot be completed can cause one or more members of the group to come down.

This message is non-scrollable. It remains highlighted on the console until you take the appropriate action.

This message might not be automatically deleted in a pre SP5.2.1-JES3 environment.

System programmer response: You can monitor the amount of SCA storage in use with the DISPLAY GROUP command.

You can free SCA storage by removing exception conditions. Another thing you can do is issue START DATABASE commands with the SPACENAM option or use the RECOVER utility to remove pages from the logical page list (LPL). Then restart any failed DB2s.

If your actions do not free up enough space, or if the problem continues to occur, you have the following options, depending on what level of MVS and the coupling facility control code you are running with:

• If all members of the group are running with MVS Version 5 Release 2 and with a coupling facility control code level greater than 0, issue the SETXCF START,ALTER command to increase the size of the SCA. The size you specify must be less than or equal to the maximum size defined the CFRM policy for the SCA structure.

  If the maximum size (SIZE in the CFRM policy) is still not big enough, take the actions described below for MVS Version 5 Release 1 systems.

• If any or all members of the group are running with MVS Version 5 Release 1 or with a coupling facility control code level of 0, increase the storage for the SCA in the CFRM policy. Then issue the MVS SETXCF START,REBUILD command to rebuild the structure.

Related concepts:

⏩ Problem: storage shortage in the SCA (DB2 Data Sharing Planning and Administration)

Related reference:

⏩ -DISPLAY GROUP (DB2) (DB2 Commands)

⏩ -START DATABASE (DB2) (DB2 Commands)

DSN7513I   *csect-name* **SCA STRUCTURE** *sca-structure-name* **HAS ADEQUATE FREE STORAGE.**

Explanation: The shared communications area (SCA) coupling facility structure has adequate free storage to function properly.

This message is shown in the following circumstances:

- After the SCA was rebuilt.
- After the SCA usage goes below 67%, after its usage was up to 75% or higher.

System action: DB2 processing continues.

DSN7514I   *csect-name* **RESPONSE TO AN MVS REBUILD RELATED EVENT FOR THE SCA STRUCTURE** *sca-structure-name* **FAILED. MVS IXLEERSP RETURN CODE =** *xxxxxxxx***, MVS IXLEERSP REASON CODE =** *yyyyyyyy***.**

Explanation: DB2 encountered an error during the rebuild of the shared communications area (SCA) structure.

System action: DB2 terminates.

System programmer response: DB2 uses the MVS IXLEERSP service to respond to SCA structure rebuild. Refer to *MVS/ESA Programming: Sysplex Services Reference* for return and reason codes from the IXLEERSP request.

DSN7515I   **DB2 MEMBER(***member-name***) IS WAITING TO JOIN THE XCF GROUP.**

Explanation: DB2 was waiting to join the XCF group, but XCF has not completed cleanup for this member's last instance. Cleanup must be completed before DB2 restart processing can proceed.

member-name
      The name of the DB2 member that is attempting to join the XCF group.

System action: DB2 will continue to try to join the XCF group periodically for up to five minutes. If DB2 is unable to join the group within five minutes, message DSN7403I is issued and restart processing terminates.

Operator response: No immediate action is required. Wait for DB2 restart to succeed or fail. If restart processing fails repeatedly, determine the status of XCF during the time that DB2 is waiting to join the XCF group. This information is used in problem determination for message DSN7403I.

Problem determination: While DB2 is waiting for XCF cleanup to complete, issue the following z/OS command, substituting the name of the data sharing group for *group-name*:

```
DISPLAY XCF, GROUP, group-name, ALL
```

Related reference:

⏩ z/OS displaying Cross System Coupling Facility information (MVS System Commands)

Related information:

"DSN7403I" on page 1119

Chapter 26. DSN8 messages

Message identifiers that begin with DSN8 communicate information about sample applications.

DSN8000I   **REQUEST IS:** *request*

Explanation: The input statement is listed as received.

System action: The request is listed.

User response: See if the input request is what you want.

DSN8001I   *csect* **EMPLOYEE NOT FOUND**

Explanation: The 'l' or 'list' request found no employee data that match your request.

System action: No employees are listed.

User response: If no employee data was expected to be found, continue.

Problem determination: If the error is unexpected, list the contents of the employee and department views VEMP and VDEPT to determine why the entry is missing.

DSN8002I   *csect* **EMPLOYEE SUCCESSFULLY ADDED**

Explanation: Employee data was added as requested.

System action: The request was executed successfully.

DSN8003I   *csect* **EMPLOYEE SUCCESSFULLY ERASED**

Explanation: Data for an employee was erased as requested.

System action: The request was executed successfully.

DSN8004I   *csect* **EMPLOYEE SUCCESSFULLY UPDATED**

Explanation: Data for an employee was updated as requested.

System action: The request was executed successfully.

DSN8005E   *csect* **EMPLOYEE EXISTS ALREADY, ADD NOT DONE**

Explanation: A request was made to add data for a new employee, but the employee number already exists.

System action: No employee data is added.

User response: Display the existing employee data. If this is the desired employee, update the employee entry. Otherwise, choose a new employee number.

Problem determination: List the contents of the employee view, VEMP, and the department view, VDEPT.

DSN8006E   *csect* **EMPLOYEE DOES NOT EXIST, ERASE NOT DONE**

Explanation: The erase requested was not performed. The employee number requested was not present in the table.

System action: No employee data is erased.

User response: Determine if the employee data to be erased should exist. Verify that the correct search criteria and data were entered. If appropriate, correct the entries and continue processing.

Problem determination: List the contents of the employee view, VEMP, and the department view, VDEPT.

DSN8007E *csect* **EMPLOYEE DOES NOT EXIST, UPDATE NOT DONE**

Explanation: The update requested was not performed. The employee number requested was not present in the table.

System action: No employee data is updated.

User response: Verify that the correct search criteria and data were entered.

Problem determination: List the contents of the employee table (VPHONE, VEMPLP, or EMP) and see if the employee number to be updated is in the table. Correct the update request or the table.

DSN8008I *csect* **NO EMPLOYEE FOUND IN TABLE**

Explanation: No employees match the selection criteria.

System action: No employee data is listed.

User response: If no employee data was expected, continue processing.

Problem determination: List the contents of the VPHONE view and the underlying tables.

DSN8011I *csect* **DEPARTMENT NOT FOUND**

Explanation: No departments match the selection criteria.

System action: No departments are listed.

User response: If no departments are expected to be found, continue.

Problem determination: If the error is unexpected, list the contents of the department views VDEPT and VDEPMG1 and their underlying tables to determine why the entry is missing.

DSN8012I *csect* **DEPARTMENT SUCCESSFULLY ADDED**

Explanation: A department was added as requested.

System action: The request was executed successfully.

DSN8013I *csect* **DEPARTMENT SUCCESSFULLY ERASED**

Explanation: Data for a department was erased as requested.

System action: The request was executed successfully.

DSN8014I *csect* **DEPARTMENT SUCCESSFULLY UPDATED**

Explanation: Data for a department was updated as requested.

System action: The request was executed successfully.

DSN8015E *csect* **DEPARTMENT EXISTS ALREADY, ADD NOT DONE**

Explanation: You requested that a new department be added, but the department exists already.

System action: No departments are added.

User response: Display data for the existing department. If it is what you want, continue. If not, you can update data for this department or choose a new department id.

Problem determination: If the error is unexpected, list the contents of the department views VDEPT and VDEPMG1 and their underlying tables to see if the data differs from what is expected.

DSN8016E   *csect* **DEPARTMENT DOES NOT EXIST, ERASE NOT DONE**

Explanation: The erase request found no departments which match your request.

System action: No department data is erased.

User response: See if the department data to be erased should exist. Did you enter the correct combination of search criteria and data? If not, correct the entries and continue.

Problem determination: If the error is unexpected, list the contents of the department views VDEPT and VDEPMG1 and their underlying tables to determine why the entry is missing.

DSN8017E   *csect* **DEPARTMENT DOES NOT EXIST, UPDATE NOT DONE**

Explanation: The update request found no department data which matches your request.

System action: No department data is updated.

User response: See if the department data to be updated should exist. If you did not enter the correct combination of search criteria and data, correct the entries and continue.

Problem determination: If the error is unexpected, list the contents of the department views VDEPT and VDEPMG1 and their underlying tables to determine why the entry is missing.

DSN8018I   *csect* **CURRENT DEPARTMENT NOT FOUND**

Explanation: A request was made by a 'current' department to reference a 'higher' department. The 'current' department was not found.

System action: No department data is processed.

User response: If the 'current' department was not expected to exist, continue processing.

Problem determination: List the contents of the department views VDEPMG1, VEMP, VASTRDE1, and VASTRDE2 and their underlying tables.

DSN8019E   *csect* **NO HIGHER DEPARTMENT EXISTS**

Explanation: A request was made by a 'current' department to reference a 'higher' level of the department structure. A 'higher' level does not exist for the requested 'current' department.

System action: No department data is processed.

User response: If a 'higher' department was not expected to be found, continue processing.

Problem determination: List the contents of the department views VDEPMG1, VEMP, VASTRDE1, and VASTRDE2 and their underlying tables.

DSN8021I   *csect* **ACTIVITY NOT FOUND**

Explanation: No activities match the selection criteria.

System action: No activity is listed.

User response: If no activity was expected to be found, continue processing.

Problem determination: List the contents of the activity views VPROJRE1, VACT, and VPROJACT and their underlying tables.

DSN8022I   *csect* **ACTIVITY SUCCESSFULLY ADDED**

Explanation: An activity was added as requested.

System action: The request was executed successfully.

DSN8023I   *csect* **ACTIVITY SUCCESSFULLY ERASED**

Explanation: An activity was erased as requested.

System action: The request was executed successfully.

DSN8024I   *csect* **ACTIVITY SUCCESSFULLY UPDATED**

Explanation: An activity was updated as requested.

System action: The request was executed successfully.

DSN8025E   *csect* **ACTIVITY EXISTS ALREADY, ADD NOT DONE**

Explanation: The addition requested was not performed. The activity requested was already present.

System action: No activity is added.

User response: If the activity was expected to already exist, continue processing.

Problem determination: List the contents of the activity views VPROJRE1, VACT, and VPROJACT and their underlying tables.

DSN8026E   *csect* **ACTIVITY DOES NOT EXIST, ERASE NOT DONE**

Explanation: The erase requested was not performed. The activity requested was not found.

System action: No activity is erased.

User response: If the activity was not expected to be found, continue processing.

Problem determination: List the contents of the activity views VPROJRE1, VACT, and VPROJACT and their underlying tables.

DSN8027E   *csect* **ACTIVITY DOES NOT EXIST, UPDATE NOT DONE**

Explanation: The update requested was not performed. The activity requested was not found.

System action: No activity is updated.

User response: Determine if the activity to be updated should exist. Verify that the correct search criteria and data were entered. If appropriate, correct the entries and continue processing.

Problem determination: List the contents of the activity views VPROJRE1, VACT, and VPROJACT and their underlying tables.

DSN8028I   *csect* **ACTIVITY TYPE NOT FOUND, PLEASE ADD IT INTERACTIVELY**

Explanation: The add or update was not performed. The activity type entered was not found.

System action: No activity is listed.

User response: If the activity type was not expected to be found, add it interactively.

Problem determination: List the contents of the activity view, VACT, and its underlying tables.

DSN8031I   *csect* **PROJECT NOT FOUND**

Explanation: No projects match the selection criteria.

System action: No projects are listed.

User response: If no projects are expected to be listed, continue.

Problem determination: If the error is unexpected, list the contents of the project view VPROJ and its underlying tables to determine why the entry is missing.

DSN8032I   *csect* **PROJECT SUCCESSFULLY ADDED**

Explanation: A project was added as requested.

System action: The request was executed successfully.

DSN8033I   *csect* **PROJECT SUCCESSFULLY ERASED**

Explanation: A project was erased as requested.

System action: The request was executed successfully.

DSN8034I   *csect* **PROJECT SUCCESSFULLY UPDATED**

Explanation: The update was performed as requested.

System action: The request was executed successfully.

DSN8035E   *csect* **PROJECT EXISTS ALREADY, ADD NOT DONE**

Explanation: You requested that a new project be added, but the project exists already.

System action: The request was not executed successfully.

User response: Display the existing project. If it is what you want, continue. If not, you can update this project or choose a new project id.

Problem determination: If the error was unexpected, list the contents of the project view VPROJ and its underlying tables to see if the data differs from what is expected.

DSN8036E   *csect* **PROJECT DOES NOT EXIST, ERASE NOT DONE**

Explanation: The erase request found no project which matches your request.

System action: No projects are erased.

User response: See if the project to be erased should exist. Did you enter the correct combination of search criteria and data? If not, correct the entries and continue.

Problem determination: If the error was unexpected, list the contents of the project view VPROJ and its underlying tables to determine why the entry is missing.

DSN8037E   *csect* **PROJECT DOES NOT EXIST, UPDATE NOT DONE**

Explanation: The update request found no project which matches your request.

System action: No projects are updated.

User response: See if the project to be updated should exist. If you did not enter the correct combination of search criteria and data, correct the entries and continue.

Problem determination: If the error is unexpected, list the contents of the project view VPROJ and its underlying tables to determine why the entry is missing.

DSN8038I   *csect* **CURRENT PROJECT NOT FOUND**

Explanation: A request was made by a 'current' project to reference a 'higher' project. The 'current' project was not found.

System action: No project data is processed.

User response: If the 'current' project was not expected to be found, continue processing.

Problem determination: List the contents of the project views VPROJRE1, VDEPT, VEMPPROJACT, VEMP, VSTAFAC1, VSTAFAC2, VPSTRDE1, and VPSTRDE2 and their underlying tables.

DSN8039E   *csect* **NO HIGHER PROJECT EXISTS**

Explanation: A request was made by a 'current' project to reference a 'higher' level of the project structure. A 'higher' level does not exist for the requested 'current' project.

System action: No project data is processed.

User response: If a 'higher' project was not expected to be found, continue processing.

Problem determination: List the contents of the project views VPROJRE1, VDEPT, VEMPPROJACT, VEMP, VSTAFAC1, VSTAFAC2, VPSTRDE1, and VPSTRDE2 and their underlying tables.

DSN8041I   *csect* **STAFFING NOT FOUND**

Explanation: No project staffing matches the selection criteria.

System action: No project staffing is listed.

User response: If no project staffing was expected to be found, continue processing.

Problem determination: List the contents of the project activity views VPROJRE1, VEMPPROJACT, VACT, and VEMP and their underlying tables.

DSN8042I   *csect* **STAFFING SUCCESSFULLY ADDED**

Explanation: A project staffing was added as requested.

System action: The request was executed successfully.

DSN8043I   *csect* **STAFFING SUCCESSFULLY ERASED**

Explanation: A project staffing was erased as requested.

System action: The request was executed successfully.

DSN8044I   *csect* **STAFFING SUCCESSFULLY UPDATED**

Explanation: A project staffing was updated as requested.

System action: The request was executed successfully.

DSN8045E   *csect* **STAFFING EXISTS ALREADY, ADD NOT DONE**

Explanation: A request was made to add a project staffing, but the project staffing already exists.

System action: No project staffing is added.

User response: Display the existing project staffing. If this is the desired project staffing, update it. Otherwise, choose a new project staffing.

Problem determination: List the contents of the project activity views VPROJRE1, VEMPPROJACT, VACT, and VEMP and their underlying tables.

DSN8046E   *csect* **STAFFING DOES NOT EXIST, ERASE NOT DONE**

Explanation: The project staffing was not erased as requested.

System action: No project staffing is erased.

User response: Determine if the project staffing to be erased should exist. Verify that the correct search criteria and data were entered. If appropriate, correct the entries and continue processing.

Problem determination: List the contents of the project activity views VPROJRE1, VEMPPROJACT, VACT, and VEMP and their underlying tables.

DSN8047E   *csect* **STAFFING DOES NOT EXIST, UPDATE NOT DONE**

Explanation: The update was not performed because the project staffing requested was not found.

System action: No project staffing is updated.

User response: Determine if the project staffing to be updated should exist. Verify that the correct search criteria and data were entered. If appropriate, correct the entries and continue processing.

Problem determination: List the contents of the project activity views VPROJRE1, VEMPPROJACT, VACT, and VEMP and their underlying tables.

DSN8051I   *csect* **PROGRAM ENDED**

Explanation: The application program finished.

DSN8053I   *csect* **ROLLBACK SUCCESSFUL, ALL UPDATES REMOVED**

Explanation: After any SQLWARNING or SQLERROR, all updates are removed via a ROLLBACK statement to prevent invalid updates.

System action: The ROLLBACK was executed successfully.

User response: Correct the prior error.

DSN8056I   *csect* **NO MORE DATA TO DISPLAY**

Explanation: The last page of data has been displayed.

System action: No more data is shown on screen.

User response: If you expect no more data to be displayed, continue.

Problem determination: If more displayed data is expected, list the contents of the view and the underlying tables to see if the data differs from what is expected.

DSN8057I   *csect* **FURTHER ENTRIES IN TABLE - UPDATE POSSIBLE**

Explanation: More data exists beyond that shown on the screen.

System action: No more data is shown on screen.

User response: Update entries as needed.

DSN8058I   *csect* **PRESS PA1 FOR NEXT PAGE / ENTER FOR SELECTION MENU**

Explanation: Press the PA1 key to see the data on the next page or press the ENTER key to return to the selection menu.

User response: Press the PA1 key or the ENTER key depending on the function you want to perform.

DSN8060E   *csect-name* **SQL ERROR, RETURN CODE IS:** *sql-code*

Explanation: The SQL statement did not execute correctly. The *sql-code* value might be an SQL error or an SQL warning.

System action: The request was not executed successfully.

User response: Determine which SQL statement in the application program caused the error and see if it is reproducible.

Problem determination: Follow the diagnosis approach suggested for the SQL code.

Related concepts:

SQL codes (DB2 Codes)

DSN8061E   *csect-name* **ROLLBACK FAILED, RETURN CODE IS:** *sql-code*

Explanation: The SQL ROLLBACK statement was not executed correctly.

System action: The request was not executed successfully.

User response: Take the action that is indicated for the SQLCODE that is identified *sql-code* value.

Problem determination: Determine which SQL statement in the application program caused the error and see if it is reproducible. Follow the diagnosis approach suggested for the SQLCODE that is identified *sql-code* value.

Related concepts:

➡ SQL codes (DB2 Codes)

DSN8062E   *csect* **AN OBJECT WAS NOT SELECTED**

Explanation: You did not specify an object code on the input line.

User response: Select one of the object codes listed on the panel.

DSN8064E   *csect* **INVALID DL/I STC-CODE ON GU MSG:** *status-code*

Explanation: In attempting to get a message from IMS, an unexpected status-code was received.

System action: The application program is terminated.

Problem determination: Refer to the messages and codes manual for IMS/VS to determine the reason for the unexpected status code.

DSN8065E   *csect* **INVALID DL/I STC-CODE ON ISRT MSG:** *status-code*

Explanation: In attempting to insert a message for IMS, an unexpected status-code was received.

System action: The application program is terminated.

Problem determination: Refer to the messages and codes manual for IMS/VS to determine the reason for the unexpected status code.

DSN8066E   *csect* **UNSUPPORTED PFK OR LOGIC ERROR**

Explanation: The sample application found an unsupported program function key or logic error while trying to execute a request.

System action: The request was not executed successfully.

User response: Report the error in the application program.

DSN8067E   *csect* **UNSUPPORTED SEARCH CRITERIA FOR OBJECT**

Explanation: You entered a search criteria value and an object value that do not find any selections.

System action: The request was not executed successfully.

User response: Enter new search criteria values.

Problem determination: If the values you entered are expected to provide valid selections, check the contents of the view and the underlying tables to see if the data differs from what is expected. If after checking the view and the underlying tables, you find the search criteria values you entered should provide valid selections, the error is in the sample program. Report the error.

DSN8068E   *csect* **INVALID REQUEST SHOULD BE 'L' OR 'U'**

Explanation: The input must begin with 'L' or 'U'.

System action: The request is ignored.

User response: Correct the input request.

DSN8069E   *csect* **NO VALID SELECTIONS QUALIFY FOR THIS REQUEST**

Explanation: You entered a search criteria value and a data values that do not find any selections.

System action: The request was not executed successfully.

User response: Enter new search criteria values.

Problem determination: If the criteria values you entered are expected to provide valid selections, check the contents of the view and the underlying tables to see if the data differs from what is expected. If after checking the view and the underlying tables, you find the search criteria values you entered should provide valid selections, the error is in the sample program. Report the error.

DSN8070E   *csect* **VITAL DATA IS MISSING IN TABLE TOPTVAL**

Explanation: No valid entries exist in the table that contains possible valid entries.

System action: The request was not executed successfully.

User response: There is no valid entry you can enter. Report the error in the sample program.

DSN8072E   *csect* **INVALID SELECTION ON SECONDARY SCREEN**

Explanation: You gave an invalid line number.

System action: The request was not executed successfully.

User response: You entered a number that is not in the list on the left under the heading NO for line number. Enter a valid number.

Problem determination: If the line number you entered is in the list on the left under the heading NO for line number, then the error is in the sample program.

DSN8073E   *csect* **SPECIFIED LINE-NUMBER NOT FOUND IN PREVIOUS SCREEN**

Explanation: You gave an invalid line number.

System action: The request was not executed successfully.

User response: You entered a number that is not in the list on the left under the heading NO for line number. Enter a valid number. If the line number is in the previous screen, then this is an error in the sample program. Report the error.

DSN8074E   *csect* **DATA IS TOO LONG FOR SEARCH CRITERIA**

Explanation: You entered a data value that has too many characters for the type of search criterion you selected. For example, department ids have a maximum of 3 characters and most other ids have a maximum of 6 characters.

System action: The request was not executed successfully.

User response: Enter a data value with the correct number of characters.

DSN8075E   *csect-name* **MESSAGE FORMAT ROUTINE ERROR, RETURN CODE IS:** *sql-code*

Explanation: There is an error in the parameters passed to the SQLCA formatting routine, DSNTIAR.

System action: The SQLCA could not be formatted properly.

User response: This is an error in the program. Report the error to the application programmer.

Related concepts:

↱ DSNTIAR (DB2 Application programming and SQL)

Related tasks:

↱ Displaying SQLCA fields by calling DSNTIAR (DB2 Application programming and SQL)

Related reference:

↱ Possible return codes from DSNTIAR (DB2 Application programming and SQL)

DSN8076E *csect* **VITAL DATA IS MISSING IN TABLE 'TDSPTXT'**

Explanation: No valid entries exist in the table that contains possible valid entries.

System action: The request was not executed successfully.

User response: There is no valid entry you can enter. Report the error in the application program.

DSN8079E **CONNECTION TO DB2 NOT ESTABLISHED**

Explanation: Sample programs DSN8SC3, DSN8SP3, and DSN8HC3 cannot be called directly from TSO. Instead, they must be called from the ISPF sample connection manager, DSN8SCM or DSN8SPM.

System action: The program ends.

User response: Call DSN8SCM or DSN8SPM through the connection manager.

DSN8080E **APPLICATION TERMINATED WITH CALL ATTACH REASON CODE** *reason-code*

Explanation: The call attachment facility service requested by the application program did not execute successfully.

System action: The requested service is not executed.

System programmer response: Check the call attachment facility reason code.

User response: Notify the system programmer.

Problem determination: Follow the failure analysis procedures for the call attachment facility reason code.

DSN8081E **RESULTS FROM THE CALL ATTACH TRANSLATE SERVICE**

Explanation: The call attachment facility OPEN service that was requested from the application program did not execute successfully. The TRANSLATE service was requested to provide information about the OPEN failure.

System action: The requested OPEN service was not executed.

User response: If you suspect an error in DB2, you might need to report the problem. For information about identifying and reporting the problem, see Collecting data (Collecting data) and DB2 Diagnosis Guide and Reference.

Related reference:

↱ CAF return codes and reason codes (DB2 Application programming and SQL)

DSN8082E **THE DISCONNECT TERMINATED WITH REASON CODE** *reason-code*

Explanation: The call attachment facility DISCONNECT service requested by the application program did not execute successfully.

System action: The requested DISCONNECT service is not executed.

System programmer response: Check the call attachment facility reason code.

User response: Notify the system programmer.

Problem determination: Follow the failure analysis procedures for the call attachment facility reason code.

DSN8083E   AN UNKNOWN SERVICE *service* WAS REQUESTED

Explanation: The user is requesting an invalid service.

System action: The requested service was not executed.

System programmer response: The service request must be a valid call attachment facility service. Ensure that the request is one of the services supported by the call attachment facility.

User response: Notify the system programmer.

Problem determination: There is an error in the Sample Application Program.

DSN8200E   *csect* INVALID DEPARTMENT NUMBER, EMPLOYEE NOT ADDED

Explanation: The employee number that is to be added refers to a department number that is not contained in the table DEPT.

System action: The employee number is not added.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the department table DEPT.

DSN8202E   *csect* EMPLOYEE NUMBER HAS DEPENDENT ROWS, NOT ERASED

Explanation: The employee number to be erased is the same as a responsible employee number in the table PROJ or an employee number in the table EMPPROJACT. The employee number cannot be erased from the table EMP.

System action: No employee data is erased.

User response: Ensure that the correct employee number was entered.

Problem determination: List the contents of the PROJ and EMPPROJACT tables.

DSN8203E   *csect* INVALID WORK DEPT, EMPLOYEE NOT UPDATED

Explanation: The employee that is to be updated has a department number that is not contained in the table DEPT. The update cannot be completed.

System action: No employee data is updated.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the employee table DEPT.

DSN8210I   *csect* INVALID MGRNO, DEPARTMENT NOT ADDED

Explanation: The department that is to be added has a manager number that is not contained in the table EMP. The add cannot be completed.

System action: No department is added.

User response: Ensure that the correct manager number was entered.

Problem determination: List the contents of the employee table EMP.

DSN8212E   *csect* DEPARTMENT NUMBER HAS DEPENDENT ROWS, NOT ERASED

Explanation: The department number that is to be erased is the same as a project department number in table PROJ. The department cannot be erased.

System action: No department data is erased.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the project table PROJ.

DSN8213E   *csect* **INVALID ADMIN DEPT ID, DEPARTMENT NOT ADDED**

Explanation: The department that is to be added has a department number that is not contained in the table DEPT. The add cannot be completed.

System action: No department is added.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the employee table DEPT.

DSN8214E   *csect* **INVALID MANAGER ID, DEPARTMENT NOT UPDATED**

Explanation: The department that is to be updated has a manager number that is not contained in the table EMP. The update cannot be completed.

System action: No department data is updated.

User response: Ensure that the correct manager number was entered.

Problem determination: List the contents of the employee table EMP.

DSN8215E   *csect* **INVALID ADMIN DEPT ID, DEPARTMENT NOT UPDATED**

Explanation: The department that is to be updated has a administrative department number that is not in the table DEPT. The update cannot be completed.

System action: No department data is updated.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the employee table DEPT.

DSN8216E   *csect* **DEPT NOT AT SPECIFIED LOCATION, EMPLOYEE NOT ADDED**

Explanation: The employee number that is to be added refers to a department number whose LOCATION field in the DEPT table is different from that specified for the employee. The add cannot be completed.

System action: No employee is added.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the department table DEPT.

DSN8217E   *csect* **DEPT NOT AT SPECIFIED LOCATION, EMP NOT UPDATED**

Explanation: The employee number that is to be updated refers to a department number whose LOCATION field in the DEPT table is different from that specified for the employee. The update cannot be completed.

System action: No employee data is updated.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the department table DEPT.

DSN8220E   *csect* **INVALID PROJECT NUMBER, ACTIVITY NOT ADDED**

Explanation: The activity that is to be added contains a project number that is not contained in the table PROJ. The add cannot be completed.

System action: The activity is not added.

User response: Ensure that the correct project number was entered.

Problem determination: List the contents of the project table PROJ.

DSN8222E *csect* **ACTIVITY HAS DEPENDENT ROWS, NOT ERASED**

Explanation: The activity that is to be erased from table PROJACT is the same as the corresponding activity in the table EMPPROJACT. The activity cannot be erased from the table PROJACT.

System action: No activity data is erased.

User response: Ensure that the correct activity was entered.

Problem determination: List the contents of the PROJACT and EMPPROJACT tables.

DSN8223E *csect* **ACTIVITY HAS DEPENDENT ROWS, NOT UPDATED**

Explanation: A start date in table PROJACT is to be updated, but there is a project staffing for this activity in table EMPPROJACT. The start date in table PROJACT cannot be updated.

System action: No activity data is updated.

User response: Ensure that the correct start date was entered.

Problem determination: List the contents of the project-activity-staffing table EMPPROJACT and project-activity table PROJACT.

DSN8230E *csect* **INVALID DEPARTMENT NUMBER, PROJECT NOT ADDED**

Explanation: The project that is to be added contains a department number that is not present in the table DEPT. The add cannot be completed.

System action: No project is added.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the department table DEPT.

DSN8231E *csect* **INVALID RESPONSIBLE ID, PROJECT NOT ADDED**

Explanation: The project that is to be added contains an employee number that is not contained in the table EMP. The add cannot be completed.

System action: No project is added.

User response: Ensure that the correct employee number was entered.

Problem determination: List the contents of the employee table EMP.

DSN8233E *csect* **PROJECT NUMBER HAS DEPENDENT ROWS, NOT ERASED**

Explanation: A project number in table PROJ is to be erased, but it is the same as a project number in table PROJACT. The project cannot be erased from table PROJ.

System action: No project is erased.

User response: Ensure that correct project number was entered.

Problem determination: List the contents of the PROJ and PROJACT tables.

DSN8235E *csect* **INVALID MAJOR PROJECT ID, PROJECT NOT ADDED**

Explanation: The project that is to be added contains a major project ID that is not contained in the table PROJ. The add cannot be completed.

System action: No project is added.

User response: Ensure that the correct major project number was entered.

Problem determination: List the contents of the project table PROJ.

DSN8236E   *csect* **INVALID ASSOCIATE DEPT ID, PROJECT NOT UPDATED**

Explanation: The project that is to be updated has a department number that is not contained in the table DEPT. The update cannot be completed.

System action: No project data is updated.

User response: Ensure that the correct department number was entered.

Problem determination: List the contents of the department table DEPT.

DSN8237E   *csect* **INVALID RESPONSIBLE ID, PROJECT NOT UPDATED**

Explanation: The project that is to be updated has a employee number that is not contained in the table EMP. The update cannot be completed.

System action: No project data is updated.

User response: Ensure that the correct employee number was entered.

Problem determination: List the contents of the employee table EMP.

DSN8238E   *csect* **INVALID MAJOR PROJECT ID, PROJECT NOT UPDATED**

Explanation: The project that is to be updated has a major project number that is not contained in the table PROJ. The update cannot be completed.

System action: No project data is updated.

User response: Ensure that the correct project number was entered.

Problem determination: List the contents of the project table PROJ.

DSN8240E   *csect* **INVALID EMPLOYEE NUMBER, STAFFING NOT ADDED**

Explanation: The staffing that is to be added refers to an employee number that is not present in table EMP. The add cannot be completed.

System action: No staffing is added.

User response: Ensure that the correct employee number was entered.

Problem determination: List the contents of employee table EMP.

DSN8242E   *csect* **INVALID PROJECT ACTIVITY DATA, STAFFING NOT ADDED**

Explanation: The staffing that is to be added refers to a project number, activity number, and start date that is not present in table PROJACT. The add cannot be completed.

System action: No staffing is added.

User response: Ensure that the correct project number, activity number, and start date were entered.

Problem determination: List the contents of the project table PROJACT.

DSN8243E   *csect* **INVALID START DATE, STAFFING NOT UPDATED**

Explanation: The start date given is not currently in the project-activity table PROJACT. The update cannot be completed.

System action: No staffing data is updated.

User response: Ensure that the correct start date was entered.

Problem determination: List the contents of the project-activity table PROJACT and the project-activity-staffing table EMPPROJACT.

Chapter 27. DSN9 messages

Message identifiers that begin with DSN9 communicate information about the general command processor.

DSN9000I   **KEYWORD** *keyword-name* **APPEARS MORE THAN ONCE**

Explanation: The same keyword (specified in the messages as 'keyword-name') appears more than once in the command. This message will be issued for each occurrence of the keyword after the first.

System action: Processing for the command is terminated.

System programmer response: Correct the command by using the keyword only once, and reenter the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9001I   **KEYWORD** *keyword-name* **IS INVALID**

Explanation: The keyword (specified in the message as 'keyword-name') is unknown or undefined. The keyword might be misspelled, or the keyword may not be applicable to the command being processed.

System action: Processing for the command is terminated.

System programmer response: Correct the keyword, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9002I   **UNBALANCED PARENTHESES WERE FOUND FOLLOWING KEYWORD** *keyword-name*

Explanation: An invalid combination of parentheses has been encountered following the keyword (specified in the message as 'keyword-name'). Parenthesis must be paired open and close, and a close parenthesis must follow an open parenthesis before another open parenthesis is encountered.

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9003I   **KEYWORD** *keyword-name* **PARAMETER CONTAINS UNBALANCED APOSTROPHES**

Explanation: An odd number of apostrophes is present in a parameter. If the parameter is a quoted string, it must have one apostrophe at each end of the string. If an apostrophe is to appear within the string, two adjacent apostrophes must be entered. If the parameter is a hexadecimal value, it must be entered as X'hex characters'.

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9004I   **KEYWORD** *keyword-name* **RANGE (***c***) INCORRECTLY SPECIFIED**

Explanation: The range for a keyword (specified in the message as 'keyword-name') was incorrectly specified using the syntax character specified in the message as *c*. The standard syntax character used to denote a range is a colon (:). Either the ending limit of the range was omitted or a subset was specified within the range.

Examples of this error are LINE(1:) and GROUP(A*:B*).

Examples of a correctly specified range are LINE(1:3) and GROUP(A:B).

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

1139

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

⬆️ The DSN command and its subcommands (DB2 Commands)

Related information:

⬆️ DB2 and related commands (DB2 Commands)

DSN9005I   KEYWORD *keyword-name* **SUBSET (c) INCORRECTLY SPECIFIED**

Explanation: The subset for a keyword (specified in the message as *keyword-name*) was incorrectly specified for use with the syntax character 'c' in the message. The standard syntax character used to denote a subset is an asterisk (*).

For example, consider a group of objects named BLACK, BLUE, GREEN, RED, and WHITE. You may either explicitly specify BLACK and BLUE, or you may specify BL*, meaning all members of the group whose names begin with BL. The character that follows the asterisk must be a blank, comma, close parenthesis, or colon.

Examples of such an error are NAME(BL*CK) and NAME(*LUE).

Examples of correct specification are NAME(BL*) and UNIT(2*).

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related reference:

⬆️ The DSN command and its subcommands (DB2 Commands)

Related information:

⬆️ DB2 and related commands (DB2 Commands)

DSN9006I   KEYWORD *keyword-name* **ASTERISK (c) INCORRECTLY USED**

Explanation: For the keyword specified in the message (as *keyword-name*), an asterisk (c) was used as the first character of a parameter to specify ALL. However, the asterisk appears in a list, or the characters in juxtaposition are not a blank, comma, equal sign, or parentheses.

Examples of such errors are LINE(LINE1,*,LINE7), NAME(**), and GROUP(PQR:*).

Examples of correct specification are UNIT(*) and LINE=* etc.

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related reference:

⬆️ The DSN command and its subcommands (DB2 Commands)

Related information:

⬆️ DB2 and related commands (DB2 Commands)

DSN9007I   THIS COMMAND REQUIRES ONE OF THE FOLLOWING KEYWORDS BE SPECIFIED,
**   *keyword-list***

Explanation: One of the keywords identified in the *keyword-list* list must be specified on this command, but none of these keywords was present.

System action: Processing for the command is terminated.

System programmer response: Identify the appropriate keyword from the list, include that keyword, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

Related reference:

➡ The DSN command and its subcommands (DB2 Commands)

Related information:

➡ DB2 and related commands (DB2 Commands)

DSN9008I   **KEYWORD** *keyword-name* **MAY NOT BE NEGATED**

Explanation: The negation characters (NO) appear in front of the keyword (specified in the message as *keyword-name*), but negating this keyword is not allowed. As an example, consider a keyword called PRINT that could indicate that you want something printed. If you did not want the material to be printed, you might specify NOPRINT. This process of prefixing a keyword with the characters NO is called negation. While negation may be meaningful on some keywords, it is not allowed on all keywords.

System action: Processing for the command is terminated.

System programmer response: Reenter the command, but do not negate the named keyword.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

➡ The DSN command and its subcommands (DB2 Commands)

Related information:

➡ DB2 and related commands (DB2 Commands)

DSN9009I   **KEYWORD** *keyword-name* **IS MISSING AND MUST BE SPECIFIED**

Explanation: The keyword specified in the message (as 'keyword-name') must be present, but it was not entered. This keyword must be present in order for the command to process properly.

System action: Processing for the command is terminated.

System programmer response: Enter the command again, and include the specified keyword.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9010I   **REQUIRED PARAMETER FOR KEYWORD** *keyword-name* **IS MISSING**

Explanation: A parameter must be specified for the keyword specified in the message (as 'keyword-name'), but no parameter was entered.

As an example of this error, consider a keyword called PRIORITY whose value can range from 0 to 9. Specification of PRIORITY(1) is acceptable, but, if you specify PRIORITY with no parameter value, the keyword is meaningless.

System action: Processing for the command is terminated.

System programmer response: Supply appropriate parameters for the specified keyword and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

➡ The DSN command and its subcommands (DB2 Commands)

Related information:

➡ DB2 and related commands (DB2 Commands)

DSN9011I   **PARAMETER(S) CANNOT BE SPECIFIED FOR KEYWORD** *keyword-name*

Explanation: No parameters can be specified for the keyword specified in the message as *keyword-name*. You may have misspelled the keyword-name, causing it to be interpreted as a different keyword, or the parameter may actually belong with another keyword instead of the one on which you entered it.

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

The DSN command and its subcommands (DB2 Commands)

Related information:

DB2 and related commands (DB2 Commands)

DSN9012I   KEYWORD *keyword-name* **PARAMETER(S) MUST BE HEXADECIMAL**

Explanation: Parameter values for the keyword specified in the message (as 'keyword-name') must be hexadecimal values. The value must be specified as X'hex characters'. Hexadecimal characters are defined as the numeric digits 0 through 9 and the letters A through F. Either uppercase or lowercase is acceptable for the letters A through F.

As an example of this error, consider the keyword OFFSET, for which you are expected to enter a hexadecimal value. If you enter OFFSET(X'2C'), it is acceptable. However, entering OFFSET(Jones) is not acceptable.

System action: Processing for the command is terminated.

System programmer response: Reissue the command, but ensure that the parameters for the named keyword are hexadecimal values.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9013I   KEYWORD *keyword-name* **PARAMETER** *parameter value* **EXCEEDS ALLOWABLE LENGTH OF** *nn*

Explanation: The parameter value (specified in the message as *parameter value*) exceeds the limit of *nn* characters in length.

As an example of this type of error, consider a PRIORITY keyword whose value may range from zero to nine. If you specify PRIORITY(21) or PRIORITY(006), the value you have entered exceeds the limit of one character.

System action: Processing for the command is terminated.

System programmer response: Correct the error, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

The DSN command and its subcommands (DB2 Commands)

Related information:

DB2 and related commands (DB2 Commands)

DSN9014I   KEYWORD *keyword-name* **LIMIT OF** *nn* **PARAMETER(S) EXCEEDED**

Explanation: Too many parameters have been specified for the keyword that is specified in the message. The command processor does not accept more than the limit of parameters, which is specified by the value of *nn*. This error is probably caused by entering too many parameters. It could also be caused by a missing close parenthesis that has not yet been detected.

System action: Processing for the command is terminated.

System programmer response: Reissue the command, and enter no more than the specified limit of parameters for the given keyword. If necessary, issue the command more than once to process all the parameter values.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related information:

DB2 and related commands (DB2 Commands)

DSN9015I   PARAMETER *parameter-value* **IS UNACCEPTABLE FOR KEYWORD** *keyword-name*

Explanation: The parameter value specified in the message is not an acceptable value for the named keyword.

As an example, consider a keyword called PRINT that can have parameters of YES or NO. Specification of PRINT(YES) or PRINT(NO) is valid. However, specification of PRINT(MAYBE) is unacceptable.

System action: Processing for the command is terminated.

System programmer response: Correct and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

➡ The DSN command and its subcommands (DB2 Commands)

Related information:

➡ DB2 and related commands (DB2 Commands)

DSN9016I   *verb-name* **COMMAND REJECTED, UNAUTHORIZED REQUEST**

Explanation: The command specified in the message (as 'verb-name') requires that you have proper authorization in order to use the command. You do not have the required authorization.

System action: The command is not executed. Processing is terminated.

Operator response: If the command must be executed on behalf of the user and your installation operating procedures permit it, enter the command on request.

System programmer response: Contact the system programmer responsible for system security, and request that this person grant you authorization to use the command. Otherwise, you must have someone who is authorized issue the command for you. If necessary, request the system operator to enter the command for you.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9017I   **FAILURE WHILE PROCESSING** *verb-name pkw-name* **COMMAND, PROCESSING TERMINATED**

Explanation: The DB2 command processor has abended while processing the command (specified in the message as 'verb-name pkw-name'). The error is been recorded in SYS1.LOGREC, and an SVC dump is requested. The command may have partially completed. Refer to any prior response messages to determine what has been done.

System action: Processing for the command is terminated.

Operator response: If a dump was produced, have it printed. Print a copy of the contents of the SYS1.LOGREC data set for the time period involved. Forward these items with a copy of the system log to the system programmer.

System programmer response: Reissue the command. If a second failure occurs, consult your supervisor.

Problem determination: Obtain a copy of the system or terminal log to verify command entry. Contact the system operator to make sure that diagnostic information is forwarded to you.

DSN9018I   *csect-name* **ABORTING** *verb-name* **PROCESSING DUE TO INSUFFICIENT STORAGE AVAILABLE**

Explanation: The command processor was unable to obtain sufficient storage to complete processing of any response messages generated by the invoked command.

System action: Processing for the command is terminated abnormally.

Operator response: Notify the system programmer before attempting to reissue the command.

System programmer response: Determine the reason for insufficient storage available to the command. If a cursory examination of the system indicates that the region sizes for DB2 are sufficient, and the current workload does not seem excessive, an MVS Console dump (MVS DUMP command) might be required.

Ensure that any retry steps defined by the command are performed prior to reissuing the command.

Problem determination: The invoked command had completed processing and returned to the command preprocessor when an attempt was made to obtain storage from the address space from which the command was entered. Because sufficient storage was unavailable, no response messages from the invoked command are available.

DSN9019I   **VERB** *verb-name* **IS NOT A KNOWN COMMAND**

Explanation: The command represented by the verb specified in the message is not known to the system; it is an undefined command. A possible cause for this error is that you made a spelling error while entering the command.

System action: Processing for the command is terminated.

System programmer response: Correct the format of the command and reissue it.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

⮞ The DSN command and its subcommands (DB2 Commands)

Related information:

⮞ DB2 and related commands (DB2 Commands)

DSN9020I   **ONLY ONE OF THE FOLLOWING KEYWORDS MAY BE SPECIFIED ON THIS COMMAND,** *keyword-list*

Explanation: Only one of the identified keywords in the *keyword-list* list can be specified at the same time on this command.

System action: Processing for the command is terminated.

System programmer response: Identify the appropriate keyword from the list, include only that keyword, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

Related reference:

⮞ The DSN command and its subcommands (DB2 Commands)

Related information:

⮞ DB2 and related commands (DB2 Commands)

DSN9021I   **VERB** *verb-name* **REQUIRED KEYWORD IS MISSING**

Explanation: The command specified in the message (as *verb-name*) was entered, but at least one required keyword is missing. The command processor was unable to determine which keyword or keywords were missing.

System action: Processing for the command is terminated.

System programmer response: Correct the format of the named command, and reissue the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry. If a failure is suspected, take a storage dump.

Related reference:

⮞ The DSN command and its subcommands (DB2 Commands)

Related information:

⮞ DB2 and related commands (DB2 Commands)

DSN9022I   *csect-name* **'***verb-namepkw-name***' NORMAL COMPLETION**

Explanation: All synchronous processing for the specified command completed successfully. Any tasks executing asynchronously for the specified command might still be executing when this message is displayed.

csect-name
        The name of the control section that issued the message.

verb-namepkw-name
        The command name.

System action: Synchronous processing for the specified command is complete. Any subsequent system actions are described in the information for the specified command.

Related information:

▶ DB2 and related commands (DB2 Commands)

DSN9023I   *csect-name verb-name pkw-name* **ABNORMAL COMPLETION**

Explanation: The command specified in the message as 'verb-name pkw-name' completed unsuccessfully. The command issued one or more error messages before this message. The name of the CSECT that issued the message is shown at the beginning of the message as a diagnostic aid.

System action: Processing for the command ended.

System programmer response: Proceed as indicated by previous error messages.

DSN9024I   **KEYWORD** *keyword-name* **PARAMETER MAY NOT SPECIFY A RANGE OF VALUES**

Explanation: The parameter of the keyword specified in the message specifies a range using the colon operator, (for instance, 1:3), but a range of values is not allowed.

As an example, consider a keyword called CLASS. You might enter CLASS(1,2,3) to specify classes 1, 2, and 3, or you might enter CLASS(1:3). However, not all commands allow you to specify a range of values.

System action: Processing for the command is terminated.

System programmer response: Correct the keyword parameter, and reenter the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

DSN9025I   *csect-name verb-name pkw-name* **NORMAL COMPLETION FOR SYNCHRONOUS PHASE, AND IN PROGRESS FOR ASYNCHRONOUS PHASE.**

Explanation: All synchronous processing for the command specified as *verb-name pkw-name* in this message completed successfully. Any tasks executing asynchronously for the specified command might still be executing when this message is displayed.

System action: Synchronous processing for the specified command is complete.

Related information:

▶ DB2 and related commands (DB2 Commands)

DSN9026I   **KEYWORD** *keyword-name* **PARAMETER(S) MUST BE ALPHABETIC**

Explanation: Parameter values for the KEYWORD *keyword-name* must be all alphabetic characters. *keyword-name* can consist of any combination of the 26 uppercase (A through Z) and 26 lowercase (a through z) letters of the English alphabet plus the three characters reserved as alphabetic extenders for national languages ($, #, and @).

An example of this type of error, is if in the keyword NAME field, you are expected to enter a person's last name. If you enter NAME(Jones) it is acceptable, but if you enter NAME(Jon4s), it is not acceptable because it contains numeric characters.

System action: Processing for the command is terminated.

System programmer response: Reissue the command but ensure that the parameters for the named keyword are of the required type.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9027I   **KEYWORD** *keyword-name* **PARAMETER(S) MUST BE ALPHAMERIC**

Explanation: Parameter values for the keyword *keyword-name* must consist of all numbers and alphabetic characters. *keyword-name* can consist of any combination of the 26 uppercase (A through Z) and 26 lowercase (a through z) letters of the English alphabet, plus the three characters reserved as alphabetic extenders for national languages ($, #, and @), as well as numeric characters.

System action: Processing for the command is terminated.

System programmer response: Reissue the command but ensure that the parameters for the named keyword are of the required type.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9028I   **KEYWORD** *keyword-name* **PARAMETER(S) MUST BE NUMERIC**

Explanation: Parameter values for the keyword (specified on these messages) must be of a specific type, such as all ALPHABETIC or all ALPHAMERIC / NATIONAL characters, or all NUMERIC values.

As an example of this type of error, consider the keyword NAME, where you are expected to enter a person's last name. If you enter NAME(Jones) it is acceptable, but if you enter NAME(Jon4s), it is not acceptable.

As a second example, consider a keyword WEIGHT that specifies a numeric weight. Entering WEIGHT(47) is acceptable, but entering WEIGHT(4u) is not.

System action: Processing for the command is terminated.

System programmer response: Reissue the command but ensure that the parameters for the named keyword are of the required type.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9029I   *csect-name* **FAILURE WHILE PROCESSING A COMMAND**

Explanation: A failure has been encountered while processing a command. The command may or may not have been executed. The error has been recorded in the system error log (the SYS1.LOGREC data set), and an SVC dump was attempted.

System action: Processing for the command is terminated.

Operator response: If a dump was produced, have it printed. Print a copy of the contents of the SYS1.LOGREC data set for the time period involved. Forward these items with a copy of the system log to the system programmer.

System programmer response: Reissue the command. If a second failure occurs, consult your supervisor or the system programmer.

Problem determination: Obtain a copy of the system or terminal log to verify command entry. Contact the system operator to make sure that diagnostic information is forwarded to the system programmer.

DSN9030I   **KEYWORD** *keyword-name* **PARAMETER MAY NOT SPECIFY A SUBSET OF VALUES**

Explanation: The parameter for the keyword specified in the message identifies a subset using the asterisk operator (for instance, ABC*, *ABC, or *ABC*DEF*), but using a subset is not allowed for this command.

As an example, consider a keyword called NAME, for which you can specify people's names. You might specify NAME(ROBERT ROBERTS ROBERTSON) to enter three names, or you might specify NAME(ROB*) to indicate all names that begin with the characters ROB. NAME(*BERT*) could also be specified to indicate all names that contain the characters BERT within the name. However, not all commands allow you to specify a subset of values in this way.

System action: Processing for the command is terminated.

System programmer response: Correct the keyword parameter, and reenter the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

DSN9031I   **SYNTAX ERROR DETECTED FOLLOWING KEYWORD** *keyword-name*

Explanation: The text that follows the named keyword contains invalid syntax.

As an example, consider specification of CLASS==A. The double equal signs do not conform to the rules of syntax and make the keyword meaningless. This is just one example of invalid syntax. It is not practical to try to illustrate every possible error here. You could, however, correctly specify CLASS=A or CLASS(A).

System action: Processing for the command is terminated.

System programmer response: Examine the text following the named keyword to ensure that no errors were made as the command was entered. Ensure that you have followed the rules for command entry, and reenter the command.

Problem determination: Obtain a copy of the system or terminal log to verify command entry.

DSN9032I   **REQUESTED FUNCTION IS NOT AVAILABLE**

Explanation: An attempt was made to invoke a command processor that was not loaded. An example of this is invoking -START DDF or -STOP DDF when DDF is not loaded (not selected during the installation process).

System action: The requested function is not performed.

User response: Determine what function was requested. Do not invoke this function until DB2 is reinstalled and DDF is selected.

Problem determination: Look at the console log to determine which command caused the error.

DSN9033I   **KEYWORD** *name1* **REQUIRES KEYWORD** *name2*

Explanation: When keyword 'name1' is specified, keyword 'name2' must also be specified.

System action: Processing for the command terminates.

User response: Resubmit the command with the missing keyword.

Problem determination: Obtain a copy of the system or terminal log to verify the command entry.

DSN9034I   **THE TOTAL LENGTH OF COMMAND RESPONSE MESSAGES EXCEEDED ITS LIMIT. DISPLAY IS TERMINATED.**

Explanation: The total length of all response messages returned from a single DB2 command exceeded the internal limit set by the system. The message chain was truncated. This message was appended at the end of the chain to inform the user of the termination of the report.

DSN9035I   **BEGIN OF DISPLAY FOR MEMBER:** *memname*

Explanation: This message precedes the command output from another member of a data sharing group as the result of the SCOPE(GROUP) keyword. The message number is not included in this message.

memname
      Member name for which command output is being displayed.

System action: Processing continues normally.

DSN9036I

Explanation: This message underscores message DSN9035I and separates the command output from another member of a data sharing group as the result of the SCOPE(GROUP) keyword. The message number is not included in this message.

System action: Processing continues normally.

DSN9037I   **END OF DISPLAY FOR MEMBER:** *memname*

Explanation: This message follows messages DSN9035I, DSN9036I, and the command output from another member of a data sharing group as the result of the SCOPE(GROUP) keyword and indicates that the output was generated successfully. The message number is not included in this message.

memname
      Member name for which command output is being displayed.

System action: Processing continues normally.

DSN9038I   *csect-name* **COMMAND REJECTED BECAUSE DB2 WAS RESTARTED IN LIGHT MODE**

Explanation: This command is not allowed when the DB2 member has been restarted with the LIGHT(YES) parameter.

System action: Command processing is halted.

System programmer response: Refrain from issuing this command on this DB2 member until after the member has been restarted without LIGHT(YES). If the command has group scope, then you may be able to issue the command to a different DB2 member in the group.

Appendix A. New, changed, and deleted messages

Messages are added to, changed in, and removed from DB2 11 for z/OS, as necessary.

Lists of new, changed, and deleted messages are available exclusively in the online version of this information.

New, changed, and deleted messages

Appendix B. z/OS diagnostic services

DB2 uses functional recovery routines that run MVS and z/OS services to provide diagnostic information for assistance in problem determination.

Functional recovery routines used by DB2 run MVS and z/OS services to provide diagnostic information for assistance in problem determination. The following MVS services provide diagnostic information:

SVC dumps

The DB2 abend completion code X'04E' uses the MVS SDUMP service to create SVC dumps. The content and storage areas associated with these dumps vary depending on the specific error and the state of the DB2 subsystem at the time the error.

SYS1.LOGREC entries

Entries are requested in the SYS1.LOGREC data set at the time of the error using the MVS SETRP service. The following are also recorded in SYS1.LOGREC:

- Subsystem abnormal terminations
- Secondary abends that occur in a recovery routine
- Recording requests from recovery routines percolated to by the recovery termination manager.

Variable recording area (VRA) data

Data entries are added to the VRA of the SDWA by using an MVS VRA defined key. VRA data includes a series of diagnostic data entries common to both DB2 abend completion codes. Additional information is provided by the invoking subcomponent recovery routine during initial error processing or recovery termination manager percolation.

During DB2 operation, an abend can occur with an MVS system completion code.

If you receive an MVS abend during DB2 operation, refer to the appropriate MVS publication.

Appendix C. Resource types

Type codes are associated with particular types of resources.

The following table describes the type of resource associated with the type codes. The abbreviations in the "Name, Content, Format" column are explained in Table 4 on page 1156.

Table 3. Resource Types

TYPE Code	Type of Resource	Name, Content, Format
00000100	Database	DB
00000200	Table space	DB.SP
00000201	Index space	DB.SP
00000202	Table space	RD.DB.TS
00000205	Compression Dictionary	DB.SP
00000210	Partition	DB.SP.PT
00000220	Data set	DSN
00000230	Temporary file	SZ
00000240	Database procedure	DBP
00000300	Page	DB.SP.PG
00000301	Index minipage	DB.SP.PG.MP
00000302	Table space page	DB.SP.PG
00000303	Index space page	DB.SP.PG
00000304	Table space RID	DB.SP.RID
00000305	Index access/table space RID	DB.SP.RID
00000306	Index access/table space page	DB.SP.PG
00000307	Index space EOF	DB.SP.01
00000400	ICF catalog	IC
00000401	Authorization function	
00000402	Security Server	SAF/RACF return/reason codes
00000500	Storage group	SG
00000602	EDM DBD Space	
00000603	EDM DYNAMIC STATEMENT Space	
00000604	EDM skeleton storage	
00000605	EDM above-the-bar storage	
00000606	EDM below-the-bar storage	
00000700	Buffer pool space	BP
00000701	Group buffer pool	GBP
00000800	Plan	PL

Table 3. Resource Types (continued)

TYPE Code	Type of Resource	Name, Content, Format
00000801	Package	COLLECTION. PACKAGE. CONTOKEN
00000802	BINDLOCK01 through BINDLOCK20	BINDLOCK01 through BINDLOCK20
00000900	32KB data area	
00000901	Sort storage	
00000903	Hash anchor	DB.SP.PG.AI
00000904	RIDLIST storage	
00000905	IRLM storage	
00000906	DB2	MEMBER
00000907	LOB storage	
00000908	Basic Floating Point Extensions Facility	
00000909	Extended Time-of-Day (TOD) Clock	
0000090A	XML storage	
00000A00	Table	RD.CR.TB
00000A10	Alias	RELDEP. OWNER. ALIAS. RD.CR.AL
00000A11	Distinct type	SC.DT
00000A12	User-defined function	SC.SN
00000A13	Stored procedure	SC.SN
00000A14	Sequence	
00000A16	Role	
00000A17	Trigger	
00000B00	View	RD.CR.VW
00000C00	Index	RD.CR.IX
00000C01	Index	CR.IX
00000D00	DBID/OBID	RD.DI.OI
00000D01	DBID/OBID	DI.OI
00000D02	OBID	OI
00000E00	SU limit exceeded	CN
00000F00	Auxiliary column	DI.OI. ROWID. COLN or DI.OI. DOCID. COLN
00000F01	LOB lock	DIX.PIX. ROWID. VRSN
00000F81	XML lock	DIX.PIX. DOCID
00001000	DDF	LOCATION or SUBSYSTEM ID

Table 3. Resource Types (continued)

TYPE Code	Type of Resource	Name, Content, Format
00001001	System conversation	LU.MODE. RTNCD. FDBK2. RCPRI. RCSEC. SENSE
00001002	Agent conversation	LU.MODE. RTNCD. FDBK2. RCPRI. RCSEC. SENSE
00001003	CNOS processing	LU. MODE. RTNCD. FDBK2. RCPRI. RCSEC. SENSE
00001004	CDB (Communication database)	LOCATION. AUTHORIZATION ID. PL
00001005	DB access agent	LOCATION
00001007	TCP/IP domain name	LINKNAME. DOMAIN. ERRNO
00001008	TCP/IP service name	LOCATION. SERVICE. ERRNO
00001080	ACCEL	SERVER.DOMAIN
00001102	Bootstrap data set (BSDS)	MEMBER
00002000	Table space CS-claim class	DB.SP
00002001	Table space RR-claim class	DB.SP
00002002	Table space write-claim class	DB.SP
00002003	Index space CS-claim class	DB.SP
00002004	Index space RR-claim class	DB.SP
00002005	Index space write-claim class	DB.SP
00002006	Table space partition CS-claim class	DB.SP.PT
00002007	Table space partition RR-claim class	DB.SP.PT
00002008	Table space partition write-claim class	DB.SP.PT
00002009	Index space partition CS-claim class	DB.SP.PT
00002010	Index space partition RR-claim class	DB.SP.PT
00002011	Index space partition Write-claim class	DB.SP.PT
00002100	Table space DBET entry	DB.SP
00002101	Index space DBET entry	DB.SP
00002102	Table space partition DBET entry	DB.SP.PT
00002103	Index space partition DBET entry	DB.SP.PT
00002104	DBET hash chain lock timeout	INTERNAL LOCK NN
00002105	Logical partition DBET entry	DB.SP.PT
00002200	Routine parameter storage	DBP
00002201	Debug agent storage	DBP
00002300	ICSF encryption and decryption facilities	
00003000	Code (release maintenance_level or system parameter)	REL, APAR, ZPARM

Table 3. Resource Types (continued)

TYPE Code	Type of Resource	Name, Content, Format
00003002	Number of Stored Procedures	
00003072	Index	
00003073	Index	
00003328	Release dependency	
00003329	DBID/OBID	DI.OI
00003330	OBID limit exceeded	
00003840	LOB column	
00004000	Profile exception threshold exceeded	PID.PTYPE.PNAME

The following table contains the definitions for the acronyms provided for the "Name, Content, Format" column of the preceding table, Table 3 on page 1153.

Table 4. Definitions for resource type name, content, and format

Where	Stands for
AI	Hash anchor ID
ALIAS	Alias owner
APAR	APAR number
AUTHORIZATION ID	DB2 authorization identifier
BP	Buffer pool identifier
CN	Column name
COLLECTION	Collection-ID of the package
COLN	Column number within the base table of the LOB column that has been marked invalid.
CONTOKEN	Consistency token of the package
CR	Creator of the object
DB[1]	Database name
DBP	Database procedure name
DI	DBID in decimal of resource
DIX	DBID in hexadecimal
DOMAIN	TCP/IP domain name or IP address
DSN	Data set name
FDBK2	VTAM secondary return code
GBP	Group buffer pool name
IC	ICF catalog alias name
IX	Index name
LOCATION	Location in which the specified resource is not available
LU	Logical unit name
MEMBER	Group member name
MODE	Logical unit mode name
MP	Hexadecimal mini-page number

Table 4. Definitions for resource type name, content, and format (continued)

Where	Stands for
NN	Decimal number of the hash chain
OI	OBID in decimal of resource
OWNER	Alias owner
PACKAGE	Package identifier
PG	Hexadecimal page number
PID	Profile ID
PIX	PSID in hexadecimal
PL	Plan identifier
PNAME	The name of the profile filtering scope
PT	Decimal partition number
PTYPE	The type of the profile filtering scope
RCPRI	APPC primary return code
RCSEC	APPC secondary return code
RD[2]	DB2 release dependency mark
REL	DB2 release name
RELDEP	DB2 release dependency mark
RID	Record Identifier
RTNCD	VTAM primary return code
SENSE	SNA sense codename
SERVER	Accelerator server name
SG	Storage group name
SP	Space name
SZ	Temporary file page size

Table 4. Definitions for resource type name, content, and format (continued)

Where	Stands for
ST	Space allocation type in work file database. This type is used when the value of the WFDBSEP subsystem parameter is YES. When this type (ST) is used with resource type 00000100 and reason code 00C90084, message DSNT501I specifies the missing resource name as `DB2-MANAGED SPACE WITHOUT SECONDARY ALLOCATION OR USER-MANAGED SPACE`. This resource name implies either an segmented (non-universal) table space that is managed by DB2 and has a SECQTY value of 0 or any user-managed table space. When this type (ST) is used with resource type 00000200 and reason code 00E7009A, SQLCODE -904 specifies the missing resource name as `DB2-MANAGED SPACE WITH SECONDARY ALLOCATION`. This resource name implies either a partition-by-growth table space regardless of the SECQTY value or a table space that is segmented (non-universal) with a non-zero value for SECQTY. **Related information:** SEPARATE WORK FILES field (WFDBSEP subsystem parameter) (DB2 Installation and Migration) "DSNT501I" on page 556 -904 (DB2 Codes)
TB	Table name
TS	Table space name
VW	View name

Notes:

1. DB can also refer to DI or ST. SP, TS, and TB can also refer to OI. TS can also refer to ST. DB2 sometimes does not have access to resource names, and uses DBID or OBID instead.
2. RD can be represented as a white space or blank to indicate no release dependency, or Y or N to indicate whether it is IBMREQD.

Information resources for DB2 for z/OS and related products

Information about DB2 for z/OS and products that you might use in conjunction with DB2 for z/OS is available in online information centers or on library websites.

Obtaining DB2 for z/OS publications

The current DB2 for z/OS publications are available from the following website:

http://pic.dhe.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db2z11.doc/src/alltoc/db2z_lib.htm

Links to the information center version and the PDF version of each publication are provided.

DB2 for z/OS publications are also available for download from the IBM Publications Center (http://www.ibm.com/shop/publications/order).

In addition, books for DB2 for z/OS are available on a CD-ROM that is included with your product shipment:

* DB2 11 for z/OS Licensed Library Collection, LK5T-8882, in English. The CD-ROM contains the collection of books for DB2 11 for z/OS in PDF format. Periodically, IBM refreshes the books on subsequent editions of this CD-ROM.

Installable information center

You can download or order an installable version of the Information Management Software for z/OS Solutions Information Center, which includes information about DB2 for z/OS, QMF, IMS, and many DB2 and IMS Tools products. You can install this information center on a local system or on an intranet server. For more information, see http://pic.dhe.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.dzic.doc/installabledzic.htm.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY   10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

This information is intended to help you interpret messages that are issued by DB2 11 for z/OS. This information also documents General-use Programming Interface and Associated Guidance Information provided by DB2 11 for z/OS.

General-use Programming Interface and Associated Guidance Information

General-use Programming Interfaces allow the customer to write programs that obtain the services of DB2 11 for z/OS.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered marks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at http://www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM

Software Products and Software-as-a-Service Privacy Statement" at
http://www.ibm.com/software/info/product-privacy.

Glossary

The glossary is available in the Information Management Software for z/OS Solutions Information Center.

See the Glossary topic for definitions of DB2 for z/OS terms.

IBM®

Product Number: 5615-DB2
5697-P43

Printed in USA

Spine information:

IBM

Messages

DB2 11 for z/OS