# Table of Contents

- **List of Figures**: ................................................................. 7
- **List of Tables**: ............................................................... 9
- **Preface**: ............................................................................ 11
  - Adobe Portable Document Format ........................................ 14
  - Candle Customer Service and Satisfaction ............................ 16
- **What's New**: ................................................................. 19

**Chapter 1.** *Background about Components and Modes of Operation* .............................................. 25
  - Product Components .......................................................... 26
  - Details about the Candle Subsystem ....................................... 28
  - Details about the CUA Interface ............................................. 30
  - Details about the Common Interface ....................................... 31
  - Details about the Historical Components ............................... 32
  - Modes of Operation ............................................................ 35

**Chapter 2.** *Configuring and Customizing OMEGAMON II for DB2* ...................................................... 41
  - Configuration Planning and Considerations ............................ 42
  - Overview of the Process ....................................................... 46
  - CICAT Background and Requirements .................................... 47
  - CICAT Configuration Procedures .......................................... 48
  - Manual Configuration Procedures ........................................ 54
  - Manual Customization Procedures ........................................ 55

**Chapter 3.** *Completing the Configuration* .......................................................... 57
  - APF-Authorizing the Libraries ............................................... 58
  - Completing the Configuration of the Classic Interface ............. 59
  - Completing the Configuration for ISPF ................................. 61
  - Completing the Configuration for ISPF and TSO .................... 64
  - Completing the Configuration for the CUA Interface ............... 65
  - Completing the Configuration of the D2-TE interface ............. 67
  - Completing the Configuration Outside of CICAT .................... 69
  - Determining What to Do Next ............................................. 70

**Chapter 4.** *Verifying the Configuration* .......................................................... 71
  - Starting the Common Interface Address Space ....................... 72
  - Starting OMEGAMON II ....................................................... 73
  - Starting OMEGAMON in TSO or ISPF Mode ......................... 74
  - Starting OMEGAMON II in Dedicated Mode ........................... 75
Table of Contents

Historical Online Collector Commands ................................................. 163
Event Collection Manager Parameters ................................................. 165

Appendix B. Dataset Authorization Requirements ................................. 167
Dataset Authorization Requirements ..................................................... 168

Appendix C. Configuration and Customization of CLIST Output ............... 171
Job Tailoring .................................................................................... 172

Appendix D. Guidelines for Sizing VSAM Datasets ................................. 177
VSAM Dataset Allocation Guidelines .................................................... 178

Appendix E. Sharing VTPOOL in a Multi-Host Environment ................. 181
Sharing VTPOOL ............................................................................ 182

Appendix F. Parameters in KD2SYSIN .................................................. 185
Parameters in KD2SYSIN ................................................................. 186

Index ................................................................................................. 193
<table>
<thead>
<tr>
<th>FIGURE</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGURE 1</td>
<td>OMEGAMON II for DB2 Logical Connections</td>
<td>27</td>
</tr>
<tr>
<td>FIGURE 2</td>
<td>Relationship of Components in VTAM, TSO/ISPF, and Dedicated Modes</td>
<td>36</td>
</tr>
<tr>
<td>FIGURE 3</td>
<td>Example of Security Update Job</td>
<td>135</td>
</tr>
<tr>
<td>FIGURE 4</td>
<td>Typical Security Update Program Header</td>
<td>136</td>
</tr>
<tr>
<td>FIGURE 5</td>
<td>Typical Control Statement Edit Listing</td>
<td>137</td>
</tr>
<tr>
<td>FIGURE 6</td>
<td>Typical Security File Listing</td>
<td>139</td>
</tr>
<tr>
<td>FIGURE 7</td>
<td>Typical Security Update Program Trace</td>
<td>140</td>
</tr>
<tr>
<td>FIGURE 8</td>
<td>Common Interface Facility</td>
<td>154</td>
</tr>
<tr>
<td>FIGURE 9</td>
<td>Sample Network</td>
<td>182</td>
</tr>
<tr>
<td>FIGURE 10</td>
<td>VTPOOL Definition Statement</td>
<td>183</td>
</tr>
<tr>
<td>FIGURE 11</td>
<td>HSAA VTAM Definition Statements</td>
<td>183</td>
</tr>
<tr>
<td>FIGURE 12</td>
<td>HSAB VTAM Definition Statements</td>
<td>184</td>
</tr>
</tbody>
</table>
List of Tables

Table 1. Product Components for OMEGAMON II for DB2 .............................................26
Table 2. Characteristics and Requirements for OMEGAMON II Modes of Operation ..........37
Table 3. Overview of the Process .................................................................46
Table 4. CICAT Configuration Procedure Checklist .................................................51
Table 5. Manual Configuration Procedure Checklist ..................................................54
Table 6. Manual Customization Procedures Checklist ................................................55
Table 7. (Page 1 of 2). Session Options and Default Settings .......................................100
Table 8. Exception Names and Default Thresholds .....................................................102
Table 9. Security Control Statements .................................................................124
Table 10. CUA Interface Security Choices ...........................................................145
Table 11. IF Command Test Values .................................................................162
Table 12. Tailored Parameters in RKD2PAR .........................................................172
Table 13. Tailored Jobs in RKD2SAM ...............................................................173
Table 14. Tailored Parameters in RKANCMD .......................................................176
Table 15. VSAM Dataset Allocation Guidelines ......................................................178
About this document

This guide describes how to configure and customize OMEGAMON II® for DB2, Version 540. It assumes that you have already installed the product as described in the Installation and Configuration of Candle Products on OS/390 and z/OS manual.

This guide contains the following types of information to help you plan and perform configuration and customization:

- a list of product publications
- background about the product components
- considerations that you need to review before you configure or customize
- an overview of the installation, configuration, and customization process
- configuration and customization instructions

Who should read this book

This guide is intended for users of OMEGAMON II for DB2, including product administrators and system programmers. It is a hands-on guide that provides the information you need to configure OMEGAMON II for your site and quickly start monitoring your DB2 network.
Notation conventions

This document uses the following conventions when referring to high-level qualifiers.

- **hilev**: A high-level qualifier. The high-level qualifier is the first prefix or set of prefixes in the dataset name. This document refers to site-specific high-level qualifiers in italics.
  
  For example,
  - *thilev* refers to the high-level qualifier for your target dataset.
  - *rhilev* refers to the high-level qualifier for your runtime dataset. (For members in target libraries, the high-level qualifier is *thilev* rather than *rhilev*.)
  - *shilev* refers to the SMP/E library high-level qualifier.

- **-HILEV-**: This notation refers to the high-level qualifier in specific JCL. Replace this with the appropriate high-level qualifier that is specific to your site.

The mid-level qualifier is used to distinguish multiple copies of the runtime environment. This qualifier allows a unique name to be created for each copy while allowing for a common high-level qualifier. Together these two qualifiers are described as *rhilev.midlev*.

Gathering documentation

If OMEGAMON II for DB2 malfunctions, please do the following before you contact Candle Customer Service and Satisfaction:

- Obtain Common Interface joblog and syslog information related to the problem.
- Record output from the DEBUG screen, when possible.

  To obtain these DEBUG screens, type DEBUG on the INFO-line and press Enter. OMEGAMON executes various screen and logs important information to hardcopy.
- Gather any dumps that OMEGAMON generated (on tape or fiche). OMEGAMON may have generated dumps to one or more of the following:
  - the Common Interface SYSABEND DD statement
  - a SYS1.DUMPcc dataset
  - the RKO2SNAP dataset

OMEGAMON may snap diagnostics to the RKO2SNAP dataset, using the MVS SNAP macro. Normally the dataset is empty. The snap dataset is a sequential dataset. You can print it using IEBGENER. To clear the snap dataset, use the JCL in *rhilev.RKD2SAM(KO2JSNAP)*.

Candle Customer Service and Satisfaction personnel may request any or all of the above information to assist with problem resolution.
Online documentation

With V520, Candle Corporation moved OMEGAMON II for DB2 manuals from IBM BookMaster to Adobe FrameMaker. This move was made to better enable us to address our customers’ needs by providing tools that enhance productivity.

One of the results of the move is that it is no longer possible to create BookManager versions of the OMEGAMON II for DB2 manuals. However, the manuals remain available online in the Adobe PDF version on CD-ROM and are also available on the Candle Corporation website at www.Candle.com.

Where to look for information

For more information related to this product, please see the

- technical documentation CD-ROM that came with your product
- technical documentation information available on the Candle Web site at www.candle.com
- online help provided with this product
Adobe Portable Document Format

Printing this book

Candle supplies documentation in the Adobe Portable Document Format (PDF). The Adobe Acrobat Reader will print PDF documents with the fonts, formatting, and graphics in the original document. To print a Candle document, do the following:

1. Specify the print options for your system. From the Acrobat Reader Menu bar, select File > Page Setup… and make your selections. A setting of 300 dpi is highly recommended as is duplex printing if your printer supports this option.

2. To start printing, select File > Print… on the Acrobat Reader Menu bar.

3. On the Print pop-up, select one of the Print Range options for
   - All
   - Current page
   - Pages from: [ ] to: [ ]

4. (Optional). Select the Shrink to Fit option if you need to fit oversize pages to the paper size currently loaded on your printer.

Printing problems?

The print quality of your output is ultimately determined by your printer. Sometimes printing problems can occur. If you experience printing problems, potential areas to check are:

- settings for your printer and printer driver. (The dpi settings for both your driver and printer should be the same. A setting of 300 dpi is recommended.)
- the printer driver you are using. (You may need a different printer driver or the Universal Printer driver from Adobe. This free printer driver is available at www.adobe.com.)
- the halftone/graphics color adjustment for printing color on black and white printers (check the printer properties under Start > Settings > Printer). For more information, see the online help for the Acrobat Reader.
- the amount of available memory in your printer. (Insufficient memory can cause a document or graphics to fail to print.)

For additional information on printing problems, refer to the documentation for your printer or contact your printer manufacturer.

Contacting Adobe

If additional information is needed about Adobe Acrobat Reader or printing problems, see the Readme.pdf file that ships with Adobe Acrobat Reader or contact Adobe at www.adobe.com.
Adding annotations to PDF files

If you have purchased the Adobe Acrobat application, you can add annotations to Candle documentation in .PDF format. See the Adobe product for instructions on using the Acrobat annotations tool and its features.
Background

To assist you in making effective use of our products, Candle offers a variety of easy-to-use online support resources. The Candle Web site provides direct links to a variety of support tools that include these services:

- **eSupport** allows you to create and update service requests opened with Customer Service and Satisfaction (CSS).
- **eDelivery** allows you to download products, documentation, and maintenance 24 hours a day, 7 days a week.
- **eNotification** notifies you of product updates and new releases.

In addition, you can find information about training, maintenance plans, consulting and services, and other useful support resources. Refer to the Candle Web site at [www.candle.com](http://www.candle.com) for detailed customer service information.

Candle Customer Service and Satisfaction contacts

You will find the most current information about how to contact Candle CSS by telephone or email on the Candle Web site. Go to the [www.candle.com](http://www.candle.com) support section, and choose the link to Support Contacts to locate your regional support center.
Introduction

This section identifies the enhancements for the Version 540 release of OMEGAMON II for DB2. The changes and additions in OMEGAMON II for DB2 Version 540 provide new function in addition to reflecting support for features in IBM DB2 Version 8.1. OMEGAMON II for DB2 Version 540 enables you to obtain additional information about the performance of your systems. This additional information allows you more flexibility in managing various types of threads. An overview of each new or changed function follows.

Note: With this release, OMEGAMON II for DB2 no longer supports IBM DB2 Version 5.

Historical Reporter redesign

With OMEGAMON II for DB2, significant enhancements were made to the historical reporting component. Some of these new features are:

- Sequential output files are produced from the online collector.
- All record types can be loaded into DB2 tables using either the LOAD utility or using SQL INSERT.
- Near term history collector for displays is configured separately from long term history for reports.
- New fields have been added to the Summarizer.

New IFCID-based reports

- Numerous new IFCID-based reports are provided. See the OMEGAMON II for DB2 Historical Reporting Guide, Version 540, for a list of the supported IFCIDs. See the README provided with this product for instructions on generating the IFCID-based reports.
- You can generate your own reports, based on the IFCIDS selected, using the product-provided COBOL, C, and SAS record layouts. All instrumentation record types (IFCID) are supported. The data are extracted from SMF, GTF or from the Online Collector input.
64-bit addressing support
Version 8 of DB2 UDB for z/OS supports 64-bit virtual storage. The zSeries 64-bit architecture allows DB2 UDB for z/OS to move various storage areas above the 2-GB bar. OMEGAMON II for DB2 Version 540 now provides the ability to display information from above the 2-GB bar structures.

64-bit addressing support applies to the:
- EDM pool
- buffer pools
- sort pools
- RID pools

Long name support
OMEGAMON II for DB2 Version 540 includes support for long object names. New displays are provided in support of this feature.

Unicode support
Unicode support includes character conversion from Unicode to EBCDIC for online displays and batch reports.

Enclave support
New displays for stored procedures using Work Load Manager (WLM) enclaves include:
- Enclave Token
- Service Class Period
- Performance Index
- Service Class
- WLM Environment Name

Stored procedure Address Spaces is being phased out in DB2 Version 8. All stored procedures now use WLM.

New filtering for Near Term History display
These keywords are added to the Near Term History display:
- COMMIT
- CORRID
- DB2 CPU TIME
- DB2 ELAP TIME
- DEADLK/TIMEOUT
- I/O ELAP TIME
- GETPAGES
- LOCK ESCAL
- PACKAGE

See the OMEGAMON II for DB2 Reference Manual, Version 540, for descriptions of these new fields.

**New DSNZPARM fields**

The DSNZPARM Thread Parameters panel now includes numerous new fields.

**Distributed Thread display improvements**

The Distributed Thread display improvements include the addition of these fields:
- Host Name
- Workstation identifier
- End user ID on workstation
- Transaction ID on workstation
Section 1.
Before You Begin
introduction

This chapter contains information about the components and modes of operation for OMEGAMON II for DB2. The chapter provides background information about the

- product components
- details about the CUA interface
- details about the Common Interface (CI)
- details about the Candle Subsystem
- details about the historical components
- modes of operation

Chapter Contents

Product Components .................................................. 26
Details about the CUA Interface .................................. 30
Details about the Common Interface .............................. 31
Details about the Historical Components ....................... 32
Modes of Operation .................................................. 35
Product Components

This section provides background information about the product components for OMEGAMON II for DB2.

Product components for OMEGAMON II for DB2

The following table lists the components available when you install OMEGAMON II for DB2, provides a brief description of each component, and indicates whether the component is required or optional.

Table 1. Product Components for OMEGAMON II for DB2

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classic interface for the real-time monitor (required)</td>
<td>Provides real-time information about a DB2 subsystem using the original OMEGAMON II menu system interface</td>
</tr>
<tr>
<td>Common Interface (CI) (required)</td>
<td>Provides common services to the OMEGAMON II for DB2 family of products</td>
</tr>
<tr>
<td>CUA interface for the real-time monitor (optional)</td>
<td>Provides real-time information about a DB2 subsystem using a graphical user interface (This component is required to use the !DB/EXPLAIN interface.)</td>
</tr>
<tr>
<td>Candle Subsystem (optional)</td>
<td>Provides dynamic I/O information to OMEGAMON</td>
</tr>
</tbody>
</table>
| ISPF interface (optional for the classic interface, but required for the online historical component, and !DB/Tools) | Provides an ISPF interface for these functions
- classic interface
- online historical component (This does not include using historical in batch.)
- !DB/Tools |
| DB/EXPLAIN interface (optional) | Provides access to the DB/EXPLAIN interface |
| Event Manager and Application Trace Facility (ATF) (optional) | Event Manager provides a collection environment for the object and volume analysis collectors.
ATF lets you dynamically allocate trace datasets using characteristics that you specify in the CUA interface or at install time. |
| Historical components (optional) | Gather and report historical information about a DB2 subsystem |
| !DB/Tools (optional) | Provides access to the !DB/Tools |
Process for components that are optional
When you install OMEGAMON II for DB2 using CICAT, CICAT automatically installs the components that are optional. For example, CICAT automatically installs the Candle Subsystem. To make these components available, you must also:

- configure the component using CICAT
- complete the configuration and customization steps for the component using the instructions in this guide (if any).

Overview of the components in the configuration
The configuration process defines the correct address space controls and logical VTAM connections necessary to run OMEGAMON II. The following figure shows those interconnections.

FIGURE 1. OMEGAMON II for DB2 Logical Connections
Details about the Candle Subsystem

Introduction

This section provides background information about the Candle Subsystem.

Candle Subsystem component

The Candle Subsystem is an MVS subsystem that enables OMEGAMON II to monitor dynamic device activity in MVS/ESA™ SP4 and higher.

When installed, the Candle Subsystem runs in its own address space, providing dynamic I/O device information to OMEGAMONs running in other address spaces.

Sharing the Candle Subsystem

You only need one Candle Subsystem for an MVS system image. A single Candle Subsystem can support multiple copies of OMEGAMON II and multiple OMEGAMON II products on a single MVS image.

The subsystem ID identifies a copy of the Candle Subsystem. To use the same Candle Subsystem for all OMEGAMON II runtime environments on a single MVS image, Candle recommends that you specify the same subsystem ID during the configuration of each OMEGAMON II product. The Candle default subsystem ID is CNDL.

System requirements

The Candle Subsystem has the following system requirements:

- The Candle Subsystem requires 4K of ECSA.
- The Candle Subsystem must be defined to MVS as a subsystem.
- The initialization module, KCNDLINT, must reside in a link list authorized library.
- An IPL is required to initialize the Candle Subsystem unless the keyword parameter form of the IEFSSNxx PARMLIB member, supported in MVS/SP 5 or above, is used. In this case, MVS dynamic SSI services can be invoked.

Note: The example for IEFSSNxx, a positional parameter, works as shown on OS/390 systems. If you would like to update SYS1.PARMLIB(IEFSSNxx) in accordance with IBM OS/390 documentation, then use the following syntax:

```
SUBSYS SUBNAME(CNDL)
INITRTN(KCNDLINT)
INITPARM('SSPROC=CANSCLN')
```
Starting the Candle Subsystem automatically

Member rhilev.RKANSAM(CANSCN) contains a sample JCL procedure for creating the Candle Subsystem. You can modify this to fit your configuration standards, and then copy it to a system procedure library.

If you want the Candle Subsystem address space to be started automatically at system IPL, then the name given to the JCL procedure must match the value of the SSPROC keyword in the IEFSSNxx member of SYS1.PARMLIB.

Determining whether or not to install the Candle Subsystem

You should migrate from your current Candle Subsystem to the current version shipped with V540. This will ensure that new maintenance gets properly installed.

If you have installed another OMEGAMON II product at your site, at the same level as shipped with the OMEGAMON II product you are currently installing, you may have already installed the Candle Subsystem.
Details about the CUA Interface

This section provides background about the component for the CUA interface.

CL/Engine and the CUA Interface

The CUA interface uses the services of CL/Engine, a host-based network software system. CL/Engine provides a base for advanced network management, integration, and interface capabilities in multiple-host SNA environments.

CL/Engine facilities include the dialog manager, the network access manager (NAM), the network accounting facility (NAF), the operator facility, the virtual terminal pool facility, and the virtual printer facility.

Note: CL/Engine is also referred to as CT/Engine.
Details about the Common Interface

This section provides background about the component for the Common Interface.

Common Interface component

The Common Interface, also known as CI, provides a centralized runtime environment that allows multiple OMEGAMON II components to execute in one MVS address space. You communicate with the CI to start and stop OMEGAMON II components, such as the online data collector or the realtime monitor. The CI accepts commands from its parameter library (OMPARM DD statement), or through the MVS operator’s console.

When you start the CI, it automatically executes a series of commands contained in a member of the parameter library. This member has the same name as the CI’s started task name. The member’s commands automatically start one or more components of OMEGAMON II.

For more information about the CI and CI commands, see “The Common Interface” on page 153.
Details about the Historical Components

This section provides background about the historical components.

Historical components

OMEGAMON II provides historical components that enable you to collect, extract and summarize historical information about the performance of your DB2 subsystem, as well as generate reports. These components include:

- online data collector
- data extractor
- data summarizer
- OMEGAMON II reporter

You can store the gathered historical data in VSAM datasets, in DB2 tables, or in sequential files. The OMEGAMON II reporter generates reports and graphs from any storage media. OMEGAMON II also provides sample reports in SAS® and QMF™. It also provides record layouts in SAS, COBOL, and C.

These historical component features are optional. You can choose to customize them along with the realtime monitor, or customize them at a later time.

For more information about specific historical features, see the OMEGAMON II for DB2 Historical Reporting Guide.
Online data collector

The online data collector gathers statistical and accounting information (including distributed database information), audit information, performance information, and DSNZPARM information from the DB2 subsystem and stores it in VSAM datasets, in DB2 tables, or in sequential datasets as activity occurs. Performance information is stored only in VSAM datasets.

You can access this information immediately. Data stored in the VSAM datasets must be archived to a sequential dataset as each VSAM dataset becomes full.

The online data collector also provides the realtime monitor with near-term historical data, which includes:

- Near-term statistics history
  
  The realtime monitor can display up to 96 intervals of DB2 statistics. Trace data can also be displayed when the online data collector is collecting DB2 statistics traces.

- Near-term thread history
  
  Thread history information is available when:
  - the historical collector is active
  - data is collected to VSAM
  - accounting data is available in the active datasets

  Thread history is not available when data is collected to DB2 tables.

If the DB2 subsystem terminates while historical collection is active, historical collection automatically reconnects to the DB2 subsystem when the subsystem is active again.

Data extractor

The data extractor enables you to extract historical DSNZPARM, statistical, accounting, performance, and audit data from SMF, GTF, or the online collector dataset. You can store the data in DB2 tables or sequential datasets. You may want to perform data extraction when you do not require immediate access to historical data, or when a system outage interrupts the collector, but DB2 continues to run.

Data summarizer

The data summarizer enables you to consolidate detailed statistical and accounting data from DB2 tables or sequential datasets into a summarized format. This allows you to save historical data and generate reports over long periods of time without using prohibitive amounts of storage space. You can also copy data from one type of storage to another.

IFCID-based reports

Using the IFCID-based product-provided reports as samples, you can generate your own reports. These reports will be based on the IFCIDs you selected while using the Extractor. For instructions on generating these types of reports, see the README for this product.
Details about the Historical Components

**OMEGAMON II reporter**

The Historical Reporter generates reports from DB2 tables, or from SMF and GTF datasets (via the extractor). It is a self-contained feature that operates in ISPF or batch mode.

The reporter includes statistics, accounting, audit, DSNZPARM, and performance reports.

**COBOL, C or C++, and SAS**

Sample SAS, COBOL and C record formats are also provided. The samples can be used as examples to develop your own reports. You can create your own reports based on the IFCIDs selected while using the Extractor.

**Historical reporter traces**

Historical reporting depends entirely on data supplied by traces. Keep in mind the following:

- The online data collector automatically turns on traces based on the options you select when you start it.
- You can customize your DSNZPARM parameters to start and externalize certain traces automatically at DB2 startup time.

To determine which traces you must activate in order to report on specific data elements, see the *OMEGAMON II for DB2 Historical Reporting Guide*.

**Access to sort program libraries**

OMEGAMON II invokes sorting functions for various online and batch historical reporting functions.

*Note:* If your installation does not include sort libraries in the MVS link list, you may have to manually update the sample tailored JCL to include the appropriate libraries. You may also need to include these libraries in your ISPF load library concatenation before using the online historical information screens. Additionally, OMEGAMON II must have access to the UNICODE conversion routines (DB2 Version 8 only).
Modes of Operation

This section provides background about the operating modes.

Operating modes available

During configuration, you will be asked to select and customize an operating mode. Available operating modes are:

- VTAM mode
- TSO/ISPF mode
- dedicated mode

VTAM mode is required to run the CUA interface. You can also configure additional operating modes for the classic interface, as described in “Starting OMEGAMON in TSO or ISPF Mode” on page 74 and the online help available when you configure the classic interface.
Illustration showing the relationship for the modes of operation

The following graphic shows the relationship of OMEGAMON II components in VTAM, TSO/ISPF, and dedicated modes.

FIGURE 2. Relationship of Components in VTAM, TSO/ISPF, and Dedicated Modes
Operating mode characteristics and requirements

The following table describes each operating mode and its requirements.

**Table 2. Characteristics and Requirements for OMEGAMON II Modes of Operation**

<table>
<thead>
<tr>
<th>Mode</th>
<th>Characteristics</th>
<th>Configuration Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>VTAM</td>
<td>VTAM mode enables you to run OMEGAMON II sessions from a VTAM terminal without an intermediate online application, such as TSO. You can set automatic update mode so that the screen refreshes automatically. VTAM mode allows all VTAM terminal users to share a single copy of OMEGAMON II. <strong>Note:</strong> Type README APP on the command line to get more information on how CICAT processes VTAM APPLIDs. Use the F6=Applids key to specify the VTAM major node and APPLID values. <strong>Note:</strong> If System Variable support is enabled, type README SYS on the command line to obtain more information on how CICAT processes VTAM APPLIDs using MVS system symbols.</td>
<td>Define a VTAM applid for OBVTAM.</td>
</tr>
</tbody>
</table>
| TSO and ISPF  | The TSO address space communicates with the OMEGAMON II address space via a VTAM application, VTM1. In this mode there is no auto screen refresh; the screen refreshes when you press the Enter key. TSO mode enables you to access OMEGAMON II without logging off TSO. ISPF mode includes split-screen capability that lets you swap between multiple OMEGAMON II sessions, or between OMEGAMON II and another ISPF application. | ▪ Define a VTAM applid for OBVTAM.  
▪ Requires an active OBVTAM application.  
▪ Define a set of virtual terminals to VTAM. You can define up to 99 virtual terminals in the virtual terminal pool (VTPOOL). |
| Dedicated     | Dedicated mode offers high availability and does not require VTAM services. Dedicated mode uses EXCP to communicate with a terminal and refreshes the screen every few seconds. Dedicated mode allows OMEGAMON II to provide realtime data even when VTAM is not available. | Availability of a locally attached non-SNA terminal. |
Section 2.
Configuration and Customization
Introduction

This chapter provides information about installing, configuring, and customizing OMEGAMON II for DB2.

This chapter provides:

- the considerations you should review before you begin to configure and customize OMEGAMON II for DB2
- a broad overview of the installation, configuration, and customization process (as well as where you can locate the information you will need)
- background about the Candle Installation and Configuration Assistance Tool (CICAT)
- an overview of how you install OMEGAMON II for DB2 using CICAT
- an overview of how you configure OMEGAMON II for DB2 using CICAT and a checklist listing the steps for the CICAT configuration procedure
- a checklist listing the steps for the manual configuration procedures
- a checklist listing the steps for the manual customization procedures

If you are installing OMEGAMON II for DB2 for the first time or you need a reminder about the different components and modes of operation for OMEGAMON II for DB2, see the chapter “Background about Components and Modes of Operation” on page 25.

Chapter Contents

CICAT Background and Requirements ............................................ 47
CICAT Configuration Procedures .................................................... 48
Manual Configuration Procedures .................................................. 54
Manual Customization Procedures .................................................. 55
Configuration Planning and Considerations

This section provides the considerations that you must review before you begin to configure and customize OMEGAMON II for DB2.

Requirements for hardware and software

For information about hardware and software requirements, see Installation and Configuration of Candle Products on OS/390 and z/OS.

Installing OMEGAMON II and OMEGAVIEW in separate CSIs

Candle Corporation strongly recommends that OMEGAMON II and OMEGAVIEW be installed in a shared CSI. However, if you decide to install each of these products in a separate CSI, then you must ensure that the started task JCL for the OMEGAMON II realtime monitor interfaces includes the rhilev.RKANMODL library for OMEGAVIEW as part of the STEPLIB and RKANMODL DD concatenations.

Specifying dispatching priority

To ensure availability, execute the OMEGAMON II Common Interface with a higher dispatching priority than the DB2 subsystem being monitored or any address space attached to the DB2 subsystem.

Note: OMEGAMON II will attempt to set its dispatching priority one level higher than the DB2 system service address space dispatching priority. Running OMEGAMON II at a lower dispatching priority than the DB2 subsystem can seriously degrade overall system performance.

Requirements for the system

OMEGAMON II must run on the same MVS system as the DB2 subsystem you are monitoring.

CI started task name

The configuration and customization CLIST will create the automatic startup member with the same name as the CI started task name. If you manually change the CI started task name, you must also change the automatic startup member name to match the new started task name.
Requirements for traces

In order to fully exploit OMEGAMON II’s realtime performance monitoring abilities, you must activate various accounting traces, depending on the type of data you wish to monitor:

- **thread level information**
  
  To obtain thread level information, activate accounting trace classes 1, 2 and 3.

- **package level information**
  
  To obtain package level information, activate accounting trace classes 7 and 8.

See the *OMEGAMON II for DB2 Reference Manual* for more information about DB2 accounting traces.

Historical considerations

You have these choices for configuring the historical components.

- For DB2 subsystems in different CIs, you can run a component in each of the CIs.
- For DB2 subsystems in the same CI, you can run a component for each DB2 subsystem in the CI.

**Important**

You must follow one of the following recommended configurations in order to display near-term history information in the realtime monitor.

Migrating classic screen spaces and profile members

During configuration of the classic interface, you can migrate any screen spaces and profile members that were created in previous versions. The migration procedure is described in step 8 on page 59.

Migrating CUA profiles

During configuration of the CUA interface, you will be able to migrate existing CUA profiles to Version 540. The migration procedure is described in step 5 on page 66.

Migrating a classic profile to CUA

To migrate a classic interface profile to a CUA subsystem, you must log on directly to the CUA interface and access the Logon Options panel. You cannot migrate a classic profile to CUA if you are logged on through OMEGAVIEW.

A classic profile should be migrated to a CUA subsystem only once per profile. You should set up a CUA subsystem profile for every DB2 subsystem you are monitoring, using the DB2 subsystem identifier as the profile name.

For more information about using the CUA Logon Options panel, see the *OMEGAMON II for DB2 User’s Guide*. 
Migrating VSAM datasets

OMEGAMON II for DB2 Version 540 uses a VSAM linear dataset (LDS) as its collection datastore. All OMEGAMON II historical utilities use this datastore for collection and reporting functions. All VSAM historical files must be archived.

You can use your existing Version 510 (and higher) VSAM LDS datasets, or you can create new VSAM LDS datasets for Version 540. The VSAM LDS dataset is also required in order to collect near-term thread history information. You do not have to initialize the LDS dataset.

Security considerations

The following table lists the elements you can migrate from prior versions of OMEGAMON II for DB2.

<table>
<thead>
<tr>
<th>Item</th>
<th>Security Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal security</td>
<td>The product is shipped with a security system for the classic interface and a security system for the CUA to prevent unauthorized use. No security is the default. For information, see “Customizing Security for the Realtime Monitor” on page 109.</td>
</tr>
<tr>
<td>External security</td>
<td>You can set up an interface between OMEGAMON II for DB2 and an external security package, such as RACF or CA-ACF2. For information, see “Configuring and Customizing OMEGAMON II for DB2” on page 41.</td>
</tr>
</tbody>
</table>
| Started task | OMEGAMON II attaches to DB2 in the historical collector, the Application Trace Facility, and the DB/EXPLAIN interface. When installing OMEGAMON II, you must authorize to DB2 the OMEGAMON II started task’s authorization identifier and the D2-TE started task. In most installations, this authorization identifier can be one of the following:
  - a default userid for all started tasks
  - the first 7 characters of the OMEGAMON II started task name
Make sure that your security package has authorized the OMEGAMON II started task identifier. During configuration, OMEGAMON II generates security jobs KO2ACF2A, KO2RACFA, and KO2SUPD, which create exit programs for external security packages. These members are in rhilev.RKD2SAM. |
<p>| Datasets     | OMEGAMON II for DB2 requires certain levels of dataset authorization for the Common Interface, the CUA interface, and users. For information about dataset authorization requirements, see “Dataset Authorization Requirements” on page 167. |</p>
<table>
<thead>
<tr>
<th>Item</th>
<th>Security Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Trace Facility (ATF)</td>
<td>OMEGAMON II for DB2 secures ATF from unauthorized use by means of the security mechanism you have defined for your OMEGAMON II classic interface component.</td>
</tr>
</tbody>
</table>
Overview of the Process

This section provides a broad overview of the installation, configuration, and customization process. It also includes information about accessing help when using CICAT.

Broad overview of the process

The following table contains the broad steps you follow when you install, configure, and customize OMEGAMON II for DB2. The table also shows where you can find the information you will need during each of the steps.

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Using CICAT, install the product and create any new runtime environments.</td>
<td>Installation and Configuration of Candle Products on OS/390 and z/OS and the online help for the product panel you are using</td>
</tr>
<tr>
<td>2</td>
<td>Using CICAT, configure the components you want to use.</td>
<td>Online help for the product panel you are using</td>
</tr>
<tr>
<td>4</td>
<td>Manually configure the components and verify that the configuration is complete.</td>
<td>Chapters 3 and 4 in this guide</td>
</tr>
<tr>
<td>5</td>
<td>Manually customize the components you want to use.</td>
<td>Chapters 5, 6, and 7 in this guide</td>
</tr>
</tbody>
</table>

Getting help with CICAT

The help for CICAT contains detailed information about using the CICAT panels. For example, the help contains information about:

- how to use the panel
- why parameters are required
- what the available action codes provide
- what the input fields mean
- what you are required to supply

To display help from any CICAT panel, press the Help key (F1) or enter HELP on the command line.

You can also display help for the help. For example, you can display information about the command to use to return to the previous topic in the help system. To display the help for help from any help panel, press the Help key (F1) or enter HELP on the command line.
CICAT Background and Requirements

This section describes using the Candle Installation and Configuration Assistance Tool (CICAT).

You must use CICAT to install and configure OMEGAMON II for DB2. CICAT is an ISPF dialog that guides you through the installation and configuration steps required to install this product. Data entry panels assist you in understanding your site-specific parameter values. Associated help panels assist you in understanding the CICAT process and describe the input fields on the entry panels.

CICAT is restartable. If necessary, you can end the dialog, start it again, and continue from the point of interruption. ISPF V2.3 or above is required to use CICAT.

If you have not previously installed CICAT during installation of this or any other Candle product, you must do so now. For instructions on installing CICAT, see the Installation and Configuration of Candle Products on OS/390 and z/OS manual. If you want to use CICAT from a previous installation, you must ensure that it is the most current version of CICAT. The Installation and Configuration of Candle Products on OS/390 and z/OS manual will help you make this determination.

Restrictions on specifying values in CICAT

Important Note: Entering ampersand (&) in any CICAT parameter string, whether you are in interactive or batch mode, results in a CICAT abend.

Reminder about the information available

If you need information about installing OMEGAMON II for DB2 using CICAT, you can locate information in the

- Installation and Configuration of Candle Products on OS/390 and z/OS manual
- online help for the product panel you are using

Examples of the tasks performed by CICAT

CICAT performs tasks that make OMEGAMON II for DB2 operational with a basic set of defaults. You use CICAT to:

- install the SMP/E datasets
- modify the JCL
- allocate datasets
- define VTAM applids
- modify the classic interface security command
- create runtime libraries
- install the Candle Subsystem
- run the migration utility
CICAT Configuration Procedures

This section describes the CICAT configuration procedures for OMEGAMON II for DB2.

Prerequisites for configuring OMEGAMON II for DB2

Before you start to configure OMEGAMON II for DB2, be sure that you have reviewed the considerations and planning information in the section “Configuration Planning and Considerations” on page 42.

The following configuration procedures assume that you have completed SMP/E installation and applied maintenance for OMEGAMON II for DB2, as described in the Installation and Configuration of Candle Products on OS/390 and z/OS manual.

Reminder about the information available

If you need information about configuring OMEGAMON II for DB2 using CICAT or specific information about the values you specify using CICAT, see the online help for the product panel you are using.

Resources available in CICAT

When you configure and customize OMEGAMON II for DB2 using CICAT, CICAT creates members that contain parameters and sample jobs for each of the components.

The $NDX$NDX member is an index that lists the members CICAT creates for components such as:

- event collection manager
- CUA interface
- historical component

The members listed in $NDX$NDX member follow the naming convention $NDX(component$).

If you want to display the index that CICAT creates, see the $NDX$NDX members in:

- rhilev.midlev.RKD2PAR
- rhilev.midlev.RKD2SAM

Accessing the Configure OMEGAMON II for DB2 menu in CICAT

There are two versions of the CICAT Installer available to install and configure Candle products. These versions include:

- CICAT Version 200
- CICAT Version 300

The method you use to access the Configure OMEGAMON II for DB2 Menu in CICAT will vary depending on the version of CICAT you are using.
Accessing the menu in CICAT Version 200

To begin OMEGAMON II for DB2 configuration:

1. Start CICAT. (For a reminder, see your Installation and Configuration of Candle Products on OS/390 and z/OS manual.)

2. On the CICAT Main Menu:
   - If you installed the MultiProduct Quick Install tape, select MultiProduct Quick Install.
     To preview the list of products included in your MultiProduct Quick Install tape, you can use action code V (View Additional Information) on MultiProduct Quick Install.
   - If you installed OMEGAMON II for DB2 as a separate product, select it.

3. On the Installation/Configuration Primary Menu, select Set up configuration environment.

4. On the Runtime Environments panel, use action code C (Configure) on the RTE you are ready to configure.

5. If you installed the MultiProduct Quick Install tape or a multicomponent product, select OMEGAMON II for DB2 on the Product Configuration Selection Menu.

6. Proceed to use the Configure OMEGAMON II for DB2 Menu.

Accessing the menu in CICAT Version 300

To begin OMEGAMON II for DB2 configuration:

1. Start CICAT. (For a reminder, see your Installation and Configuration of Candle Products on OS/390 and z/OS manual.)

2. On the Main Menu, select Configure products.

3. On the Configure Products menu, select Set up configuration environment.

4. On the Configuration Environment Information panel, specify the values.

5. Return to the Configure Products menu.

6. On the Configure Products menu, select Select product to configure.
   Result: CICAT displays the Product Selection menu.

7. On the Product Selection menu, select OMEGAMON II for DB2.
   Result: CICAT displays the Runtime Environments (RTE) panel.
Example of the Configure OMEGAMON II for DB2 menu in CICAT

The following figure is an example of the Configure OMEGAMON II for DB2 menu.

```
---------------- CONFIGURE OMEGAMON II FOR DB2 / RTE: PDSIRA ------------
OPTION ===>

Perform these configuration steps in order:                     Last selected
1 Specify configuration values             Date     Time
2 Allocate additional runtime datasets
3 Create runtime members
4 Modify Classic interface command security
5 Complete the configuration

Optional:
6 Install Candle Subsystem
7 Run migration utility

F1=Help   F3=Back
```
**CICAT configuration checklist**

The following table contains the steps you perform on the CICAT Configure OMEGAMON II for DB2 menu. The steps are listed in the sequence in which they are to be performed. Use the ✔ column to check off steps as you complete them.

**Table 4. CICAT Configuration Procedure Checklist**

<table>
<thead>
<tr>
<th>✔</th>
<th>CICAT Configuration Step</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Use <strong>Specify configuration values</strong> to:</td>
</tr>
<tr>
<td></td>
<td>- configure the realtime monitor interfaces, including the Classic, CUA, and DB/EXPLAIN interfaces</td>
</tr>
<tr>
<td></td>
<td>- select a DB2 subsystem, or specify and configure a DB2 subsystem (as well as the Event Manager and historical components for that subsystem)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Both options must be configured prior to creating the runtime members.</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Allocate additional runtime datasets</strong> to review the JCL that CICAT generates to allocate the datasets for the historical component.</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Create runtime members</strong> to:</td>
</tr>
<tr>
<td></td>
<td>- review the JCL that CICAT generates to create the members for the interfaces for the realtime monitor</td>
</tr>
<tr>
<td></td>
<td>- select the DB2 IDs that you want CICAT to install in the RTE and review the JCL that CICAT generates</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Modify Classic interface command security</strong> to customize the security exit, and to install security information into each runtime environment that requires Classic interface security.</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Complete the configuration</strong> to perform the additional steps required to complete the configuration outside of CICAT.</td>
</tr>
<tr>
<td></td>
<td>If you want to install a Candle subsystem, specify the values using <strong>Install Candle Subsystem</strong>. (This step is optional and is not required.)</td>
</tr>
<tr>
<td></td>
<td>Use <strong>Run migration utility</strong> to migrate tables and datasets from a previous version of OMEGAMON II for DB2 (Version 510 and above). (This step is optional and is not required.)</td>
</tr>
<tr>
<td></td>
<td>Load the runtime libraries using <strong>When to load the runtime libraries</strong> on page 52.</td>
</tr>
</tbody>
</table>
Migrating from an earlier version of OMEGAMON II for DB2

The Run migration utility option lets you migrate elements that you have already configured in a previous version of OMEGAMON II for DB2 (Version 510 and above).

When you use this option, you must specify the complete RTE high-level qualifiers you used for the libraries in the previous version.

**Important:** The complete RTE high-level qualifiers consist of both the high-level qualifier and the mid-level qualifier used for the earlier version.

For example, you might have a Non-VSAM high-level qualifier, a VSAM high-level qualifier, and a mid-level qualifier as your complete non-VSAM and VSAM high-level qualifiers.

The values are used to migrate these elements:
- Product-level security
- Internal tables database
- Screen spaces and menus
- Profiles and exception thresholds

*Note:* This step only applies if you are migrating from another installation library. If you upgraded the same installation library to Version 540, you may skip this step.

When to load the runtime libraries

You use action code L (Load Libs after SMP/E) on the Runtime Environments (RTEs) panel to populate the load libraries for a selected RTE. This action code upgrades your RTE to the latest Candle maintenance level. Use action code L at the following points in the CICAT process:
- After you install and configure the products you want in a new RTE.
- After you install and configure an additional product into an existing RTE.
- After you apply additional Candle maintenance.

When you defined or updated your RTE, you had the option to selectively load from the target to the runtime libraries only those members that changed.

If you requested Load Optimization, the load job generated when you use action code L (Load):
- Copies only modified modules.
- Requires access to IBM’s SuperC (ISRSUPC) utility.
- Uses less DASD space.
- Performs additional analysis which uses more CPU and I/O.
If you bypass Load Optimization, the load job:

- Copies all members.
- Requires more DASD space.
- Uses less CPU time.
Manual Configuration Procedures

This section provides information about performing manual configuration procedures for OMEGAMON II for DB2.

Reminder about the information available

The checklist in the following table contains the location where you can find the information you will need.

Manual configuration checklist

The following table contains the steps you perform manually to configure OMEGAMON II for DB2. The steps are listed in the sequence in which they are to be performed. Use the ✔ column to check off steps as you complete them.

Table 5. Manual Configuration Procedure Checklist

<table>
<thead>
<tr>
<th>✔</th>
<th>Manual Configuration Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔</td>
<td>Complete the configuration using “Completing the Configuration” on page 57.</td>
</tr>
<tr>
<td>✔</td>
<td>Verify configuration of the realtime monitor using “Verifying the Configuration” on page 71.</td>
</tr>
</tbody>
</table>
Manual Customization Procedures

This section provides information about performing manual customization procedures for OMEGAMON II for DB2.

Reminder about the information available

The checklist in the following table contains the location where you can find the information you will need.

Manual customization checklist

The following table contains the steps you perform manually to customize OMEGAMON II for DB2. The steps are listed in the sequence in which they are to be performed. Use the ✔ column to check off steps as you complete them. Candle recommends that you review the entire process before you begin customizing OMEGAMON II for DB2.

Table 6. Manual Customization Procedures Checklist

<table>
<thead>
<tr>
<th>✔</th>
<th>Manual Customization Step</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>If you configured ATF using CICAT, complete the customization and verify operation using “Completing ATF Customization” on page 87.</td>
</tr>
<tr>
<td></td>
<td>If you configured the historical components using CICAT, complete the customization using “Completing Historical Customization” on page 90.</td>
</tr>
<tr>
<td></td>
<td>If you configured the historical components using CICAT, start the historical collector and verify operation using “Starting and Stopping the Historical Data Collector” on page 93.</td>
</tr>
<tr>
<td></td>
<td>Create an installation-defined profile using “Customizing Realtime Monitor Profiles” on page 97.</td>
</tr>
<tr>
<td></td>
<td>Select and customize a security system for the realtime monitor using “Configuring and Customizing OMEGAMON II for DB2” on page 41.</td>
</tr>
</tbody>
</table>
Manual Customization Procedures
Chapter Overview

This chapter provides a broad overview of the manual configuration process and the manual configuration procedures for these components.

- classic interface
- ISPF interface
- CUA interface
- OMEGAMON II for DB2 to !DB/EXPLAIN interface

The chapter also covers the actions you must take if you reconfigure the components in an existing runtime environment.

Chapter Contents

- APF-Authorizing the Libraries .................................................. 58
- Completing the Configuration of the Classic Interface .................. 59
- Completing the Configuration for ISPF .................................... 61
- Completing the Configuration for ISPF and TSO ........................ 64
- Completing the Configuration for the CUA Interface .................... 65
- Completing the Configuration of the D2-TE interface .................. 67
- Completing the Configuration Outside of CICAT ........................ 69
- Determining What to Do Next .................................................. 70
APF-Authorizing the Libraries

This section provides information about APF-authorization of load libraries. The load libraries must be APF-authorized or some of the functions in the product may not be available.

You APF-authorize the load libraries by adding them to your list of APF-authorized libraries.

APF authorizing the load libraries

You must APF-authorize one or more load libraries, depending on whether or not you are running the CUA interface.

Classic interface

For the classic interface, you must APF-authorize the load library rhilev.RKANMOD.

If one library in a steplib or joblib concatenation is required to be APF-authorized, you must APF-authorize all libraries in the concatenation or all libraries will lose their APF status.

CUA interface

For the CUA interface, you must APF-authorize the following load libraries:

- rhilev.RKANMODL
- rhilev.RKANMODL for OMEGAVIEW (required only if you plan to connect the D2 CUA interface to OMEGAVIEW and they are running in separate CIs)

Note: Any runtime libraries concatenated in the STEPLIB DDNAME and in the RKANMODL DDNAME of the started task must be APF-authorized.
Completing the Configuration of the Classic Interface

This section provides you with the final steps for completing the configuration of the classic interface.

Completing configuration of the classic interface

You must perform the following steps to complete the configuration of the classic interface:

1. Make sure that you APF-authorized load library *rhilev.RKANMOD*.

2. Copy the JCL for the started task to the JES procedure library for the started task. (The member has the started task name you specified for OMEGAMON II for DB2 using CICAT.)

   The configuration CLIST generated a member that contains this JCL in *rhilev.RKD2SAM*. Copy this member into a procedure library that is searched in response to an operating system START command. (The member name was specified for Started Task Name on the OMEGAMON II for DB2 Classic Interface Information panel.)

3. Provide access to the DB2 SYSUTIL dataset.

   If ACF2 or Top Secret is used as a security package, the CI must have READ access to the SYSUTIL dataset. If RACF is used, the CI must have control access to the SYSUTIL dataset. The SYSUTIL dataset name is *cccccccc.DSNDBC.DSNDB01.SYSUTILX.I0001.A001*, where *cccccccc* is the DB2 catalog name. (Although control access is required in the case of RACF, OMEGAMON II for DB2 does not update the SYSUTIL dataset’s contents).

4. Copy the VTAM major node definition member for the OBVTAM program from *rhilev.RKD2SAM* to SYS1.VTAMLST. The VTAM major node name was specified on the Realtime Monitor Classic Interface panel.

5. Review the member that contains the automatic start command and remove the comment character (*) for the components you want to start automatically when the started task starts. (The member is in *rhilev.RKD2PAR* and has the started task name you specified for OMEGAMON II for DB2 using CICAT.)

6. For TSO mode only, edit the *rhilev.TKANCLI(KO2TSO)* to modify the default input parameters for your installation.

   **Note:** If you made copies of this CLIST library, update the library that you are currently using.

7. Authorize OMEGAMON II for DB2 started tasks for TCP/IP privileges.

   The distributed threads support uses a DNS lookup to translate from the TCP/IP address to display hostnames on the thread details screens. This requires RACF authorization to OMVS. You must grant the OMEGAMON II for DB2 started tasks (CID2050 and CIO2050) OMVS access in RACF so hostname translation can be performed.

8. Migrate any classic interface screen spaces and profile members that you wish to use from previous versions by copying existing members.
Completing the Configuration of the Classic Interface

- for RTEs that are Sharing with another RTE, Full, or Base, to \textit{rhilev.RKO2PROC} and \textit{RKO2PROF}
- for RTEs that are Sharing with SMP, to \textit{rhilev.TKO2PROC} and \textit{TKO2PROF}
Completing the Configuration for ISPF

This section provides information on the requirements for the ISPF interface. It also covers the methods for accessing the ISPF interface and the steps you must complete for the method you choose to use.

Space requirements for the ISPF interface
CICAT and the ISPF dialogs add members to the ISPPROF dataset. Be sure that all users who access these OMEGAMON II components have sufficient space and directory blocks in their ISPF profile datasets.

Requirements for accessing !DB/EXPLAIN
To enable access to !DB/EXPLAIN from OMEGAMON II, you must make sure that the CLIST library is allocated in the SYSPROC concatenation. For more information about !DB/EXPLAIN, see the !DB/® Tools for DB2 Installation and Customization Guide.
Methods for accessing the ISPF interface

The following table lists the methods you can use to access the ISPF interface and the steps to follow.

<table>
<thead>
<tr>
<th>Method</th>
<th>Steps</th>
<th>Required Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>If you want to access the ISPF interface by specifying the name of the dataset and member on the ISPF command line.</td>
<td>No additional actions required.</td>
</tr>
</tbody>
</table>
| B      | If you want to access the ISPF interface by specifying just the name of the command, update the SYSPROC concatenation using this method.  
  - Copy the `rhilev.RKD2SAM(KD2SPF) CLIST` to an existing library in the SYSPROC concatenation. | None (The ALTLIB statement automatically locates the appropriate CLIST library.) |
| C      | If you want access the ISPF interface by specifying just the name of the command, update the SYSPROC concatenation using this method.  
  1. Copy the `rhilev.RKD2SAM` member, `KD2SPF`, to the `rhilev.RKANCLI` dataset.  
  2. Add the `rhilev.RKANCLI` dataset to the SYSPROC concatenation via your logon procedure. | Comment out the ALTLIB statement in the KD2SPF CLIST. |

The `rhilev.RKANCLI` dataset is distributed as a fixed-block dataset. If your installation uses variable-blocked CLIST datasets, follow these steps:

1. Rename the `rhilev.RKANCLI` dataset to `rhilev.RKANCLI.FB`.
2. Allocate a new `rhilev.RKANCLI` dataset.
3. Copy the contents from `rhilev.RKANCLI.FB` to `rhilev.RKANCLI`. 
Background about the KD2SPF and KD2SPF1 CLISTs

The configuration process uses the KD2SPF and KD2SPF1 CLISTs to pass some variables from CICAT to the OMEGAMON II for DB2 ISPF interface CLIST(KD2SPF). By passing the variables this way, it allows any user to use the KD2SPF CLIST without having to know the OMEGAMON II for DB2 dataset names.

The KD2SPF1 CLIST passes the following variables from CICAT to KD2SPF:

- OMEGAMON II for DB2 version
- high-level qualifier of the target datasets
- high-level qualifier of the runtime libraries

The KD2SPF and KD2SPF1 CLISTs are created and populated into the rhilev.RKD2SAM library every time you use Create runtime members on the Configure OMEGAMON II for DB2 menu. If you use method C in the table to concatenate and then you use Create runtime members, you must concatenate again using the method C.
Completing the Configuration for ISPF and TSO

This section contains information about completing the configuration for the TSO or ISPF modes.

Follow these steps to define a virtual terminal pool:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Define VTPOOL to VTM1:</td>
<td>Load module KOBVTPL is created.</td>
</tr>
<tr>
<td></td>
<td>1. Edit the member KOBVTPL in rhilev.RKANSAM to change the sample VTPOOL definition.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Follow the instructions for assembling and linking in KOBVTPL</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Define the VTAM application IDs in SYS1.VTAMLST for the VTM1 program.</td>
<td>VTAM definition is complete.</td>
</tr>
<tr>
<td></td>
<td>Dataset member KOBVT1AP in rhilev.RKANSAM contains a sample application major node definition.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Type README APP on the command line to get more information on how CICAT processes VTAM APPLIDs. Use the F6=Applids key to specify the VTAM major node and APPLID values.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If System Variable support is enabled, type README SYS on the command line to obtain more information on how CICAT processes VTAM APPLIDs using MVS system symbols.</td>
<td></td>
</tr>
</tbody>
</table>

**Result** You have now completed configuration of the classic interface.

If you want sessions under more than one TSO or ISPF

To provide support for OMEGAMON II sessions under more than one TSO or ISPF, you must also complete the steps in the appendix “Sharing VTPOOL in a Multi-Host Environment” on page 181.
Completing the Configuration for the CUA Interface

This section covers the requirements for the CUA interface and information about the CUA user profiles. It also covers the steps you must follow to complete the configuration using one of the two address spaces available.

Requirements for the CUA interface

VTAM mode is required to run the CUA interface.

Location of the CUA information and profiles

The `rhilev.RKD2TDB` dataset contains basic CUA startup information, CUA user profiles, and CUA subsystem profiles. The first time that the CUA is started, the `rhilev.RKD2TDB` dataset is automatically initialized with basic CUA startup information. If you choose to create any customized CUA user profiles or CUA subsystem profiles, Candle highly recommends that you regularly backup the `rhilev.RKD2TDB` dataset.

Address spaces available for the CUA interface

Candle recommends running the CUA address space under a separate D2 CUA address space.
Completing the configuration using D2 CUA address space

Follow these steps to complete CUA configuration under the D2 CUA address space:

1. Be sure you have APF-authorized the `rhilev.RKANMODL`.

2. Copy the CUA interface major node definition to SYS1.VTAMLST. The CUA major node name was specified on the Realtime Monitor CUA Interface panel.

3. Copy the CUA interface started task JCL to the JES procedure library for the started task. The customization CLIST generated a member that contains this JCL in `rhilev.RKD2SAM`. The member has the name that was specified for D2 CUA Started Task Name on the Started Task and VTAM Information panel. Copy this member into a procedure library that is searched in response to an operating system START command.

4. Connect D2 CUA to OMEGAVIEW (optional).
   Edit the OMEGAVIEW started task JCL:
   For OMEGAVIEW V120, locate and remove the comment character (*) for the `&D2LEV` symbolic, and set `rhilev` to `&D2LEV` symbolic.
   For OMEGAVIEW V130, if OMEGAMON II for DB2 and OMEGAVIEW are not sharing the same libraries, verify that the `rhilev.RKANPENU`, `rhilev.RKD2PAR`, and `rhilev.RKANPAR` datasets exist in the RKANPENU DD statement concatenation of the OMEGAVIEW startup JCL. Add these datasets if they do not already exist in the concatenation.

5. Migrate CUA profiles from previous versions of OMEGAMON II for DB2. Use the IDCAMS REPRO utility to copy existing profiles to the version 540 table dataset. To perform the migration, copy the `rhilev.RKD2TDB` dataset for V270, V300, or V400 to `rvhilev.RKD2TDB` for V540.

**Result:** CUA configuration is complete.
Completing the Configuration of the D2-TE interface

This section contains the steps you must follow to complete the configuration for the OMEGAMON II for DB2 to !DB/EXPLAIN interface.

Completing the configuration of the D2-TE interface

You must perform the following steps to complete the configuration of the D2-TE interface:

1. Copy the CANSTE started task JCL to the JES library procedure library for started tasks. The configuration CLIST generated a member that contains this JCL in rhilev.RKD2SAM. This member’s name was specified for TESTC on the Classic Interface - DB/EXPLAIN panel. Copy this member into a procedure library that is searched in response to an operating system START command. Do not start this task. OMEGAMON II for DB2 starts this task on an as needed basis via an ASCRE.

2. !DB/EXPLAIN V510 PSP must be installed on every DB2 subsystem for which you want Explain statements. See the !DB/Tools for DB2 Installation and Customization Guide, Version 510 for information on installing !DB/EXPLAIN.

3. Configure DB/EXPLAIN extractid(s) to be used if they are not already configured. See Customizing DB/EXPLAIN, in the !DB/Tools for DB2 Installation and Customization Guide.

4. Define Destination Plan_Tables. The Destination PLAN_TABLE will be the current SQLID.PLAN_TABLE. If the current SQLID.PLAN_TABLE does not exist, DB/EXPLAIN will attempt to create the PLAN_TABLE in the tablespace PLANTBLIS in the !DB/TOOLS default database. If the PLAN_TABLE cannot be created there, then DB/EXPLAIN will attempt to create the PLAN_TABLE in the database DSNDB04. If it cannot create the table, DB/EXPLAIN will issue an error message indicating that the EXPLAIN could not be performed.

   The current SQLID is determined by

   - whether EXP_POWNERX is defined in the DB/TOOLS profile dataset concatenation as a non-blank value (example, EXP_POWNERX=SAMHILL). If so, the current SQLID is the value specified for EXP_POWNERX,

   - or the current SQLID is determined by the DB2 security exit.

**Note:** If you decide to configure the DB/EXPLAIN interface after initial configuration, then please ensure that you follow these steps:
Completing the Configuration of the D2-TE interface

1. Recreate the runtime members for both options:
   A. CUA and Classic interface members - generates the CITE050 started task in RKD2SAM
   B. DB2 subsystem selection/install DB2 related members - regenerates the RVTMdddd member(s) in RKD2PAR to add the DB/EXPLAIN-related parameters (where dddd = DB2 subsystem ID)

2. Refresh system procedure members (if applicable).
Completing the Configuration Outside of CICAT

Overview

After using CICAT to configure OMEGAMON II for DB2, you must perform the following procedure outside of CICAT to finalize installation and configuration.

Procedure

Perform the following steps for each of the items listed below, if applicable.

1. Copy procedures to PROCLIB.
   You must copy to your procedure library the CICAT configuration created started task procedures in RKD2SAM. Update your started task library as follows.
   
   A. If you have not already done so, copy the Candle Subsystem started task (CANSNC) from RKANSAM to PROCLIB.
   B. Copy the CUA interface started task (CANS2) from RKD2SAM to PROCLIB.
   C. Copy the Classic interface started task (CANSO2) from RKD2SAM to PROCLIB.
   D. Copy the DB/EXPLAIN interface started task (CANSTE) from RKD2SAM to PROCLIB.

2. Copy VTAM definitions to VTAMLST.
Determining What to Do Next

This section helps you to determine what to do next.

What to do if you want to create a new RTE or reconfigure an existing RTE

You must create new runtime members whenever you want to
  - create another RTE
  - reconfigure an existing RTE

If you need to create new runtime members, select option 3 (Create runtime members) from the Configure OMEGAMON II for DB2 menu.

Verifying configuration

When you have completed the steps required to configure the realtime monitor, you should verify configuration as described in “Verifying the Configuration” on page 71.
Introduction

This chapter describes how to verify configuration by starting the Common Interface (CI) and any OMEGAMON session running under it. The CI provides a centralized facility that allows multiple OMEGAMON II for DB2 sessions and online data collectors to execute in a single MVS address space.

Note: Candle recommends that you start the CI automatically when using OMEGAMON on a routine basis. By starting the CI automatically, you will not have to manually issue the MVS MODIFY commands used to start OMEGAMON. They will be issued as part of the automatic startup procedure.

Instructions for starting the CI automatically begin on page 155.

This chapter explains how to verify that you have correctly configured OMEGAMON and the CI.

Chapter Contents

Starting the Common Interface Address Space.................... 72
Starting OMEGAMON II........................................ 73
Starting OMEGAMON in TSO or ISPF Mode...................... 74
Starting OMEGAMON II in Dedicated Mode..................... 75
Logging on to the Classic Interface............................ 76
Verifying Configuration of the Classic Interface............... 78
Stopping the Classic Interface................................ 79
Starting the CUA Address Space............................ 81
Stopping the CUA interface address space................... 83
Starting the Common Interface Address Space

This section provides information on starting the Common Interface (CI) address space.

Starting the CI
To start the CI, issue the following command on the MVS operator console:

```
S cccccc
```

where `cccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

Verifying startup
When the CI is started, message CIO700 is displayed:

```
CI is ready for commands
```

*Note:* The CI is capable of running multiple OMEGAMON II’s within a single CI address space.

Starting the CI automatically
To start the CI automatically, see “Automatic Startup of the Common Interface (CI)” on page 155.

Stopping the CI
To stop the CI, issue the following command at any MVS console:

```
P cccccc
```

where `cccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

When you stop the CI, OMEGAMON also stops any active sessions. Stop OMEGAMON sessions before stopping the CI to allow normal termination and cleanup.
Starting OMEGAMON II

This section provides information on starting OMEGAMON II.

Starting the OBVTAM program

You must activate the OBVTAM program in order to run OMEGAMON II. You need to activate OBVTAM only once, regardless of how many operating modes you plan to use for OMEGAMON II.

OBVTAM starts automatically when you start the CI as described in “Automatic Startup of the Common Interface (CI)” on page 155. If you have not already started OBVTAM automatically, follow these steps:

1. Issue the following commands on the MVS operator console to activate and start OBVTAM:
   
   A. Vary OBVTAM active:
      
      \[ V \ NET,ACT,ID=cccccccc \]
      
      where cccccccc is the VTAM major node name that you specified using CICAT.
      
      Check the operator console to verify that the VARY command completed successfully.

   B. Start OBVTAM as a task under the CI. Issue the following command:
      
      \[ F \ cccccccc,EXEC RVTMcccc \]
      
      where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.
      
      Check the operator console to verify that the MODIFY command completed successfully. Look for the following OMEGAMON message: OMV001I, OMV002I.

   
   Note: If you try to start OMEGAMON and the VTAM application major node is not active, OBVTAM retries for up to 30 minutes, waiting for the application ID to be varied active.
Starting OMEGAMON in TSO or ISPF Mode

This section provides information on starting OMEGAMON II in TSO or ISPF mode.

Starting VTM1 for TSO and ISPF mode

The TSO address space communicates with the OMEGAMON II address space through a VTAM application, VTM1. TSO allows you to access OMEGAMON II without logging off TSO. ISPF mode includes split-screen capability that lets you swap between multiple OMEGAMON II sessions, or between OMEGAMON II and another ISPF application.

This procedure assumes that you have started the CI as described in “Starting the Common Interface Address Space” on page 72, and that OBVTAM is active, as described in “Starting the OBVTAM program” on page 73.

If your virtual terminal pools are not in the same major node as OBVTAM, then you must activate VTM1. Complete the following steps.

1. Activate VTM1.

To activate VTM1, you must activate the VTAM definitions for the virtual terminals. To do so, issue the following command on the MVS operator console:

   V NET,ACT,ID=cccccccc,E

   where cccccccc is the name of the VTM1 application major node of the OMEGAMON II virtual terminal interface.

2. Check the operator console to verify that the VARY command completed successfully. To do so, issue the following command on the MVS operator console:

   D NET,ACT,ID=cccccccc,E

   where cccccccc is the name of the VTM1 application major node.
Starting OMEGAMON II in Dedicated Mode

This procedure assumes that you have started the CI as described in “Starting the Common Interface Address Space” on page 72.

A dedicated session is started automatically when you start the CI as described in “Automatic Startup of the Common Interface (CI)” on page 155.

If you have not started OBVTAM automatically, perform the following procedure.

Dedicated start procedure

Follow this procedure to start a dedicated session.

1. Check the terminal.
   Ensure that the terminal you use is not an MVS console or is not allocated to any other job or started task (including VTAM). A dedicated OMEGAMON session requires the use of a locally-attached, non-SNA terminal.

2. Start a dedicated session.
   To start a dedicated session, issue the following command on the MVS operator console:
   
   ```
   F cccccc,EXEC RDEDccccc
   ```
   
   where cccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

   The MVS console log indicates that the dedicated session has started. The first panel you see on the dedicated terminal is the OMEGAMON copyright panel. The session automatically refreshes itself every few seconds.
Logging on to the Classic Interface

This section provides information about logging on to the classic interface.

In VTAM mode

To start an OMEGAMON session under VTAM, issue the following command on a VTAM terminal:

```
LOGON APPLID(aaaaaaaa) DATA('DB2=cccc,LROWS=nnn,USER=bb')
```

where values for the variables are as follows:

- `aaaaaaa` The VTAM APPLID specified to OBVTAM when it was started. This value was specified on the Realtime Monitor Classic Interface panel.
- `cccc` The subsystem ID of the DB2 subsystem to be monitored. If not specified, the default established during configuration will be used.
- `nnn` The number of logical rows. The default is 255.
- `bb` The profile suffix. The default is /C.

You can override any of your site’s default startup parameters except APPL, AUP, MODE, and TIMEOUT.

MVS console log message OMV006I indicates that the VTAM session has started.

```
Session established for aaaaaaaa/terminalid
```

The first panel you see is the OMEGAMON copyright panel. Press Enter to continue.

Logging on to a TSO/ISPF session

To start an OMEGAMON realtime session under TSO or ISPF, choose one of the following methods:

- Execute CLIST `%KD2SPF` from anywhere you normally would execute a CLIST or TSO command. Then select REALTIME from the OMEGAMON II for DB2 main menu.

- If you customized a selection for OMEGAMON II for DB2 on your site’s Primary Options Menu, select OMEGAMON II for DB2 from that menu to invoke KD2SPF, then select REALTIME from the OMEGAMON II for DB2 main menu.

If you use ISPF mode to log on, OMEGAMON displays the following panel.
Verifying the Configuration

Logging on to the Classic Interface

The default PF key settings are PF2 for the ISPF split function and PF9 for the ISPF swap function. Be sure that the return PF key is set to 00 (this is not the same as the ISPF PF4 return function.)

Provide appropriate values for the required fields. Then, type L, and press Enter to log on to the classic interface to OMEGAMON II for DB2. You can override the settings of the user profile, OBVTAM APPLID, LROWS, and the DB2 subsystem.

Result: When your logon is successful, the first screen you see is the OMEGAMON copyright screen. Press Enter to continue. OMEGAMON is now running in ISPF mode.

Note: If you do not want an ISPF environment, invoke CLIST %KO2TSO. This method provides access to the realtime functions only. The first panel you see is the OMEGAMON copyright panel. Press Enter to continue. OMEGAMON is now running in TSO mode.
Verifying Configuration of the Classic Interface

After you have performed the necessary configuration procedures described in “Chapter Contents” on page 41, and after you have started the CI as described in this chapter, you should verify configuration of the classic interface.

Verifying classic configuration

Perform the following steps to verify your configuration:

1. Log on to one of the operating modes described in the first part of this chapter.
   
   **Result:** The Candle copyright panel is displayed on your terminal (regardless of your mode of operation).

2. Press Enter from the copyright panel.
   
   **Result:** The OMEGAMON Realtime Main Menu appears.

3. Enter T on the top line.
   
   **Result:** The All Threads Connected to DB2 panel appears.

4. Enter B on the top line.
   
   **Result:** The TSO Thread Summary panel appears.

5. Press PF1.
   
   The TSO Thread Summary help panel appears.

6. Press PF1 again.
   
   **Result:** The OMEGAMON II for DB2 general help menu appears.

7. Enter D on the top line.
   
   **Result:** The Zooming help panel appears.

   
   **Result:** The OMEGAMON Realtime Main Menu appears.

9. To stop the OMEGAMON session, follow the steps in the section “Stopping the Classic Interface” on page 79.
Stopping the Classic Interface

The following sections describe how to stop an OMEGAMON II session in the classic interface.

Stopping a realtime session

To stop an OMEGAMON realtime session in any mode, press PF4 from any panel to return to the Realtime Main Menu, and then select option X and press Enter. In TSO and ISPF modes, select X again.

The OMEGAMON session stops, but OMEGAMON remains active in the system if you want to start a session later.

Stopping in dedicated mode

To stop OMEGAMON in dedicated mode, press PF4 to return to the Realtime Main Menu, and then select option X or issue the following command on the MVS operator console:

\[ F \, \text{ccccc},\text{STOP O2Unnn} \]

where cccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT and nnn is the device number.

When you stop OMEGAMON in dedicated mode, the CI remains active.

Stopping in TSO or ISPF mode

To vary VTM1 inactive, issue the following command from the MVS operator console:

\[ V \, \text{NET,INACT,I,ID=ccccc} \]

where cccccc is the major node name of the VTM1 APPLID.

When you stop OMEGAMON in TSO or ISPF mode, the CI remains active.
Stopping in VTAM mode

To stop all OMEGAMON VTAM mode sessions monitoring a specified DB2 subsystem, issue the following command on the MVS operator console:

```
F cccccccc,STOP OBVTAM
```

where `ccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

**Note:** This command also stops any active sessions running under TSO or ISPF modes.

To forcibly detach an individual session running under OBVTAM, issue the following commands from the MVS operator console:

1. To stop the task running at the secondary logical unit (or terminal), issue
   ```
   V NET,INACT,I,ID=sluname
   ```
   Wait until OBVTAM stops the task and displays the message
   ```
   NODE NOW INACTIVE
   ```

2. To make the terminal available for use again by VTAM, issue
   ```
   V NET,ACT,I,ID=sluname
   ```
   When you stop OMEGAMON in VTAM mode, the CI remains active.
Starting the CUA Address Space

This section describes how to

- start and stop the address space for the CUA interface
- log on to and off the CUA interface.

These procedures assume that you have started the CI and address space as described in “Starting the Common Interface Address Space” on page 72.

CUA address start procedure

To start the CUA interface address space, do the following:

1. Activate the D2 CUA VTAM major node by issuing MVS console operator command

   \[ V \ NET,ACT,ID=cccccccc \]

   where \( cccccccc \) is the name of the D2 CUA VTAM major node name specified during configuration of the CUA interface.

   \textbf{Note:} Check the operator console to verify that the VARY command completed successfully.

2. Issue MVS console operator command

   \[ S \ cccccc \]

   where \( ccccccc \) is the started task name you specified for the D2 CUA using CICAT.

   Monitor the console messages issued by the task. When the following message or messages appear, the CUA interface is ready for you to sign on.

   \text{KLVOP653 DIALOG APPLICATION } \text{papplid} \text{ STARTED: DIALOG(KD2ENTRY)}

   \text{KLVOP653 DIALOG APPLICATION } \text{sapplid} \text{ STARTED: DIALOG(KD2SSTR)}

   The second message appears if you are using the multisession option. The variable \( \text{papplid} \) is the primary applid, \( \text{sapplid} \) is the secondary applid.

Logging on to the CUA interface

To log on to the CUA interface directly, issue the following command from a VTAM terminal:

\[ \text{LOGON APPLID(papplid)} \]

where \( \text{papplid} \) is the primary CUA application identifier that is used to log on to the CUA interface. This applid is associated with DIALOG(KD2ENTRY), which was created when you started the D2 CUA address space.

Logging on from OMEGAVIEW

If you have created a session definition for OMEGAMON II for DB2, you can access an OMEGAMON II for DB2 session. If you have not created a session definition, refer to the OMEGAVIEW User’s Guide for assistance.
**Verifying configuration of the CUA interface**

When you log on to the CUA interface, perform the following steps to verify configuration:

1. Press Enter.
   
   **Result:** The CUA interface Sign On Panel appears.

2. Enter userid and press PF11.
   
   **Result:** The Logon Options pop-up panel appears.

3. Enter the values requested, and press Enter.
   
   **Result:** The Sign On panel appears.

4. If security is enabled, type your password, and press Enter.
   
   **Result:** The CUA interface System Status Panel appears.

5. Place the cursor in front of Active Threads and press Enter.
   
   **Result:** The All Threads Connected to DB2 appears.

   
   **Result:** The System Status panel appears.

   To stop the CUA session, follow the steps in the section “Logging off CUA” on page 82.

**Logging off CUA**

To sign off OMEGAMON II:

- Press F3 from the System Status panel.

  **Result:** The Exit Confirmation pop-up window appears.

```
Exit Confirmation
Select one of the following, then press ENTER.
_ 1. Exit from OMEGAMON II for DB2  F3
_ 2. Return to OMEGAMON II for DB2

F12=Cancel
```

Exit OMEGAMON using any of these methods:

- press Enter
- press F3
- type x or 1 in the entry field, and press Enter.
Stopping the CUA interface address space

To stop the CUA started task, issue the following MVS STOP command,

```
P cccccc
```

where cccccc is the started task name that you specified for the D2 CUA using CICAT.
Introduction

This chapter describes how to complete the customization for the event collection manager, ATF, and the historical components. This chapter also covers verifying these components.

Chapter Contents

Completing Customization of the Event Collection Manager .................. 86
Completing ATF Customization .......................................................... 87
Starting and Stopping the Event Collection Manager .......................... 88
Completing Historical Customization ............................................... 90
Accessing the Historical Main Menu ................................................. 92
Starting and Stopping the Historical Data Collector ............................ 93
Adding Another DB2 Subsystem ....................................................... 95
Completing Customization of the Event Collection Manager

When you have provided all the information requested on the CICAT panels, you must perform the following procedure to complete the customization for the event collection manager.

1. If you are running the event collection manager in a separate started task, copy the started task procedure for the event manager to the library that contains the started task procedures for the realtime monitors.

2. Review the member that contains the automatic start command for the event collection manager and remove the comment characters(*) for event manager and object analysis start commands that you want to start automatically when the event manager started task starts.

**Result:** Customization of the event collection manager is complete. You can now start and stop the event collection manager as described in “Starting and Stopping the Event Collection Manager” on page 88.
Completing ATF Customization

You must perform the following steps for every DB2 subsystem on which ATF will be used:

Note: All jobs referenced in this section are found in rhilev.RKD2SAM.

1. Give the CI access to DB2.
   If you are using an external security package (such as RACF or CA-ACF2) to control access to DB2, you must authorize the Common Interface started task user ID to connect to DB2.

2. Bind the ATF plan.
   Job ABNDcccc will bind plan KO2540AP, which is required by ATF. This plan contains an empty DBRM that is used only for Instrumentation Facility Interface (IFI) collection.

3. Grant the CI access to DB2.
   Job AGRTcccc will issue the DB2 GRANT statements that are necessary for ATF to function. This includes TRACE, MONITOR2, and EXECUTE plan privileges.

4. Create and initialize VSAM datasets.
   Job ACRTcccc will create and initialize a VSAM LDS dataset that can be used for storage of ATF data. If more than one dataset is desired, use this job as a sample to create more datasets.

Verifying ATF customization

To verify ATF customization, log on to the classic interface, and then do the following:

1. Enter A.A on the command line.
   Result: The Specify Application Trace panel appears.

2. Start an application trace using PLANNAME=ALL. Type the name of a VSAM dataset in the DSN field, and press Enter. (Use the name of one of the datasets that you allocated in step 4 on page 87)
   Result: A confirmation message is displayed.

3. Press Enter.
   Result: ATF is started.

4. Enter A.B on the top line.
   Result: The Application Trace Thread Summary panel appears.

5. Verify that a trace is active and that trace records are being collected. (Press Enter a few times to see if the Trace Records Collected field is being incremented.)
   Note: If no thread activity is taking place, no trace records will be available for collection.

6. Enter A.C to stop the trace when you have verified that ATF is functioning.
Starting and Stopping the Event Collection Manager

This section provides information on starting and stopping the event collection manager.

Starting the event manager

You must have started the Common Interface before you can start the event collection manager.

When started, EVENTMGR executes as an MVS subtask in the address space for the common interface. It provides a collection environment that is used by selected OMEGAMON II collectors. EVENTMGR performs no collection activity itself; it simply provides an execution environment that is used by other data collection mechanisms.

The event collection manager must be active before you can start object analysis collection for a selected DB2 subsystem.

Event manager startup procedure

Perform these steps to start the event collection manager:

1. Issue the following command:
   \[ F \text{cccccccc,EXEC EMGRssss} \]
   where ccccccccc is the started task name that you specified for the event collection manager using CICAT and ssss is the MVS system ID.
   This command invokes the member rhilev.RKD2PAR(EMGRssss) for the started task in which the event collection manager is to run.

2. Issue the following command:
   \[ F \text{cccccccc,EXEC sssssccc} \]
   where ccccccccc is the started task name that you specified for the event collection manager using CICAT, ssss is the MVS system ID, and cccc is the DB2 subsystem ID. This command invokes member rhilev.RKD2PAR(sssscccc) for the event manager started task to start the object and volume analysis collectors.
   Once active, EVENTMGR is a valid MVS modify ID. You can query EVENTMGR status with the following command:
   \[ F \text{cccccccc,F EVENTMGR,STATUS} \]
   where ccccccccc is the started task name you specified for the event collection manager using CICAT.

Result: A list of the collectors that are running within the event manager environment is displayed.
Stopping the event manager

To stop the event collection manager (EVENTMGR), issue the following command from the MVS operator console:

```
F cccccccc,P EVENTMGR
```

where `cccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

Upon execution, all active object analysis collectors and EVENTMGR terminate. Termination of EVENTMGR does not affect the status of the started task for OMEGAMON II for DB2 or any other tasks active within the task when the command is processed.

Verifying operation of the Event Manager

To verify operation of the event collection manager, log on to the classic interface, and then do the following:

1. Enter `O` on the command line.
   **Result:** The Object Allocation Summary panel appears. This panel lists the currently allocated databases that contain space.

2. Enter `C` on the command line of the Object Allocation Summary panel.
   **Result:** The Volume Activity Summary panel appears. This panel displays a list of volumes containing DB2 objects that are currently allocated.
Completing Historical Customization

When you have completed filling in information on the historical customization CLIST panels, you must perform the following procedure to complete customization of the historical component.

**Required customization steps**

You must perform the following steps for every DB2 subsystem on which the historical functions will be used. You must perform these steps to ensure that the DB2 plan and authorizations will support VSAM datasets or DB2 tables.

*Note:* All jobs referenced in this section, with the exception of preparing sample QMF reports, are found in `rhilev.RKD2SAM`.

<table>
<thead>
<tr>
<th>Important</th>
</tr>
</thead>
<tbody>
<tr>
<td>You must review the member that contains the automatic start command and remove the comment character (*) for the components you want to start automatically when the started task starts. This member is in <code>rhilev.RKD2PAR</code>.</td>
</tr>
</tbody>
</table>

1. **Give the CI access to DB2.**
   
   If you are using an external security package (such as RACF or CA-ACF2) to control access to DB2, you must give authorization to connect to DB2. This must be done for the Common Interface started task user ID if you are using the online collector.

2. **Create or migrate historical data storage.**
   
   If you are using DB2 tables as your storage medium, job HCRTcccc will either create new DB2 tables or migrate existing tables to the new table level. Any data that currently resides in DB2 tables will still be available after migration.

   If you are using VSAM as your storage medium, you will need VSAM log and archive datasets. The VSAM log and archive datasets are allocated for the first DB2 subsystem during the CICAT installation process. For additional DB2 subsystems, use job HCRVcccc to create the VSAM log and archive datasets. You only need to use HCRVcccc if you are customizing the historical component for an addition DB2 subsystem.

   *Note:* If you use Generation Data Group (GDG) for historical archive datasets, and you want to use a new high-level qualifier, you must run the define step of the HCRVcccc job. This step creates the model DCB for the GDG.

3. **Bind the historical plan.**
   
   Job HBNDdccc will bind plan KO2540HP. This plan is required by the historical components.

4. **Grant access to DB2.**
   
   Job HGRTcccc will issue the DB2 GRANT statements that are necessary for the historical components to function. The necessary GRANT statements depend upon your installation configuration, but may include TRACE, MONITOR2, INSERT, SELECT, DELETE, and EXECUTE plan privileges.
Completing Historical Customization

*These privileges are required by and must be granted to the CI.*

**Important**

OMEGAMON II for DB2 can monitor and collect historical data for more than one DB2 subsystem. For any given DB2 subsystem, you can record the historical data to either VSAM or DB2.

When you are collecting historical data for multiple DB2 subsystems, the CI can have multiple historical subtasks recording to any combination of DB2 and/or VSAM storage locations. However, each subtask can record to only one medium.

It is not uncommon to record real time to VSAM so that the near-term history feature is available, but then post-process the archive files to DB2 tables.

To do this, you must configure historical data collection for both VSAM and DB2 recording. *Perform the DB2 configuration first, then the VSAM configuration.* This procedure will result in the RKD2PAR(COPTxxxx) member being configured for real-time VSAM recording.

You cannot simultaneously collect and write to DB2 and VSAM if you are using a DB2 system that monitors a single system.

---

**Preparing for sample QMF reports**

If you want to store historical data in DB2 tables, you can use sample QMF reports that are provided as an alternative to the OMEGAMON II reporter.

To make the necessary procedures, forms, and queries available, issue the following command in native QMF:

```
IMPORT PROC FROM 'rhilev.TKANSAM(KO2PSIMP)'
```

Issue the following command to execute this procedure:

```
RUN PROC
```

Each of the procedures executed will prompt you to supply a value for the variable *hilev.* Supply the high-level qualifier that you used for the OMEGAMON II datasets, preceded by a single quote. For example,

```
hilev =>> 'CANDLE'
```
Accessing the Historical Main Menu

This section provides information about accessing the historical main menu.

Select an access method

To access the historical reporter, choose one of the following methods:

- Execute CLIST %KD2SPF from anywhere you normally would execute a CLIST or TSO command. Then select HISTORICAL from the OMEGAMON II for DB2 main menu.

- If you customized a selection for OMEGAMON II for DB2 on your site’s Primary Options Menu, select OMEGAMON II for DB2 from that menu to invoke KD2SPF; then select HISTORICAL from the OMEGAMON II for DB2 main menu.

Notes:

- In TSO and ISPF modes, the session is not relegated to TSO’s dispatch priority.

- If KD2SPF is invoked from the TSO READY prompt, an ISPF environment will be established. Press the split-screen key to display another OMEGAMON II for DB2 main menu instead of the ISPF/PDF Primary Option Menu.

Exiting the historical menu

To stop an OMEGAMON historical session, return to the Primary Options Menu and perform one of the following actions:

- Type X on the command line and press Enter.

- Type END on the command line and press Enter.

- Press PF3/PF15.
Starting and Stopping the Historical Data Collector

The historical data collector gathers information from DB2 as the activity occurs. You can start the collector with or without the realtime component of OMEGAMON II. The collector must be running to provide data for near-term history displays in the classic and CUA interfaces.

This procedure assumes that you have started the CI as described in “Starting the Common Interface Address Space” on page 72.

**Note:** A collector can gather data from only one DB2 subsystem. Each collector requires a separate VAM file.

**Collector startup procedure**

Perform the following steps to start a historical data collector under the CI:

1. To start a historical data collector under the CI, issue the following command:

   ```
   F cccccccc,EXEC STRTcccc
   ```

   where `cccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT and where `STRTcccc` is the historical startup member name (see Table 12: Tailored Parameters in RKD2PAR on page 172).

2. When a collector comes up, the collection specifications in effect are those specified in the data collection options member. Then you can invoke a different set of collection specifications by issuing the following command:

   ```
   F cccccccc,F H2aaaa,VARY OPTION=bbbbbbbb
   ```

   where `ccccccccc` is the started task name that you specified for OMEGAMON II for DB2 using CICAT, where `aaaa` is the ID of the DB2 subsystem you are monitoring, and where `bbbbbbbb` is the name of the member (in `rhilev.RKD2PAR`) that contains the desired collection specifications.

   If you change the specifications in the collection options member, or if you use a different member name in the above command, a new set of collection specifications will take effect after the current collection interval has expired.

   For more information about data collection, see the *OMEGAMON II for DB2 Historical Reporting Guide*.

**Stopping the data collector**

To stop the historical collector, issue the following command from the MVS operator console:

```
F cccccccc,STOP H2cccc
```

where `cccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT and where `cccc` is the ID of the DB2 subsystem you monitored.

When you stop the collector, the CI remains active.
**Verifying collector operation**

Perform the following steps to verify operation of the data collector. You must have started the historical data collector.

1. In the Classic interface for the realtime collector, enter **H.C**.
   
   **Result:** The Online Historical Collection Options panel appears.

2. Verify that the online data collector options are correct.

3. Enter **B** on the top line.
   
   **Result:** The Historical Collection Record Information panel appears.
Adding Another DB2 Subsystem

Overview

The only components of OMEGAMON II for DB2 that require additional setup when another DB2 subsystem is added to an LPAR are:

- the Online Historical Collector
- the Object Analysis component
- the Application Trace Facility

If you are not running any of the above components for this DB2 subsystem, you should be able to monitor the DB2 subsystem as soon as the DB2 subsystem is active.

The instructions for adding another DB2 subsystem running any of the above components are documented in the CICAT online help. Refer to the CICAT online help for details.
Introduction

You can customize a realtime monitor profile by selecting PROFILE from the Realtime Main Menu and changing the profile options to meet the requirements of your system. This chapter provides an overview of the available profile options. Refer to the OMEGAMON II for DB2 Reference Manual for details.

Chapter Contents

Profile Overview ................................................................. 98
Candle Profile Defaults ....................................................... 100
Creating an Installation–defined Profile ............................... 105
Implementing the Profile .................................................... 107
Profile Overview

The realtime monitor profile controls the characteristics of an active session. Both the installer and the general user can create and save customized profiles. This section describes the types of profiles, how to create an installation profile, and discusses profile security.

Types of profiles

There are three types of realtime monitor profiles:

- **The Candle-supplied profile** contains session configuration defaults and default exception analysis thresholds. It enables you to install the realtime monitor without customization and ensures that you can always initialize a session, even if no other profiles are defined.

- **The installation-defined profile** enables the installer to define default settings that are different from the Candle-supplied profile settings. You can specify this customized profile as the default for all realtime monitor sessions at your installation.

- **Individual users can create one or more user-defined profiles** to customize their individual realtime monitor sessions.

The Candle-supplied profile is always available and cannot be changed. The other profiles are optional and can exist independently of one another.

Profile suffix

Each profile has a unique two-character suffix. The suffixes for the three types of realtime monitor profiles are:

- **/C** Candle-supplied profile.
- **/I** Installation-defined profile.
- **cc** User-defined profiles—any two alphanumeric characters.

The suffix /I is assigned automatically when an installation profile is saved. The suffix for a user-defined profile is specified by the user when the profile is saved.

The profile suffix is also used to specify the desired profile on the USER parameter in your realtime monitor startup JCL or CLIST, or on the USER SUFFIX option on the ISPF logon menu. The current session’s profile suffix appears on the INFO-line next to the product version number:

```
ZMENU VTAM LOG 02 V510./I $DB2 mm/dd/yy 17:03:37 5 AB
```
Profile search order

When the realtime monitor is initialized, it loads the Candle-supplied profile, as well as the installation-defined profile and user-defined profiles if they exist. To see which profile to use, the realtime monitor checks the value on the USER start parameter.

- If /C is specified, it uses the Candle-supplied profile.
- If /I is specified, it uses the installation-defined profile. If no installation profile is found, it defaults to /C, the Candle-supplied profile.
- If a user-defined profile (cc) is specified, and the user member is not found, it searches for the /I profile. If no installation profile is found, it defaults to /C, the Candle-supplied profile.

Profile storage

The Candle-supplied profile is stored in the load library and cannot be changed. Therefore, the Candle-provided values are always available as shipped.

The realtime monitor saves the installation-defined and user-defined profiles in profile datasets referenced by the DD statement RKO2PFSV. They are read from datasets referenced by the DD statement RKO2PROF, which can be concatenated.

The installation-defined profile and the user-defined profiles are stored in the same profile datasets, and use the same naming conventions as their DD name statement.
Candle Profile Defaults

Candle ships a profile that contains session configuration options needed to initialize an OMEGAMON session. The Candle profile also contains the default exception analysis thresholds. You can override most of these defaults by creating an installation or user profile.

Session options defaults

Table 7: (Page 1 of 2). Session Options and Default Settings on page 100 lists the session options and their defaults.

Table 7. (Page 1 of 2). Session Options and Default Settings

<table>
<thead>
<tr>
<th>Screen Name</th>
<th>Option</th>
<th>Default Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Performance Options</td>
<td>NONSWAP</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>PAGEFIX</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>RESERVE</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>TSOPFIX</td>
<td>OFF</td>
</tr>
<tr>
<td>Set Display Options</td>
<td>ASF</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>BELL</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>BELLINT</td>
<td>60.00</td>
</tr>
<tr>
<td></td>
<td>DATEFORMAT</td>
<td>USA</td>
</tr>
<tr>
<td></td>
<td>FIRSTSCREEN</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>LOG</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>MINORCASE</td>
<td>LOWER</td>
</tr>
<tr>
<td></td>
<td>SCREENCASE</td>
<td>MIX</td>
</tr>
<tr>
<td></td>
<td>SCROLL</td>
<td>CSR</td>
</tr>
<tr>
<td></td>
<td>TSF</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>XLF</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>ZEROS</td>
<td>OFF</td>
</tr>
</tbody>
</table>
Table 7. (Page 1 of 2). Session Options and Default Settings

<table>
<thead>
<tr>
<th>Screen Name</th>
<th>Option</th>
<th>Default Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Display Options</td>
<td>Display</td>
<td>BASIC</td>
</tr>
<tr>
<td></td>
<td>ProfileDefinitionMode</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>ExtendedHighlighting</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>Major</td>
<td>LOW</td>
</tr>
<tr>
<td></td>
<td>Minor</td>
<td>LOW</td>
</tr>
<tr>
<td></td>
<td>Immed</td>
<td>LOW</td>
</tr>
<tr>
<td></td>
<td>Default</td>
<td>LOW</td>
</tr>
<tr>
<td></td>
<td>Clr1</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr2</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr3</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr4</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr5</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr6</td>
<td>HIGH</td>
</tr>
<tr>
<td></td>
<td>Clr7</td>
<td>HIGH</td>
</tr>
<tr>
<td>Set Control Function Options</td>
<td>FGOLIMIT</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td>FGOLoop</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>GDEVUCBS</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>INTERVAL</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>IODELAY</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>LOOPCOUNT</td>
<td>15000</td>
</tr>
<tr>
<td></td>
<td>LOOPTIME</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>OCMDDMASTER</td>
<td>ON</td>
</tr>
<tr>
<td></td>
<td>PAGELIMIT</td>
<td>400</td>
</tr>
<tr>
<td></td>
<td>PEEKSIZE</td>
<td>32768</td>
</tr>
<tr>
<td></td>
<td>STATUSMODE</td>
<td>OFF</td>
</tr>
</tbody>
</table>

For a detailed description of the session options, see the *OMEGAMON II for DB2 Reference Manual*. 
Exception thresholds defaults

Table 8: Exception Names and Default Thresholds on page 102 lists the exception names and their default states and thresholds. For a detailed description of the exceptions, see the OMEGAMON II for DB2 Reference Manual.

<table>
<thead>
<tr>
<th>Exception Name</th>
<th>Exception Group</th>
<th>Default State</th>
<th>Default Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCM</td>
<td>TH</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>ARCV</td>
<td>SY</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>BMAX</td>
<td>SY</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>BMTH</td>
<td>SY</td>
<td>ON</td>
<td>90% of buffer pool's capacity</td>
</tr>
<tr>
<td>BXPN</td>
<td>SY</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>CICT</td>
<td>CI</td>
<td>ON</td>
<td>80% of THRDMAX value</td>
</tr>
<tr>
<td>COMT</td>
<td>TH</td>
<td>ON</td>
<td>100 page updates to 1 commit</td>
</tr>
<tr>
<td>CTHD</td>
<td>TH</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>DDFS</td>
<td>SY</td>
<td>OFF</td>
<td>n/a</td>
</tr>
<tr>
<td>DRCV</td>
<td>SY</td>
<td>OFF</td>
<td>1000 kilobytes/second</td>
</tr>
<tr>
<td>DSND</td>
<td>SY</td>
<td>OFF</td>
<td>1000 kilobytes/second</td>
</tr>
<tr>
<td>DWAT</td>
<td>TH</td>
<td>OFF</td>
<td>10 seconds</td>
</tr>
<tr>
<td>EDMU</td>
<td>SY</td>
<td>ON</td>
<td>90%</td>
</tr>
<tr>
<td>ENTO</td>
<td>CI</td>
<td>ON</td>
<td>3 transactions</td>
</tr>
<tr>
<td>ENTU</td>
<td>CI</td>
<td>ON</td>
<td>95% of THRDMAX value</td>
</tr>
<tr>
<td>ENTW</td>
<td>CI</td>
<td>ON</td>
<td>2 transactions</td>
</tr>
<tr>
<td>ETIM</td>
<td>TH</td>
<td>ON</td>
<td>600 seconds</td>
</tr>
<tr>
<td>GETP</td>
<td>TH</td>
<td>ON</td>
<td>1.5 getpages to read I/Os</td>
</tr>
<tr>
<td>GTRC</td>
<td>SY</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>IDBC</td>
<td>TH</td>
<td>ON</td>
<td>0.75 seconds</td>
</tr>
<tr>
<td>IDBK</td>
<td>SY</td>
<td>ON</td>
<td>90% of IDBACK installation parameter</td>
</tr>
<tr>
<td>IDBT</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
<tr>
<td>IDF R</td>
<td>SY</td>
<td>ON</td>
<td>85% of IDFORD installation parameter</td>
</tr>
<tr>
<td>IMCN</td>
<td>IM</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>IMND</td>
<td>IM</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>INDB</td>
<td>TH</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>INDT</td>
<td>SY</td>
<td>ON</td>
<td>1 in doubt thread</td>
</tr>
<tr>
<td>Exception Name</td>
<td>Exception Group</td>
<td>Default State</td>
<td>Default Threshold</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>LKUS</td>
<td>TH</td>
<td>ON</td>
<td>80% of NUMLKUS installation parameter</td>
</tr>
<tr>
<td>LOGN</td>
<td>SY</td>
<td>ON</td>
<td>1 available log</td>
</tr>
<tr>
<td>MCNV</td>
<td>SY</td>
<td>OFF</td>
<td>85% of maximum conversations for DDF logmode entry</td>
</tr>
<tr>
<td>MDBT</td>
<td>SY</td>
<td>OFF</td>
<td>85% of MAXDBAT</td>
</tr>
<tr>
<td>MDBW</td>
<td>SY</td>
<td>OFF</td>
<td>2 threads waiting because MAXBAT was reached</td>
</tr>
<tr>
<td>MSGE</td>
<td>SY</td>
<td>ON</td>
<td>n/a</td>
</tr>
<tr>
<td>PGUP</td>
<td>TH</td>
<td>ON</td>
<td>10 updates/second</td>
</tr>
<tr>
<td>POLU</td>
<td>CI</td>
<td>ON</td>
<td>90% of allowed POOL threads</td>
</tr>
<tr>
<td>POLW</td>
<td>CI</td>
<td>ON</td>
<td>2 transactions</td>
</tr>
<tr>
<td>PREF</td>
<td>TH</td>
<td>ON</td>
<td>10 prefetch requests</td>
</tr>
<tr>
<td>RCPU</td>
<td>TH</td>
<td>OFF</td>
<td>5 seconds of CPU time</td>
</tr>
<tr>
<td>RELM</td>
<td>TH</td>
<td>ON</td>
<td>85% of resource high-water mark</td>
</tr>
<tr>
<td>RIO</td>
<td>TH</td>
<td>ON</td>
<td>10 read requests/second</td>
</tr>
<tr>
<td>SUSL</td>
<td>SY</td>
<td>ON</td>
<td>5 suspended threads</td>
</tr>
<tr>
<td>TCPU</td>
<td>TH</td>
<td>ON</td>
<td>20% of total processor utilization</td>
</tr>
<tr>
<td>THDQ</td>
<td>SY</td>
<td>ON</td>
<td>2 users</td>
</tr>
<tr>
<td>TMAX</td>
<td>SY</td>
<td>ON</td>
<td>85% of CTHREAD installation parameter</td>
</tr>
<tr>
<td>TRCV</td>
<td>TH</td>
<td>OFF</td>
<td>1000 kilobytes</td>
</tr>
<tr>
<td>TSND</td>
<td>TH</td>
<td>OFF</td>
<td>1000 kilobytes</td>
</tr>
<tr>
<td>UTIL</td>
<td>SY</td>
<td>OFF</td>
<td>N/A</td>
</tr>
<tr>
<td>VDIO</td>
<td>SY</td>
<td>ON</td>
<td>50 I/Os per second</td>
</tr>
<tr>
<td>VEDR</td>
<td>SY</td>
<td>ON</td>
<td>5 datasets</td>
</tr>
<tr>
<td>VRSP</td>
<td>SY</td>
<td>ON</td>
<td>30 milliseconds</td>
</tr>
<tr>
<td>VTIO</td>
<td>SY</td>
<td>ON</td>
<td>50 I/Os per second</td>
</tr>
<tr>
<td>VUTL</td>
<td>SY</td>
<td>ON</td>
<td>30%</td>
</tr>
<tr>
<td>WCLM</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
<tr>
<td>WDLK</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
<tr>
<td>WGLK</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
<tr>
<td>WLGQ</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
</tbody>
</table>
### Table 8. Exception Names and Default Thresholds

<table>
<thead>
<tr>
<th>Exception Name</th>
<th>Exception Group</th>
<th>Default State</th>
<th>Default Threshold</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSPS</td>
<td>TH</td>
<td>ON</td>
<td>60 seconds</td>
</tr>
<tr>
<td>WSRV</td>
<td>TH</td>
<td>ON</td>
<td>30 seconds</td>
</tr>
<tr>
<td>WTRE</td>
<td>TH</td>
<td>ON</td>
<td>100 seconds</td>
</tr>
</tbody>
</table>
Creating an Installation–defined Profile

You can change some or all of the Candle-supplied profile defaults to customize the realtime monitor for your system. By modifying selected options and thresholds and saving them, you can create an installation-defined profile. Then you can specify this profile as the default for your system.

The customization options for creating a user-defined profile are the same as those described below for the installation-defined profile. Only the profile save screen is different.

Customizing screen spaces

OMEGAMON provides the /SAVE and /REP commands to allow you to create and save your own realtime monitor screens. The use of these commands is documented in the OMEGAMON II for DB2 Reference Manual.

Establishing default options

Run the realtime monitor with the Candle-supplied profile to become familiar with basic realtime monitor session and the exception analysis options. Determine what default options are appropriate for your installation and what changes are required to create your installation profile.

See “Candle Profile Defaults” on page 100 for a list of the Candle profile defaults.
Selecting session and exception analysis options

Most of the session options you can choose to create the installation-defined profile also apply to creating a user-defined profile. You can change the Candle default settings and/or add options that are appropriate for your installation. Please note, some realtime monitor performance options can be set only in the installation-defined profile.

Follow these steps to display the screens that allow you to select the session options and exception settings for the installation profile.

1. Choose PROFILE from the Realtime Main Menu to display the Profile Maintenance Menu.

2. From the Profile Maintenance Menu, choose option A or B as follows:
   - Option A accesses the screens containing session options. These include the performance options that apply to the installation-defined profile.
   - Option B accesses the screens that allow you to set each exception threshold and other exception attributes.

The OMEGAMON II for DB2 Reference Manual describes each of these screens in detail.

Note: You may want to restrict access to the Set Performance Options screen to prevent general access to the installation profile. (See “Step 2: Secure the Installation Screens” on page 108 for information about protecting the Set Performance Options screen from unauthorized use.)
Implementing the Profile

Implementing the profile consists of saving it and specifying it as the default profile.

Step 1: Saving the installation profile

You can change the setting of any installation-defined profile option at any time during an OMEGAMON session. OMEGAMON uses the changed setting for the duration of the current session (except for the IOPT command, where only the RESERVE parameter takes effect immediately).

Return to the Profile Maintenance Menu and select the “Save Install” option. OMEGAMON displays the Save New/Altered Installation Profile screen.

OMEGAMON automatically assigns the /I suffix to profiles saved with this screen.

**Important**

The saved profile picks up not only the settings you just changed, but all current profile definitions.

To delete the installation-defined profile, return to the Profile Maintenance Menu and select the “Delete Install” option. OMEGAMON displays the Delete New/Altered Installation Profile screen.
Step 2: Secure the Installation Screens

For the purpose of discussing OMEGAMON’s security feature, this section refers to the particular OMEGAMON command rather than to the screen.

The commands specific to the installation-defined profile, IPRF and IOPT, are shipped unsecured so that you can easily install and start OMEGAMON. However, if you create an installation-defined profile, you may want to protect it from inadvertent damage or modification by the general user community. There is no need for users to have access to the installation-defined profile, because each user can create and save a unique user-defined profile (using the PPRF command) to override the installation-defined and Candle default profiles.

To protect the installation-defined profile, you can use either OMEGAMON’s default internal security or OMEGAMON’s interface to external security packages, such as RACF or CA-ACF2. Candle’s internal security requires a password for authorization to issue a command. An external security package checks authorization via the user ID and logon password. Securing the IPRF and IOPT commands prevents unauthorized users from using the following screens:

- Save New/Altered Installation Profile screen (contains the IPRF SAVE command)
- Delete New/Altered Installation Profile screen (contains the IPRF DELETE command)
- Set Performance Options screen (contains the IOPT command)


Step 3: Specify the Default Profile

If you want the installation-defined profile to be your site’s default profile, set the USER parameter in your startup procedure or CLIST to /I. For ISPF mode, specify /I in the USER SUFX field on the ISPF invocation panel.
Introduction

OMEGAMON II provides security systems for both the classic and CUA interfaces. There is also a security feature for the OMEGAMON II for DB2 to !DB/Tools.

Chapter Contents

Classic Interface Security
- Customizing Security for an Installation ........................................... 110
- Using Internal Security for Authorized Commands .......................... 112
- Using External Security ................................................................. 114
- Logging on Using External Security ............................................. 115
- Accessing Security from an Active Session ................................. 116
- Implementing External Security .................................................. 117
- Modifying the Security Table ....................................................... 124
- Control Statements ........................................................................ 127
- Generating a List of Security Updates ........................................... 136
- Using Optional External Security Features ................................. 141

CUA Interface Security
- Customizing CUA Security for an Installation .............................. 144
- Choosing a CUA Security Configuration ....................................... 145
- Using OMEGAMON II Internal Security .................................... 146
- Using CA-ACF2 External Security .............................................. 148
- RACF External Security ............................................................... 149
- CA-TOP SECRET External Security ............................................ 150

D2-TE Security
- Customizing D2-TE Security for an Installation ............................ 152
Customizing Security for an Installation

You can set up an interface between the realtime monitor and an external security package, such as RACF or CA-ACF2. The product is shipped with no security feature as the default.

Whether you use internal security, external security, or a combination of the two, you can customize the realtime monitor security table to the needs of your installation.

Terminology

This chapter uses the following terms in its discussion of the customization procedures:

- **Update Program**: The KOBSSUPDT member of rhilev.RKANMOD is a utility program that performs the update to the realtime monitor security table.
- **Control Statements**: The KO2SUPDI member of rhilev.RKD2SAM contains control statements that you can edit to change the defaults for internal security or to specify external security. KO2SUPDI provides the input for the update program.
- **JCL**: The KO2SUPD member of rhilev.RKD2SAM contains the JCL to run the security update program.
- **Exit Routine**: At initialization, the realtime monitor accesses the user’s security exit routine, which provides the interface to the external security package. The name of this routine must be specified by the installer.

**Important**

In this section, the use of the term *authorized* does not imply APF authorization.
Determining external versus internal security

When the realtime monitor is initialized, it determines whether an exit routine has been installed for an external security package.

- If the exit routine exists, it gets control for those commands that have been marked for external security and determines authorization through the external security package. If external security allows the command, the realtime monitor does not check internal security.

- If external security is not used for the command, internal security takes effect. The realtime monitor is shipped with certain authorized commands which require an internal security password for execution.

You can use external security alone, internal security alone, or external security on some commands and internal on others.
Using Internal Security for Authorized Commands

This section provides information on using internal security for authorized commands.

Levels of authorization

All realtime monitor commands (major, minor, immediate, and INFO-line) have a security level of 0, 1, 2, or 3. Level 3 provides the highest degree of protection. A setting of 0 means that any user can access the command.

Candle ships all commands with a default security level of 0. The KO2SUPDI member in rhilev.RKD2SAM contains instructions and the control statements for updating the security level of authorized commands as needed.

Warning

If you do not run the security update, certain commands that can potentially damage your system will be available to all users.

For the procedure to update the security level of realtime monitor commands, see “Security table update procedure” on page 125.

Authorized commands

Candle ships certain commands with a default security level of 3. We refer to these as authorized commands. The following commands are the authorized commands for this product:

- CONS
- DCMD
- DCNS
- .DSA
- MCHN
- MLST
- MSCN
- MZAP
- OCMD
- OSPC
- PEEK
- SCHN
- SLST
- SSCN
- SZAP
- TCMD
- XMCH
- XMLS
- XMSC
- XMZP

Note: The PEEK major command has the following minors: AMAP, DDNS, JOBS, MODS, STEP, SUBP, and TCBS. Candle also ships these minor commands with a default security level of 3.

Using passwords

Each security level can have its own password. The level 3 password accesses all levels; the level 2 password accesses levels 2 and 1; and the level 1 password accesses only the lowest level. Level 0 commands execute without a password.

If you enter a command that requires higher authority than yours, the realtime monitor responds with the message:

**OB0921 Security check failed (Internal)**

To gain access to the authorized commands, use the /PWD command in the following manner:
1. Type `/PWD` on the INFO-line. When you press Enter, the realtime monitor responds with the password prompt.

```
_              <== Please enter password
```

2. Type your password on the INFO-line. The password does not display as you type it.

3. Press Enter. If the **PASSWORD ACCEPTED** message displays, press Enter again to gain access to all authorized commands associated with that password, as well as lower command levels.

If you are using the realtime monitor with an external security package, you can prevent the use of the `/PWD` command. See “Using locking to maintain security” on page 142 for information.

To reset the security level to 0 when you have completed authorized functions, press the PA1 or the ATTN key. You can also reset it by using the `/PWD` command in the following manner:

4. Type the `/PWD` command on the INFO-line and press Enter. The password prompt displays.

5. Do **not** enter a password; just press Enter. You will see:

```
_________________ Password level reset
```

Access to authorized commands will be restricted until the password is re-entered.
Using External Security

The realtime monitor supports external security for all modes of operation.

External security is supported for both logon and command use. When using external security, users can log on to the realtime monitor only if they are allowed to access the INITIAL resource name. A resource name of INITIAL0, INITIAL1, INITIAL2, or INITIAL3 may be used to allow logon to the realtime monitor and set the internal security level to 0, 1, 2, or 3, respectively.

Performing external security checks

When a user issues a command, the realtime monitor performs an external security check if the following conditions are met:

- The user exit module name is specified in the security table.
- An external security exit routine is located and loaded.
- External security is specified for the issued command in the security table (using the COMMAND control statement with the EXTERNAL=YES keyword setting).
- For VTAM mode, the library that contains the KOBVTAM load module is APF-authorized.

If any commands are specified for external security checking and an exit routine is not found, the realtime monitor recognizes a possible security exposure and disables those commands with an internal security level of 0 for the session. Those commands with a level of 1, 2, or 3 are allowed to execute after you enter the internal password, as described in “Using Internal Security for Authorized Commands” on page 112.
Logging on Using External Security

This section explains special considerations for logging onto the OMEGAMON II realtime monitor using external security.

VTAM, TSO, or ISPF mode logon panel

When you logon through VTAM, OMEGAMON II presents a logon panel for the realtime monitor VTAM application program (KOBVTAM). The VTAM logon panel also appears for ISPF and TSO modes, since OMEGAMON II uses the realtime monitor VTAM application program for these modes as well. The copyright screen you normally see at logon time has additional fields for USERID, PASSWORD, GROUP, and NEW PASSWORD. The advantages of using the KOBVTAM logon screen are:

- The exit routine can cause the realtime monitor to stop an unauthorized logon.
- The exit routine makes all security checks based on the user’s logon ID and not on the authority for the realtime monitor address space.

Note that if you are in an active VTAM session and you want to alter the external security level of authorization, you can use the relogon feature discussed in “Accessing Security from an Active Session” on page 116.

Dedicated mode logon

Security in dedicated mode differs from the other modes since, at startup time, there is no user ID or password associated with the session. Therefore, the only security available by default is internal security. You must enter the /PWD command, using the relogon feature discussed in “Accessing Security from an Active Session” on page 116 in order to access external security.
Accessing Security from an Active Session

The relogon feature is a function of the /PWD command that

- allows you to enter your user ID and password for the external security package from an active realtime monitor session
- logs onto external security when used in dedicated mode
- enables users to alter the security level without having to bring down a current VTAM session, when used in VTAM mode

Using the relogon feature

To use the relogon feature, type in the /PWD INFO-line command and your user ID as in this example:

```
/PWD user01 O2INIT01 DED O2 V540./C $DB2 mm/dd/yy 17:03:37
```

Press Enter and type in your external security password at the prompt.

Note the following points regarding the use of the relogon feature:

- Be sure not to mark the /PWD command as EXTERNAL=YES in the security table because, in dedicated mode, you must use /PWD to log onto external security.
- You can determine in your user exit what the default action should be when the user ID or logon password supplied is not valid. For example, you may specify the disabling of all the realtime monitor commands marked as EXTERNAL=YES, or you may specify that the session reverts to the previous user ID. The available options are explained in the sample exit routines.
- If you use the relogon feature and your password has expired, you cannot enter a new one via the /PWD command.
Implementing External Security

This section provides information on implementing external security.

Procedure

To implement external security, follow these steps:

1. Modify the rules in the external security package to interface with the realtime monitor. See “Modifying RACF security rules” on page 118.

2. Customize the sample exit routine provided on the realtime monitor tape according to the procedure in “Creating an exit routine” on page 120. Refer to “Using Optional External Security Features” on page 141 for a description of options you may want to use.

3. Assemble and link-edit the routine.

4. Modify and update the security table to specify the commands to be checked by RACF or ACF2 and the name of the module that contains the exit routine. (No default is supplied for the module name.) Follow the steps in “Security table update procedure” on page 125.

Additional requirements for the started task

Make sure that your security package has authorized the OMEGAMON II started task identifier. During configuration, OMEGAMON II generates security jobs KO2ACF2A, KO2RACFA, and KO2SUPD, which create exit programs for external security packages. These members are in rhilev.RKD2SAM.
Modifying RACF security rules

To modify the RACF rules to interface with the realtime monitor, follow these steps:

1. Update the resource class description table to define a class name (for example, O2CANDLE) using the ICHERCDE macro call. (Be sure to use the same name when you define the resource class in the security exit routine.) We recommend coding the ICHERCDE macro as follows:

   ```
   ICHERCDE CLASS=classname,
   ID=nnn,
   MAXLNGTH=8,
   FIRST=ALPHANUM,
   OTHER=ANY,
   POSIT=nnn,
   DFTUACC=NONE
   ```

   Values for `classname` and `nnn` are determined by your installation. Additional operands for this macro may also be required at your installation.

2. Activate the newly defined resource class.

3. Define a resource profile for logging onto the realtime monitor. Use the TSO RDEFINE command with a resource of INITIAL. Here is an example of a definition that allows all users to sign onto the realtime monitor and use the /PWD command for internal security (that is, it allows access only to those commands marked EXTERNAL=NO):

   ```
   RDEFINE classname INITIAL UACC(READ)
   ```

   The variable `classname` is the name assigned in step 1 on page 118.

   This definition is the minimum required for logon. If you want to restrict the use of the /PWD command, see “Using locking to maintain security” on page 142.

4. Define resource profiles for the commands you wish to protect using external security (EXTERNAL=YES commands).

   A. Use the TSO RDEFINE command and specify the realtime monitor command as the resource. Be certain to specify that only specific users may execute the command by setting UACC(NONE).

   B. Use the PERMIT command to define those users who can access the resource (execute the command). Give them READ access.

   The following example shows how to authorize a user to execute the PEEK command with RACF:

   ```
   RDEFINE classname PEEK UACC(NONE)
   PERMIT PEEK CLASS(classname) ID(USER01) ACCESS(READ)
   ```
Implementing External Security

**Note:** When you authorize commands, the realtime monitor modifies the command name by replacing the slash of INFO-line commands with a dollar sign (/cccccc becomes $cccccc) and the period of immediate commands with @ (.ccc becomes @ccc). For example, /LOGOUT is defined to RACF as $LOGOUT in CLASS(cccccccc).

5. Include macro libraries in the assembly of the security exit routine. You can use SYS1.MACLIB and SYS1.AMODGEN as the macro libraries for RACF. In addition, you must include the Candle macro library, thilev.TKANMAC.

**Modifying ACF2 security rules**

To modify the ACF2 rules to interface with the realtime monitor, follow these steps:

1. If you are running the realtime monitor in dedicated or VTAM mode, define the name of the realtime monitor started task to ACF2.

   The started task name you use for the realtime monitor in VTAM mode should have the MUSASS attribute assigned. This allows ACF2 to check the individual user’s authorization rather than using the realtime monitor address space ID. If STC(NO) is specified, you must run the Common Interface in batch with a job name that has the MUSASS attribute.

2. Once you install the exit, you must set up a resource class for ACF2 to allow the realtime monitor to make the security checks. Define a generalized resource class name, for example O2S. This name will be three characters long for generalized resources, but will be prefixed with the letter R within the security exit. (Be sure to use the same name when you define the resource class in the security exit routine.)

3. Define an ACF2 rule for resource INITIAL to allow VTAM users to logon to the realtime monitor, as in the following example:

   ```
   ACFNRULE KEY(INITIAL) TYPE(O2s) ADD(UID(**********uid) ALLOW)
   ```

   O2S must match the resource class name that you defined in step 1 on page 119. `uid` is a user ID or user ID mask. If you want to restrict the use of the /PWD command, see “Using locking to maintain security” on page 142.

4. Use the ACF2 rule compiler to define resource rules for the command you wish to protect. Specify the command with the KEY operand.

   The following example shows how to authorize a user to execute the PEEK command with ACF2. See your security administrator for information on the format of the string.

   ```
   ACFNRULE KEY(PEEK) TYPE(O2S) ADD(UID(**********USER01) ALLOW)
   ```

   **Note:** When you authorize commands, the realtime monitor modifies the command name by replacing the slash of INFO-line commands with a dollar sign (/cccccc becomes $cccccc) and the period of immediate commands with @ (.ccc becomes @ccc). For example, /LOGOUT is stored in ACF2 as $LOGOUT.

5. Include the ACF2 macro library in the assembly of the routine. In addition, you must include the Candle macro library, thilev.TKANMAC.
Modifying CA-TOP SECRET rules

This section explains how to modify CA-TOP SECRET® rules.

Use the TSS PERMIT command to define those users who can access the resource (execute the realtime monitor command).

The following example shows how to authorize a user to execute the PEEK command with CA-TOP SECRET.

```
TSS PERMIT(userid) cccccccc(PEEK)
```

The variable `ccccccc` is the resource class name.

Creating an exit routine

The exit routine provides an interface between the realtime monitor and the security product. You can specify any unique name for your routine, but that name must also be specified in the control statements that update the security table. For more information, see the `MODULE=` parameter under “Control Statements” on page 127. The exit routine can be shared between systems.

The KO2ACF2X and KO2RACFX members of `rhilev.RKD2SAM` contain models of ACF2 and RACF routines. Many installations use these members without modification, but since security procedures are installation-dependent, they have been documented with comments to enable you to modify them. They are supplied as examples only.

**Note:** You must define a resource class in the exit routine. You must also be sure that the resource class you define in the exit routine has the same name as the resource class you defined when modifying RACF/ACF2 rules.

The `rhilev.RKD2SAM` dataset contains members called KO2ACF2A and KO2RACFA, which supply sample JCL to help you assemble and link-edit your routine.

You can use the same exit routine to define security for multiple realtime monitors. Use the same name on the `MODULE=` statement for each realtime monitor (see “MODULE control statement” on page 130). You could use the value of the B#DDPRFX field in the $BIA data area as part of a resource name to be used for the realtime monitor currently in use.

If you have a security system other than RACF or ACF2, you can still implement a security interface using these models. Use the sample RACF and ACF2 exits as guides to see what information is passed to the exit routine and what information is returned to the realtime monitor.
Realtime monitor calling conventions

The realtime monitor uses a single control block $UCHECK, to pass information to the exit routine. The exit routine also uses $UCHECK to pass information back to the realtime monitor. The $UCHECK control block is mapped by the $UCHECK macro. The macro is defined in member KOBGMAC of thilev:TKANMAC.

The realtime monitor maintains the control block for the entire life of the session, and gives the installation a 512-byte work area for its own use.

**Important**

The $UCHECK work area is limited to 512 bytes. If your installation requires a larger work area, GETMAIN the additional storage required and place the pointer to this GETMAINed area in $UCHECK. An attempt to enlarge this work area beyond its 512-byte limit in any other way causes an overlay of essential realtime monitor control blocks, and results are unpredictable. If you modify the RACF RACROUTE macro, you must GETMAIN at least 512 bytes for use as the WORKA parameter.

The user exit module is called by the realtime monitor with the following conventions:

- **Register 1**: Address of parameter list.
- **Register 13**: Address of a standard save area.
- **Register 14**: Return address.
- **Register 15**: Entry point address (in).
- **Register 15**: Return code (out).

Parameter list:

- **Word 1**: Address of control block.
Realtime monitor calling flow

The following procedure describes the flow for calls to your user security exit routine at initialization, during command verification, and at termination.

1. At initialization, when the realtime monitor passes control to your user exit routine, the initialization call is indicated by an I in the U#CHTYP field. This indicates that the realtime monitor requires a logon validation.
   A. If the user ID field length is nonzero, the user ID and password information are available.
   B. If additional information or some form of retry is required, the routine can request a reshow of the screen, and reset any field lengths to indicate that no data is present (user ID, password, group, or new password).
      To perform a reshow in VTAM mode, set a message into the U#CHMSG field (120 bytes maximum length), set the U@CHRSHO bit in U#CHRESP, and return to the caller. The message appears below the panel. Appropriate fields are filled in (original user ID and password), unless overridden (length = 0).
   C. When validation is complete, a return code of 0 from the user exit indicates that the user should be allowed to log on. Any other return code will cause the session to be aborted.
   D. Upon successful logon acceptance, the validation routine may perform resource validation and optionally assign a command security level (0, 1, 2, or 3) to the user. The default is 0. Place the appropriate number into U#CHAUT4. To force the user to use only this level, also set the U@CH1LOK bit in U#CHAUT1.

2. During command verification, the realtime monitor places a C in the U#CHTYP field. At this point, the user’s authorization can be checked. The decision to allow or disallow a command on the first encounter cannot be changed on subsequent tries by the same user unless security is reset with the /PWD command. However, on each try, the user exit is notified, an audit record may be written, and a customized error message may be issued. Return codes from the exit routine may be:

   **RC = 0** Indicates that the command is allowed (RACF and ACF2).
   **RC = 4** Indicates that the command is unknown to RACF (RACF only).
      The realtime monitor will allow the command to execute. See “Modifying RACF security rules” on page 118 for instructions to define a command to RACF.
   **RC = 8** Indicates that the command is known to the security package and access is denied (RACF and ACF2).

When you authorize commands, the realtime monitor modifies the command name by replacing the slash of INFO-line commands with a dollar sign (/cccccc becomes $cccccc), and the period of immediate commands with @ (.ccc becomes @ccc).

3. At relogon, the realtime monitor places an R in the U#CHTYP field to indicate a logon validation. The processing is the same as at initialization time, except that users may not enter a new password or group because the realtime monitor does not display a logon panel.
4. At termination, the realtime monitor passes a T to the user’s exit routine. You can then do any termination cleanup required, such as freeing user control blocks and FREEMAINing any GETMAINed areas.
Modifying the Security Table

This section describes how to update the security table for both external and internal security.

Security control keywords

The following table provides a summary of available security control keywords and their descriptions. For more information about security control statements and keywords, see “Control Statements” on page 127.

Table 9. Security Control Statements

<table>
<thead>
<tr>
<th>Control Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHLIB</td>
<td>Specifies an authorized screen space (PROC) library for initialization that bypasses the security check.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>Sets the internal security levels of commands, marks them for external security, and requests an audit.</td>
</tr>
<tr>
<td>LIST</td>
<td>Specifies whether a listing of the current security settings is to be produced on this run.</td>
</tr>
<tr>
<td>MINOR</td>
<td>Specifies the security options for minor commands.</td>
</tr>
<tr>
<td>MODULE</td>
<td>Specifies the name of the module containing the user’s external security exit routine.</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>Specifies the internal passwords.</td>
</tr>
<tr>
<td>RESET</td>
<td>Clears current settings.</td>
</tr>
<tr>
<td>SMFNUM</td>
<td>Specifies the record number for SMF audit requests.</td>
</tr>
<tr>
<td>UPDATE</td>
<td>Specifies whether updating is to be performed on this run.</td>
</tr>
</tbody>
</table>
Security table update procedure

To update the security table, follow these steps:

1. Edit the control statements in the KO2SUPDI member of rhilev.RKD2SAM.

   Important
   To change an existing setting for a parameter, you must specify a new setting rather than just blanking out the old setting. For example, to remove a command from external security checking, change EXTERNAL=YES to EXTERNAL=NO.

   The rules and keywords for the control statements are described in “Control Statements” on page 127.

   If you are implementing external security, you must enter the MODULE command statement naming the load module that will contain the exit routine. You must also indicate which commands are to use external security with the EXTERNAL=YES setting on the COMMAND control statement.

   To remove control from external security, blank out the value of the MODULE= keyword. Remember that if you do not change commands marked EXTERNAL=YES to EXTERNAL=NO, those with an internal security level of 0 will be nonexecutable.

2. Submit the job using the KO2SUPD member of rhilev.RKD2SAM. KO2SUPD contains the JCL to run KOBSUPDT, the security update utility program.

   KOBSUPDT performs the updates to the security table, and generates a list of the edits and, if requested, a complete list of security information. Successful completion of the job produces the message:

   OB9147 LOAD MODULE TEXT SUCCESSFULLY UPDATED

   If the update program flags statements as being in error during an update run, correct the statements and submit them again.

3. Reinitialize the realtime monitor. Changes made to the security table are effective only after the security update job completes successfully and a new realtime monitor session is started. Since the security table is part of a re-entrant load module, all the realtime monitor sessions in an address space must be stopped before security tables are effective. For example, if five VTAM mode sessions are active, all of them must stop before new sessions can use the updated security tables.
Format rules for control statements

These general format rules apply to all control statements:

- Control statements can begin anywhere in the input record, but cannot extend beyond column 72.
- Statements can be in any order in the input stream. The update program processes the statements as it encounters them, with the exception of the LIST and UPDATE statements, which take effect after all other input is processed.
- All information for a particular control statement must fit in a single line.
- All input must be in uppercase letters.
- Statements must be in this format:

```
CONTROLSTATEMENT=cccccccc,KEYWORD1=cccccccc,KEYWORD2=cccccccc,
``` etc.

There can be no intervening blanks. The update program treats data that follows a blank as a comment. The data prints on the edit listing, but is ignored for processing purposes.

- To insert comment lines anywhere in the input stream, place an asterisk (*) in column one of the input record.
- If the update program flags statements as being in error, correct the statements and resubmit them. To change a setting, you must specify a new setting rather than blank out the old setting. This is especially important to remember when changing a command from EXTERNAL=YES to EXTERNAL=NO.
- The realtime monitor does not recognize changes to control statements until the update job successfully terminates and a new realtime monitor session is started. The control statement edit listing should indicate successful completion of the update.
Control Statements

The following sections describe the available control statements and their keywords. Keyword defaults are underscored.

**AUTHLIB control statement**

This control statement specifies the dataset name of an authorized screen space library that contains commands to invoke at the realtime monitor initialization, bypassing any security checks. This option lets you execute protected commands as part of the initialization screen without entering a password.

Since all security checking for screens coming from the AUTHLIB dataset is bypassed, WRITE access to this dataset should be restricted.

Security checking resumes when the realtime monitor fetches a screen from an unauthorized library, or a screen that has been loaded into memory, or when a user enters any keystroke, including a cursor movement.

**Important**

If you create an authorized screen library and use the realtime monitor menu system, security checking will cause initialization to fail when either of the following occurs:

- The realtime monitor fetches a screen containing an authorized command. Menu system users should leave the .FGO and .VAR commands unprotected.
- The realtime monitor fetches a screen space that has been loaded into memory. Screen @ZSCRNDF loads screen spaces into memory. Screen @ZSCRNDF loads screen spaces into memory.

The format of the AUTHLIB control statement is as follows:

```
AUTHLIB=dsname,VOL={volume|NOVOLUME}
```

where *dsname* is the name of the authorized screen library you have created.

AUTHLIB accepts the following keyword:

**VOL**

Specifies the volume serial where the specified dataset resides. This acts as an additional security measure. You may specify a volume serial number even if the dataset is cataloged.

The AUTHLIB statement always requires the VOL keyword. If you do not want the additional volume serial number checking to be performed, specify NOVOLUME.

Concatenate the dataset containing the authorized screens in your O2PROC DD statement.

The dataset that contains the authorized screen libraries is not an APF-authorized dataset.
**COMMAND control statement**

This control statement specifies the name of a realtime monitor major, immediate, or INFO-line command to be protected. Minor commands are protected at the major command level unless the MINOR control statement is specified.

When you update an INFO-line command, you must use the actual command name and not its alias. The realtime monitor automatically assigns the same protection attributes to all aliases of the command.

The realtime monitor does not check for multiple COMMAND statements for the same command in the same run. The last COMMAND statement for the command is the one that the realtime monitor processes.

The format of the COMMAND control statement is as follows:

```
COMMAND=cccc, .ccc, /cccccc
[, LEVEL={0|2|3|DISABLE}]
[, EXTERNAL={YES| NO}]
[, AUDIT={WTO|SMF|BOTH|NONE}]
```

where cccc, .ccc, or /cccccc is the name of the realtime monitor command to be protected.

To have the control statement edit listing show the current security settings for a command, enter a `COMMAND=cccc, =.ccc, or =/cccccc` statement with no additional operands.

COMMAND accepts the following keywords:

**LEVEL**

Specifies the internal security level to be associated with this command. Level 0 allows the command to execute without an internal security check. Levels 1, 2, and 3 specify that the command executes only if you have previously entered the corresponding password for that level (or for a higher level) via the /PWD INFO-line command.

DISABLE specifies that the realtime monitor is never to execute the command. You can audit attempts to execute the command for the session, but you cannot specify internal or external security.

**EXTERNAL**

Specifies whether an external security package checks this command. The realtime monitor ignores the EXTERNAL keyword if you specify LEVEL=DISABLE.

If you code EXTERNAL=YES for a command and no exit routine is available, the realtime monitor disables the command for the session if it has an associated security level of 0, or defaults to internal security if the command has a security level of 1, 2, or 3.

Once you specify EXTERNAL=YES, you can change it only if you specify EXTERNAL=NO and rerun the security update program.

**AUDIT**

Specifies whether the realtime monitor is to audit the command each time a user invokes it. The possible values are:
Control Statements

When running the realtime monitor, you can specify control statements that customize its operation. These statements allow you to tailor the behavior of the monitor according to your security policies and requirements.

### LIST control statement

This control statement specifies whether the update program produces a security file listing. A security file listing is a complete record of the security table that shows the name of the authorized screen library, its volume serial number, the name of the user exit module, and all command names along with their corresponding security information. It does not list the internal security passwords.

If you also specify UPDATE=NO, the listing shows what the control statements and security information would look like if the update had taken place.

To generate the security file listing independent of edits to the control statements, you may submit LIST=YES as the only control statement in the input stream.

Only one LIST statement is allowed per run. The default is LIST=NO.

The format of the LIST control statement is as follows:

\[
\text{LIST} = \{\text{YES} | \text{NO}\}
\]
Control Statements

MINOR control statement

This control statement specifies the name of a realtime monitor minor command to protect. The realtime monitor protects the minor commands independent of the majors. Therefore, any changes to minor commands apply to all minors with the same name and attributes, regardless of their major commands.

Access to a minor command requires access to the appropriate major command. If you do not specify an EXTERNAL keyword, the associated major controls access to this minor command.

No check is made for multiple MINOR statements for the same minor command in the same run. The last MINOR statement for the minor takes effect.

The format of the MINOR control statement is as follows:

```
MINOR=cccc
[, LEVEL={1 | 2 | 3 | DISABLE}]
[, EXTERNAL={YES | NO}]
[, AUDIT={WTO | SMF | BOTH | NONE}]
```

where cccc is the name of the minor command to be protected.

Refer to the COMMAND control statement for explanations of the keywords.

MODULE control statement

This control statement specifies the name of the module that contains the user’s external security exit routine. You must specify this parameter for an external security check to take place. There is no default.

The format of the MODULE control statement is as follows:

```
MODULE=cccccccc
```

where cccccc is the name of the module that contains the user’s external security exit routine. Be sure that this name matches the load module name you specified in KO2ACF2A or KO2RACFA.

To remove control from external security, blank out the value of MODULE=, run the security update job, and restart the realtime monitor.
**PASSWORD control statement**

This control statement specifies the 1- to 8-character password for each internal security level, to be used with the /PWD command. Use a separate PASSWORD control statement for each security level. Use unique passwords for each security level. When you enter a valid password for one security level, the realtime monitor allows access to commands secured at that level, and commands secured at lower levels. The realtime monitor checks the password for a match in the following order:

- Level 1
- Level 2
- Level 3

If you assign the same password to more than one level, the realtime monitor will match it only at the lowest level, and deny access to commands protected at higher levels.

The format of the PASSWORD control statement is as follows:

```
PASSWORD=password,LEVEL={1|2|3}
```

where `password` is the unique password for this level.

PASSWORD accepts the following keyword:

- **LEVEL** Specifies the security level associated with this password. Levels 1, 2, and 3 specify that the command executes only if you have previously entered the corresponding password for that level (or for a higher level) via the /PWD INFO-line command. A level is always required for a password.


RESET control statement

This control statement clears the current settings of the other control statements. Reset
commands remain unprotected unless you specify new settings with the appropriate
control statements and rerun the update program.

Only one RESET statement is allowed per run.

The format of the RESET control statement is as follows:

RESET=cccccccc

where cccccccc is one of the following arguments.

ALL
Clears settings for all control statements and all keywords in the security
table for the realtime monitor.

AUTHLIB
Clears the name and volume serial number of the authorized library.

INFO
Clears settings for all INFO-line commands (on the COMMAND control
statement). For example, if you do not want to use the Candle Corporation
default security levels for INFO-line commands and want to start over, enter
RESET=INFO. For INFO-line commands, this resets all LEVEL settings to
security level 0 and also clears any existing EXTERNAL and AUDIT
settings.

MAJOR
Clears settings for all major and immediate commands (on the COMMAND
control statement). See INFO above for an example.

MINOR
Clears settings for all minor commands.

MODULE
Clears the name of the user’s exit routine module.

PASSWORD
Clears the internal passwords.

SLASH
Clears the record number for SMF audits.

YES
Same as ALL.
SMFNUM control statement

This control statement indicates the ID number of the SMF record that the realtime monitor should use for its audit.

The format of the SMFNUM control statement is as follows:

```
SMFNUM=nnn
```

where `nnn` is the SMF record ID number. This ID number assigned to the realtime monitor must be between 128 and 255, inclusive, and should be different from that used by any other application. There is no default.

UPDATE control statement

This control statement specifies whether the realtime monitor updates the control statements during this run. `UPDATE=NO` specifies that this run of the security update program should be a trial run.

Only one UPDATE statement is allowed per run. The default is `UPDATE=YES`.

The format of the UPDATE control statement is as follows:

```
UPDATE={YES | NO}
```

Using control statements to update the security table

The following figure shows an example of using control statements to update the security table:

```
COMMAND=PEEK LEVEL=1
COMMAND=.DSA,LEVEL=3,EXTERNAL=YES,AUDIT=WTO
COMMAND=MLST,EXTERNAL=YES
COMMAND=XML,LEVEL=DISABLE,AUDIT=BOTH
COMMAND=XMLS,LEVEL=2
COMMAND=MINOR=JOBS,LEVEL=2
COMMAND=/SAVE,LEVEL=1,AUDIT=None
MODULE=MYSECURE
SMFNUM=233
LIST=YES
UPDATE=NO
```
The command control statements in the previous figure result in the following settings for the realtime monitor commands listed:

**PEEK**
A user who has specified the internal security level 1 or higher password can execute PEEK and its minors. The realtime monitor does not perform external security checking.

**.DSA**
The realtime monitor performs external security checking and writes a message on the master console each time .DSA is invoked. If external security is unavailable, only a user who specified the internal security level 3 password can execute .DSA.

**MLST**
The realtime monitor performs external security checking, but no auditing.

**XMZP**
The command cannot be executed. The realtime monitor writes a message on the master console and writes an SMF record when XMZP is issued. There is no external security checking.

**XMLS**
A user who has specified either the level 2 or level 3 internal security password can execute XMLS.

**JOBS**
JOBS is a minor of the PEEK command which was specified above as a level 1 authorized command; however, the LEVEL=2 setting on JOBS specifies that only level 2 or 3 users can access it.

**/SAVE**
A user who has specified either the level 1, level 2, or level 3 password can execute the /SAVE command. It is not audited.

The remaining control statements result in the following settings:

**MODULE**
MYSECURE is the name of the module that contains the security exit routine.

**SMFNUM**
The SMF ID is set as 233.

**LIST**
YES indicates that the realtime monitor produces a listing.

**UPDATE**
NO indicates that the realtime monitor does not update the security table. This is a trial run.
Example of a security update job

Figure 3 on page 135 shows Candle-supplied JCL to run the security update program. This JCL resides in rhilev.RKD2SAM(KO2SUPD).

FIGURE 3. Example of Security Update Job

```
//SECURITY JOB ,
//STEP1 EXEC PGM=KOBSUPDT,PARM=KO2OCMDB
//STEPLIB DD DSN=rhilev.RKANMOD,DISP=SHR
//SYSLIB DD DSN=rhilev.RKANMOD,DISP=SHR
//SYSPRINT DD SYSOUT=* 
//SYSIN DD DSN=rhilev.RKD2SAM(KO2SUPD),DISP=SHR
```

Modify this job to contain information appropriate for your installation. Then, submit the job to run the security update program.

**PARM=**
Identifies the load module that contains the realtime monitor security table.

**STEPLIB**
Specifies the name of the load library where KOBSUPDT resides.

**SYSLIB**
Specifies the name of the library where KO2OCMDB resides.

**SYSPRINT**
Specifies the print output for the various reports that the security program produces.

**SYSIN**
Specifies the name of the dataset that contains the control statements.
Generating a List of Security Updates

The security update program generates a listing of the control statement modifications that have been made. With the LIST control statement, you have the option of producing an additional listing that includes all security information.

The security update program listing consists of four parts.

- Header
- Control Statement Edit Listing
- Security File Listing
- Security Update Program Trace

Header

The header of the security update program contains the following information:

- The dataset name where the load module resides.
- The module name of the security table.
- The DB2 version number in the format XnnnCOM, where nnn represents the DB2 version number. For example, X600COM represents DB2 Version 6.1.
- Messages indicating successful completion of the job or error conditions, such as a failure to open the SYSLIB dataset or read the security table.

Figure 4 on page 136 shows a typical header.

FIGURE 4. Typical Security Update Program Header

```
OBSECUP 1.2--OMEGAMON SECURITY UPDATE PROGRAM--(c) CANDLE CORPORATION-- mm/dd/yy 16:41
OB9261 OBSECUP BEGUN
OB9144 OBSELR00 CALLED TO READ O2OCMDB
OB9148 SYSLIB DCB OPENED SUCCESSFULLY
OB9149 LIBRARY DSNAME IS: rhilev.RKANMOD
OB9158 LOAD MODULE ID:
OMCMDEX
X540COM
mm/dd/yy 19:02
OB9146 LOAD MODULE TEXT SUCCESSFULLY READ
OB9150 SYSLIB DCB CLOSED
OB9262 LOAD MODULE READ RETURN CODE IS 0000
```
Control statement edit listing

The control statement edit listing contains a list of the control statements that have been edited. The list shows the previous contents (except for previous passwords), as well as the new contents.

If you specified `UPDATE=YES`, the realtime monitor reports the date and time of the previous update.

Figure 5 on page 137 shows a typical listing.

FIGURE 5. Typical Control Statement Edit Listing

```
OBSECUP 1.2--OMEGAMON SECURITY UPDATE PROGRAM--(c) CANDLE CORPORATION-- mm/dd/yy 16:41

*** CONTROL STATEMENT EDIT ***

AUTHLIB=rhilev.RKO2PROC,VOL=NOVOLUME
PREVIOUS CONTENTS =
NEW CONTENTS = rhilev.RKO2PROC  NOVOLUME

* CHANGE THE PASSWORD FOR LEVEL 3 COMMAND ACCESS
PASSWORD=CANDLE3,LEVEL=3
PREVIOUS CONTENTS = ******** NEW CONTENTS = CANDLE3

* DISPLAY SECURITY INFORMATION FOR THE PEEK COMMAND
COMMAND=PEEK
PREVIOUS CONTENTS = 3 B NEW CONTENTS = 3 B

* DISPLAY SECURITY INFORMATION FOR MINOR JOBS
MINOR=JOBS
PREVIOUS CONTENTS = 0EW NEW CONTENTS = 0EW

* PROTECT MZAP COMMAND
COMMAND=MZAP,LEVEL=3
PREVIOUS CONTENTS = 0 NEW CONTENTS = 3

* DISABLE CONS COMMAND
COMMAND=CONS,LEVEL=DISABLE,AUDIT=BOTH
PREVIOUS CONTENTS = 0 NEW CONTENTS = * B

*** END OF CONTROL STATEMENT INPUT ***
```
The codes for the **PREVIOUS CONTENTS** and **NEW CONTENTS** of commands are positional. There are three positions:

1. The first position shows the number of the internal security level or an asterisk (*) if the command has been DISABLED.

2. The second position shows the external security option:

   - **E** Use external security for this command.
   - **b** A blank specifies no external security.

3. The third position shows the auditing option:

   - **W** Audit this command via WTO.
   - **S** Audit this command via SMF.
   - **B** Audit this command via WTO and SMF.
   - **b** A blank indicates no auditing.
Security file listing

If you specify **LIST=YES** anywhere in the input stream, the security update program generates a complete listing of the security information, including the name of the authorized screen library and its volume serial number, the name of the external security user exit module, the SMF record number, and all of the commands along with their security information. The listing does not show the internal security passwords.

Figure 6 on page 139 shows a partial security file listing.

**FIGURE 6. Typical Security File Listing**

```
OBSECUP 1.2 - OMEGAMON SECURITY UPDATE PROGRAM-- (c) CANDLE CORPORATION-- mm/dd/yy 16:41
 * * * S E C U R I T Y  F I L E  L I S T I N G  * * *

AUTHLIB=rhilev.RKO2PROC       VOLUME=NOVOLUME
LEVEL1=********    LEVEL2=********    LEVEL3=********
SMFNUM=233
MODULE=MYSECURE

COMMAND= /A  0           TYPE=S    (ALIAS)
COMMAND= /ABORT 0           TYPE=S
.
COMMAND= /AUP  0           TYPE= S
.
COMMAND= .AUP  0           TYPE= I
COMMAND= .DSA  0           TYPE= I
COMMAND= .SCC  *           TYPE= I
.
COMMAND= OCMD  3EB         TYPE= I
.
SECURITY TABLE LAST UPDATED ON mm/dd/yy 06:00:10
```

**TYPE** specifies the following kinds of the realtime monitor commands:

- **C** Major
- **I** Immediate
- **S** Slash (INFO-line)

The security level follows the command. An asterisk (*) indicates that a command has been disabled. Minor commands are listed below their corresponding majors.
Generating a List of Security Updates

Security update program trace
The last part of the listing indicates whether an update has successfully completed.

Figure 7 on page 140 shows a typical trace.

FIGURE 7. Typical Security Update Program Trace

Creating the SMF audit
When creating the SMF audit, make sure that the SMF Record Exits (IEFU83 and IEFU84) and the SMF system parameters specifications (SMFPRMcc) do not suppress the ability of the realtime monitor to journal the audit activity records.

The SMF record consists of:
- The IBM header (mapped by IFASMFR).
- The Candle Corporation Common Header (mapped by $CANHDR, which is defined in member KOBGMAC of thilev.TKANMAC).
- The security audit record (mapped by $AUDIT, which is defined in member KOBGMAC of thilev.TKANMAC).

The audit record contains:
- A date/time/system stamp.
- A user ID/job name associated with the session.
- The actual command text as it was entered on the realtime monitor screen. Records of minor commands also reference their associated major commands.

Use the SMF audit selectively. Because the overhead for producing SMF records is high, it is intended for use only with sensitive commands, such as those that could disrupt the system (for example, OCMD and MZAP).

The KOBSMFRRP member of the thilev.RKD2SAM dataset contains a sample SMF post-processor and report generator in source code format. This is supplied as an example only.
Using Optional External Security Features

You can set up your user exit routine (as explained in “Creating an exit routine” on page 120) to use any of the options discussed in this section. Remember that you can also use the control options supplied with the security package, such as SHIFT validation and SOURCE validation, by marking the commands EXTERNAL=YES and implementing the option as the security package directs.

Customizing error messages

To suit your individual requirements, your installation can create custom error messages to display when a user has insufficient authority, or enters an invalid user ID or password. See the sample exit routines in the members KO2RACFX and KO2ACF2X in the rhileu.RKD2SAM dataset for information about how to do this. The user security message may be up to 120 bytes long, except for INFO-line messages (for example, /PWD relogin messages), which may be a maximum of 60 bytes.

Updating a password

You can give the user the capability of interactive communication when logging on to external security. For example, if a user logs on with an expired password, the security exit may prompt the user for a new password and update the security database. This capability is not available when relogging on with the /PWD command.

Suppressing audit

The realtime monitor gives you the flexibility of suppressing WTO or SMF auditing. At initialization or relogin, your exit routine may set a flag in $UCHECK to indicate WTO or SMF suppression.

Supplementing audit

In addition to the WTO and SMF audits available with the realtime monitor, you can use the audit features of the external security package to supplement command tracking. The RACF Report Writer and ACF2 ACFRPT utility programs are examples of this supplemental audit capability.
Using locking to maintain security

The locking feature is designed to prevent users from changing their internal security level with the /PWD command. Their level of authority is set only once and only at logon. It can be fixed to one of four levels (level 0, 1, 2, or 3).

Note the following considerations:

- Although the locking feature is implemented in the external security exit routine, it is designed to lock the user’s internal security level. Therefore, it affects only those commands marked as EXTERNAL=NO.
- You must define a user’s security level in ACF2 or RACF as an INITIALn resource, where n is a number from 0 to 3, and assign corresponding values to commands in the security update program (using the LEVEL keyword of the COMMAND control statement).
- The locking feature only disables the /PWD command for supplying internal passwords. /PWD can still be used for relogon to a new external userid.
- Users assigned INITIAL authority (no value of 0 to 3 attached) are allowed to change their internal security level by using /PWD.
- The routine starts checking INITIALn resources at the highest level. If you define INITIAL3 and INITIAL2 and PERMIT user to INITIAL3, the user is locked to level 3.

Validating a user via RACF

To validate a user, the user exit routine checks on the RACF resource class that is defined by the ICHERCDE macro.

The resources that allow the realtime monitor startup include INITIAL, INITIAL0, INITIAL1, INITIAL2, and INITIAL3, as shown in the following example:

```
<Allows /PWD to work>
RDEFINE cccccccc INITIAL UACC(READ)

<Defines security level 0 as unaccessible>
RDEFINE cccccccc INITIAL0 UACC(NONE)

<Defines security level 1 as unaccessible>
RDEFINE cccccccc INITIAL1 UACC(NONE)

<Defines security level 2 as unaccessible>
RDEFINE cccccccc INITIAL2 UACC(NONE)

<Defines security level 3 as unaccessible>
RDEFINE cccccccc INITIAL3 UACC(NONE)

<Locks USER02 to level 2 power>
PERMIT INITIAL2 CLASS(classnme) ID(USER02) ACC(READ)
```

The variable classnme is the resource class name you defined in “Modifying RACF security rules” on page 118.
Validating a user via ACF2

The user exit routine checks the ACF2 resource class to validate a user.

The resources that allow the realtime monitor startup include INITIAL, INITIAL0, INITIAL1, INITIAL2, and INITIAL3. To allow users to change their authorization level with the /PWD command, use INITIAL. Here are sample definitions:

```
<Allows /PWD to work for USER01>
ACFNRule KEY(INITIAL) TYPE(cls) ADD(UID(**************USER01) ALLOW)

<Locks USER02 to security level 0 commands>
ACFNRule KEY(INITIAL0) TYPE(cls) ADD(UID(**************USER02) ALLOW)

<Locks USER03 to security level 1 commands>
ACFNRule KEY(INITIAL1) TYPE(cls) ADD(UID(**************USER03) ALLOW)

<Locks USER04 to security level 2 commands>
ACFNRule KEY(INITIAL2) TYPE(cls) ADD(UID(**************USER04) ALLOW)

<Locks USER05 to security level 3 commands>
ACFNRule KEY(INITIAL3) TYPE(cls) ADD(UID(**************USER05) ALLOW)
```

The variable `cls` is the generalized resource class name you defined in “Modifying RACF security rules” on page 118.

**Note:** The UID operand is installation-specific in format and content. For information about UID, contact your security administrator.

Determining the ID of the DB2 subsystem being monitored

The U#CHPIA field of the #CHECK macro points to an area that the following DSECT maps:

```
O2PIA      DSECT
O2PIAEYE   DS CL8 'O2PIA’ eyecatcher
O2PIALEN   DS F  'length of workarea’
O2PIASSI   DS CL4 'DB2 subsystem ID’
    DS 20F   reserved
O2PIANDEYE DS CL8 'O2PIA’ ending eyecatcher
```

Field O2PIASSI contains the subsystem ID of the DB2 that OMEGAMON is currently monitoring. The string that contains this information is in key 7 and above the 16MB line in the OMEGAMON II for DB2 private storage area.
Customizing CUA Security for an Installation

This section provides information to help you choose and implement a security facility that meets the CUA interface requirements at your site. You can use the CUA interface security facility in conjunction with the classic interface security facility (see “Customizing Security for the Realtime Monitor” on page 109).

Terminology

You specify the names of the started tasks using CICAT. Remember to distinguish between the name of the classic interface started task and the name of the CUA interface started task.
Choosing a CUA Security Configuration

CUA interface security provides a user ID and password validation capability to detect and prevent unauthorized access to the product.

Network Access Manager (NAM) provides security at the CUA interface (to prevent unauthorized access of OMEGAMON II), and OMEGAMON II provides command password protection for security at the menu system interface (to prevent unauthorized use of OMEGAMON II commands).

In choosing an appropriate security system for your site, you can:
- select NAM as the security systems for your site
- implement an external security package such as CA-ACF2 (also referred to as ACF2), RACF, or CA-TOP SECRET to replace NAM at the CUA interface
- implement an external security package to be used in conjunction with OMEGAMON II at the classic interface

**Important**
Security should be implemented at both the CUA interface and the classic interface.

### CUA security options

To help you choose the right security system for your OMEGAMON II environment, we recommend that you see your security administrator for information about the types of security used at your site.

The following table shows the CUA interface security customization choices available for use with OMEGAMON II.

<table>
<thead>
<tr>
<th>System Name</th>
<th>Exit Required?</th>
<th>Exit Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAM</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>CA-ACF2</td>
<td>Yes</td>
<td>KD2A2NEV</td>
</tr>
<tr>
<td>RACF</td>
<td>No</td>
<td>None</td>
</tr>
<tr>
<td>CA-TOP SECRET</td>
<td>Yes</td>
<td>KD2TSNEV</td>
</tr>
</tbody>
</table>
Using OMEGAMON II Internal Security

This section describes how to use the security system shipped with OMEGAMON II.

NAM CUA interface security

The OMEGAMON II Network Access Manager (NAM) can serve as a standalone security system. NAM security functions at the CUA interface only.

Use the following steps to implement NAM security:

1. Modify the security system definition in member rhilev.RKD2PAR(KD2INNAM) as follows:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) NORACF
   ```

   **Note:** Keywords cannot extend past column 72. To continue keyword parameters on separate lines, end each line with a space and a hyphen.

   Example:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) -
   NORACF
   ```

2. Define all authorized OMEGAMON II users to NAM. Remember to include your own user ID and password in this step as NAM will not assume them automatically. Issue the MVS MODIFY command from the MVS console as follows:
   
   ```
   F cccccccc,NAM SET userid1 PASSWORD=password1
   F cccccccc,NAM SET userid2 PASSWORD=password2
   ```

   where cccccccc represents your name for the CUA interface started task. You must issue this command for each new user of OMEGAMON II.

3. Log on to OMEGAMON II using your user ID and password. The password you set using the MVS MODIFY command will expire the first time you log on. Each user must set a new password to enter the product for the first time.

   NAM maintains a record of each user’s previous passwords. The last eight passwords set by a user are kept on file, and new passwords cannot match any of the eight listed.
Maintaining NAM security

After implementing NAM security, maintain users and passwords as follows:

- To modify a user’s password, reissue the modify command you used to initially set the password.
- To delete a NAM user, enter the following command:
  
  F cccccccc,NAM DELETE userid

  where cccccccc is the name of the started task for the D2 CUA that you specified using CICAT.

- To control the number of times a user can logon before a change of password is required, add the EXPIRE=nn parameter to the NAM SET command. The default is 0 (no expiration).
This section describes how to link the CUA interface to the ACF2 external security system.

**CA-ACF2/CUA interface security**

Follow these steps to interface with ACF2 security validation at the CUA interface:

1. Modify the security system definition in member `rhilev.RKD2PAR(KD2INNAM)` as follows:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) EXIT=KD2A2NEV NORACF NODB
   ```

   **Note:** Keywords cannot extend past column 72. To continue keyword parameters on separate lines, end each line with a space and a hyphen.

   Example:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) -
   EXIT=KD2A2NEV -
   NORACF -
   NODB
   ```

2. Install the exit for ACF2 security validation. Member `rhilev.RKD2SAM(KD2A2NEV)` is the sample assembler interface to ACF2.
   
   Assemble and link KD2A2NEV with AC=1 into `rhilev.RKANMODL`. Member `rhilev.RKD2SAM(KD2@ASM)` contains assembly JCL that can be modified according to instructions in the member.

   **Note:** ACF2 exits must be run with AMODE=31.

3. The multiuser system access control point used by OMEGAMON II has all the characteristics of an ACF2 Multiple User Single Address Space System (MUSASS). That is, system access validations are initiated and enforced by the address space on behalf of the network user.
   
   Define OMEGAMON II as a MUSASS to ACF2. Follow this sequence from the TSO READY prompt:
   
   ![](image)

---

148 OMEGAMON II for DB2 Configuration and Customization Guide, V540
RACF External Security

This section describes how to interface with the RACF external security system.

RACF/CUA interface security

Follow these steps to interface with RACF security at the CUA interface:

RACF must give CONTROL authority to the VSAM files rhilev.RKD2NAM, rhilev.RKD2TDB, and rhilev.RKD2VLOG.

Modify the security system definition in member rhilev.RKD2PAR(KD2INNAM) as follow:

```
DEFAULT DSNAME(rhilev.RKD2NAM) RACF NODB
```

**Note:** Keywords cannot extend past column 72. To continue keyword parameters on separate lines, end each line with a space and a hyphen.

Example:

```
DEFAULT DSNAME(rhilev.RKD2NAM) -
    RACF -
    NODB
```

Protecting the OMEGAMON II VTAM applid

Perform the following steps to protect the OMEGAMON II applid.

**Note:** The CUA primary applid was supplied on the CUA Interface Customization panel. Change cuapplid in the following steps to the value that was specified for the CUA primary applid.

1. Create the following profile in the APPL class:
   ```
   RDEFINE APPL cuapplid UACC(NONE)
   ```

2. Allocate READ access to the profile as follows:
   ```
   PERMIT cuapplid CLASS(APPL) ID(userid) ACCESS(READ)
   ```

3. Activate the APPL class as follows:
   ```
   SETROPTS CLASSACT(APPL)
   ```
CA-TOP SECRET External Security

This section describes how to interface with the CA-TOP SECRET external security system. CA-TOP SECRET can be implemented at the CUA interface (to prevent unauthorized access of OMEGAMON II), the menu system interface (to prevent unauthorized use of critical OMEGAMON II commands), or both.

CA-TOP SECRET/CUA interface security

Follow these steps to interface with CA-TOP SECRET external security at the CUA interface:

1. Modify the security system definition in member `rhilev.RKD2PAR(KD2INNAM)` as follows:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) EXIT=KD2TSNEV RACF NODB
   ```
   
   **Note:** Keywords cannot extend past column 72. To continue keyword parameters on separate lines, end each line with a space and a hyphen.

   Example:
   
   ```
   DEFAULT DSNAME(rhilev.RKD2NAM) -
   EXIT=KD2TSNEV -
   RACF -
   NODB
   ```
   
2. Install the exit for CA-TOP SECRET security validation. Member `rhilev.RKD2SAM(KD2TSNEV)` is the sample assembler interface to CA-TOP SECRET.

   Assemble and link KD2TSNEV with AC=1 into the `rhilev.RKANMODL` library. Member `rhilev.RKD2SAM(KD2@ASM)` contains assembly JCL that you can modify according to instructions in the member.

   **Note:** CA-TOP SECRET exits must be run with AMODE=24.

3. Define the OMEGAMON II address space as a started task in the STC record, along with the related master FACILITY ACID. For example, enter:

   ```
   TSS ADD(STC) PROC(task) ACID(master facility acid)
   ```

   where `task` represents the name of the CUA interface started task. The default started task name is D2CUA.

4. (Optional) Define all the OMEGAMON II datasets you want protected to CA-TOP SECRET. Make sure the OMEGAMON II started task has access to them.

5. Define `task` as a FACILITY to CA-TOP SECRET in the Facility Matrix Table, where `task` is the started task name. If the name you define in the FACILITY statement is different from the started task name, see the appropriate CA-TOP SECRET documentation for information on setting up the FACILITY statement.
The following example shows the FACILITY statements from a production installation using CA-TOP SECRET as the security system:

```
FACILITY(USER3=NAME=task)
FACILITY(task=MODE=FAIL,ACTIVE,SHRPRF)
FACILITY(task=PGM=VTP,NOASUBM,NOABEND,NOXDEF)
** IM# 208852 Take out LUMSG and STMSG parameters... NP 6/29/92
FACILITY(task=ID=1,MULTIUSER,REG,WARNPW,SIGN(M))
FACILITY(task=NOINSTDATA,NORNDPW,AUTHINIT,NOPROMPT,NOAUDIT,NOMRO)
FACILITY(task=NOSOC,LOG(INIT,SMF,MSG,SEC9))
```

Some of these statements may not be relevant to your system, and may need to be modified to fit your standards and configuration.
Customizing D2-TE Security for an Installation

This section provides security considerations for DB2 and DB/TOOLS.

DB2 Security Considerations

The effective AUTHID SET must have the following DB2 authorities:

- SELECT on DB2 catalog tables.
- EXECUTE on DB/EXPLAIN plans and packages. The DB/EXPLAIN JCL? command can be used to bind and issue grants.
- SELECT, UPDATE, and DELETE on the destination PLAN_TABLE.

Note: The destination PLAN_TABLE is the current SQLID PLAN_TABLE. If the current SQLID PLAN_TABLE does not exist, DB/EXPLAIN attempts to create it in the tablespace PLANTBLS in the DB/TOOLS default database. If the PLAN_TABLE cannot be created there, DB/EXPLAIN attempts to create it in the DSNDB04 database. If the create fails, DB/EXPLAIN issues an error message that the EXPLAIN could not be performed.

The current SQLID is determined by EXP_POWNERX, which is defined in the DB/TOOLS profile dataset. If EXP_POWNERX is blank, the current SQLID is determined by the DB2 security exit.

- Authority to execute the SQL statements to be EXPLAINed. If the D2 USERID LOGON is enforced, the USERID is placed in JCTUSER and ASXBUSER. If the D2 USERID LOGON is not enforced, the D2-TE started task name is placed in JCTUSER and ASXBUSER.

If the DB2 security exit uses either the JCTUSER or the ASXBUSER fields as the source for the USERID, the effective AUTHID SET is the primary AUTHID (AIDLPRIM) plus the secondary AUTHIDs (AIDLSEC) for the USERID, if applicable.

If the DB2 security exit does not use either the JCTUSER or the ASXBUSER fields as the source for the USERID, the effective AUTHID SET most likely will be the primary AUTHID plus the secondary AUTHIDs of the USERID of the D2-TE started task name, depending upon how the DB2 security exit is implemented.

DB/TOOLS Security Considerations

There are currently no DB/TOOLS specific considerations.
Introduction

This appendix contains the commands that you can issue to the Common Interface (CI).

Chapter Contents

Introduction .................................................. 153
Automatic Startup of the Common Interface (CI) .................. 155
CI Commands ...................................................... 156
Subtask Commands ........................................... 159
Historical Online Collector Commands ................................. 163
Event Collection Manager Parameters ................................. 165
Background about the CI

The CI provides a centralized facility that allows multiple OMEGAMON II for DB2 sessions to execute in a single MVS address space.

The CI accepts commands from either a member of its parameter library or an MVS operator’s console. With these commands you can do the following:

- Start OMEGAMON II monitoring sessions with a different set of startup parameters for each session. For example, each session can pass the address of a different dedicated local 3270 terminal to use as its output device.

- Start the OMEGAMON II VTAM application, OBVTAM. Once this program is running under the CI, any 3270 terminal in your VTAM network (depending on authorization) can log on and have its own OMEGAMON II session.

- Start the historical collector to gather DB2 data for later review.

- Display all programs currently running under the CI.

- Stop any program running under the CI.

- Obtain a short description of all CI functions.

- Modify the options that specify the types of data the historical collector is to collect.

Graphical overview of the Common Interface

FIGURE 8. Common Interface Facility

![Common Interface Facility Diagram]
Automatic Startup of the Common Interface (CI)

Candle recommends that you start the Common Interface (CI) automatically after DB2 is started, since the CI must be active before OMEGAMON II can start any sessions. To do this, add a start command for the CI started task procedure to SYS1.PARMLIB(COMMNDCc).

When you start the CI, it automatically executes the series of commands in the rhilev.RKD2PAR member that has the same name as the CI’s started task name. This member and the individual members containing the start commands to be executed are generated by the customization CLIST based on your input.

**Note:** If you have customized all the OMEGAMON II components that you will be using and if you requested automatic startup, you will not need to manually issue the MVS MODIFY commands described in “Verifying the Configuration” on page 71.

CI start commands

The start commands that may have been included in the startup member are described below:

- EXEC RDEDcccc automatically starts a dedicated OMEGAMON session.
- EXEC RVTMcccc automatically starts the OMEGAMON II VTAM application.
- EXEC EMGRssss automatically starts the event collection manager.
- EXEC sssssccc automatically starts the object analysis collector.
- EXEC STRTcccc automatically starts the online historical collector for DB2 subsystem cccc. One of these EXEC statements (and the corresponding start command member) will be generated for each DB2 subsystem for which online historical collection and automatic startup was requested.

After the CI has performed all actions defined in this special member, you can issue other CI commands through the MVS MODIFY facility.
CI Commands

You can issue commands to the CI from the MVS operator console or from an EXEC member. You must precede all commands from the console with the MODIFY command and the modify ID. You can start the commands in an EXEC member in any column as long as you complete the command before column 72. You can continue the command by placing any nonblank character in column 72.

Command syntax

The CI command syntax is

\[ \text{command task,command subtask,subcommand} \]
CI commands

The following lists commands you can issue to start, stop, modify, and get help for the CI.

START command

Use the START command to start the CI:

To start the started task for OMEGAMON II for DB2, issue the following command:

```
S cccccccc
```

where `ccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

*Note:* You may specify `S` or `START`.

STOP command

Use the STOP command to stop the CI.

To stop CI, issue the following command:

```
P cccccccc
```

where `ccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

*Note:* You may specify `P` or `STOP`.

MODIFY command

Use the MODIFY command to make changes to the CI.

To modify the CI, issue the following command:

```
F cccccccc
```

where `ccccccc` is the started task name that you specified for OMEGAMON II for DB2 using CICAT.

*Note:* You may specify `F` or `MODIFY`.

HELP command

The HELP command displays help information for the CI commands. You can use the HELP command without an operand to find the names of all the commands that are supported by the CI. Issue the HELP command using the format:

```
F cccccccc,HELP
```

where `ccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT.

The HELP command output is shown below.

LOG ‘HELP’ Command
**Syntax: HELP command-name**

You can use the HELP command to display information about any command that the CI processes by following HELP with the name of a specific command. The CI displays the general help text if

- you do not specify a command name
- the CI does not recognize the command name you specified
- the command you specify is the HELP command

To request HELP for a specific command, use the format:

```
F cccccccc,HELP commandname
```

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

**CI comment (*)**

Use an asterisk (*) in column 1 of any line to comment out the text that follows. Use any nonblank character in column 72 to continue the comment. An example of a comment line in an EXEC member is shown below:

```
* THIS COMMENT CAN SAY ANYTHING.
```
Subtask Commands

The following lists commands you can issue to tasks running under the CI. The modify command, F cccccc, must precede subtask commands; it is included in the examples.

START command

Use the START command to start a subtask under the CI. The following members of rhileu.RKD2PAR contain examples of start commands for the tasks that can be started under the CI:

- RDEDcccc (OMEGAMON II for DB2 dedicated monitoring session)
- RVTMcccc (OMEGAMON II for DB2 OBVTAM initialization)

Most of the parameters you can specify have defaults derived from the CI or from the task itself.

To issue the START command for a CI subtask, use the format:

\[ F \text{ cccccccc},S \text{ subtaskname},\text{aaaaaaaa} \]

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT and aaaaaaaa are the parameters for the subtask.

**Note:** You may specify S or START.

An example of using the START subtask command to start the Historical Collector subtask (H2WLMGR) to collect trace data from the DB2 subsystem D41X:

\[ F \text{ cccccccc},S \text{ H2WLMGR,OPTION=COPT cccc} \]

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT and cccc is the DB2 subsystem ID.

STOP command

Use the STOP command to stop any subtask under the CI.

To issue the STOP command for a subtask of the CI, such as OBVTAM, use the format:

\[ F \text{ cccccccc},P \text{ subtaskID} \]

where cccccccc is the started task name you specified for OMEGAMON II for DB2, using CICAT and subtaskID is the ID for the sub task.

You must specify a task ID with the STOP command to stop a subtask of the CI, such as OBVTAM. To find this ID, use the DISPLAY command described in “DISPLAY command” on page 160, or the LIST command described in “LIST command” on page 162.

An example of using the STOP subtask command to stop the Historical Collector subtask (H2WLMGR) from collecting trace data from the DB2 subsystem D41X follows.

\[ F \text{ cccccccc},P \text{ H2D41X} \]
Subtask Commands

Where cccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

**Note:** You may specify P or STOP.

**MODIFY command**

Use the MODIFY command to change any subtask under the CI.

To issue the MODIFY command for a subtask of the CI, such as OBVTAM, use the format:

```
F cccccc,F subtaskID
```

where cccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

**Note:** You may specify F or MODIFY.

**DISPLAY command**

The DISPLAY command lists all tasks that are currently active. The CI displays an internal ID along with the program name of the task. To issue the DISPLAY command, use the format:

```
F cccccc,DISPLAY
```

where ccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

An example of output from the DISPLAY command follows.

```
CIO543: THE FOLLOWING TASK IDS ARE ACTIVE:
CIO594  ID=O2U518      PROGRAM=KO2OINTB
CIO594  ID=O2U520      PROGRAM=KO2OINTB
CIO594  ID=OBVTAM      PROGRAM=KOBVTAM
CIO594  ID=H2          PROGRAM=KO2HWLMB
CIO594  ID=EVENTMGR    PROGRAM=KO2EINTB
```

Each OMEGAMON II task has a unique ID. OMEGAMON II sessions running under the VTAM application program OBVTAM do not have separate IDs. In dedicated mode, OMEGAMON II has the task ID of O2Ucuu, where cuu is the dedicated terminal address. The online historical data collector uses H2cccc as its task ID.

**EXEC command**

Use the EXEC command to process a predefined set of commands stored as a member in the *rhilev.RKD2PAR* library. Enter this command via MODIFY, or as a command in an EXEC member to process another predefined set of commands. Use the format shown below to issue the EXEC command.

```
EXEC cccccc
```

Replace cccccc with the name of the member in the *rhilev.RKD2PAR* to be executed.
The EXEC member contains commands such as START, STOP, and even another EXEC command. When an EXEC command is processed inside another EXEC member, it is as if all of the commands of the other EXEC member were placed into the calling EXEC member in the same position as the calling command.

For example, if you enter the following command:

```
F cccccccc,EXEC MEMBERA
```

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

and if MEMBERA contains the following commands:

```
LOG *** Execute the commands in member B ***
EXEC MEMBERB
```

and MEMBERB contains the following commands:

```
LOG *** O2/DB2 VTAM common interface START - APPL=OBVTAM ***
START OBVTAM,OM=O2INIT,DB2=$DB2,APPL=OBVTAM,UMAX=05
```

then the effect of entering EXEC MEMBERA would be the same as if you entered the following commands:

```
F cccccccc,LOG
*** Execute the commands in member B ***
F cccccccc,LOG *** O2/DB2 VTAM common interface START - APPL=OBVTAM ***
F cccccccc,START OBVTAM,OM=O2INIT,DB2=$DB2,APPL=OBVTAM,UMAX=05
```

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

The CI can process a maximum of 10 EXEC commands at one time. This helps minimize EXEC loops, where A executes B and B executes A.

### IF command

The IF command conditionally processes the EXEC, START, or STOP command that follows it on the same command input. Use the IF command to invoke the same startup member in the `rhilev.RKD2PAR` library to control copies of OMEGAMON that run with different DB2 subsystems. This is a convenient way to cut down on the installation and maintenance effort in certain environments.

The IF command lets you test several different values to determine whether the command that follows its THEN keyword will execute. The format of the IF command is:

```
IF SMFID=CCCC       THEN P     ...
CPUID=CCCCCCCCCCCCC  S     ...
STOP ...
START ...
START ...
```
**Note:** You may specify P (or STOP) or S (or START).

The output of the IF command depends upon the success of its tests.

### Table 11. IF Command Test Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPUID</td>
<td>ccccccccccccc</td>
<td>The 12-character hardware CPU identifier of the machine.</td>
</tr>
<tr>
<td></td>
<td>ccccc</td>
<td>(You can find this identifier on the first page of a dump.)</td>
</tr>
<tr>
<td></td>
<td>The 6-character CPU serial number of the machine.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cccccc</td>
<td>The CPU serial number of the machine. It is available on the RMF CPU report.</td>
</tr>
<tr>
<td></td>
<td>The CPU serial number of the machine. It is available on the RMF CPU report.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For a multiprocessor, OMEGAMON compares the CPUID with all those within the multiprocessing complex.</td>
<td></td>
</tr>
<tr>
<td>SMFID</td>
<td>cccc</td>
<td>The SMF identifier of the MVS system you are executing upon. This information is in SYS1.PARMLIB(SMFPRMNN).</td>
</tr>
</tbody>
</table>

If the test is successful, OMEGAMON issues the normal messages for the conditionally processed command. If the test fails because of an invalid value in the command, OMEGAMON issues a message indicating this.

**LIST command**

LIST is an alternate name for the DISPLAY command. The LIST command displays all tasks that are currently active under the CI.

For a complete description of this command and its output, see “DISPLAY command” on page 160.

**LOG command**

The output from the LOG command looks exactly like its input. You can use the LOG command in an EXEC member to display messages at the console while the CI is processing the EXEC member.

For example, if you type the following LOG command in an EXEC member called O2PARM05, the CI displays the line at the system console when it executes the EXEC member, O2PARM05.

```
LOG *** Processing O2PARM05 ***
```

**WAIT command**

Use the WAIT command to have the CI wait \( nn \) seconds before issuing the next command in the input file. This command can be used with functions that cannot be started until another function has been initialized. The value for \( nn \) can be from 1 to 99.

For example, the following command would cause the CI to wait 10 seconds before issuing the next command in the input file.

```
WAIT 10
```
**Historical Online Collector Commands**

The following lists commands you can issue to the historical online collector running under the CI. The modify command, F cccccc, must precede these commands, and is included in the examples.

**LISTH2DS command**

The LISTH2DS command lists the historical (VSAM) datasets that are currently defined in the collection options member. It also displays the status of each dataset and the amount of space it is using.

Use the following format to issue the LISTH2DS command:

`F cccccccc,F H2cccc,LISTH2DS`

where `ccccccc` is the started task name you specified for OMEGAMON II for DB2 using CICAT and `cccc` is the ID of the DB2 subsystem from which data is being collected.

An example of the output of this command follows:

```
O20561    H2 DATA SET                      STATUS           %FULL
O20561  CANDLE.DB2P.VSAM01  AVAIL             20.1
CANDLE.DB2P.VSAM02                 AVAIL             20.1
CANDLE.DB2P.VSAM03                 ACTIVE 25.5
```

**SWITCH command**

If you use VSAM datasets to store historical data, the online collector can switch and archive datasets automatically when the dataset it is using becomes full. The SWITCH command allows you to perform this operation manually, whenever you wish.

To issue the SWITCH command, use the format:

`F cccccccc,F H2cccc,SWITCH`

where `ccccccc` is the started task name that you specified for OMEGAMON II for DB2 using CICAT and `cccc` is the ID of the DB2 subsystem that the collector is monitoring.

**Important**

Before you use this command, use the LISTH2DS command to check the availability of your VSAM datasets. If you execute SWITCH when you have no available datasets, no data will be saved until a dataset becomes available.

Refer to the *OMEGAMON II for DB2 Historical Reporting Guide* for more information on switching and archiving datasets.
VARY command

While the historical data collector is active, you can change the collection options by issuing the following command:

\[ F \text{cccccccc}, F \text{H}2\text{aaaa}, \text{VARY OPTION=} \text{membername} \]

where \text{aaaa} is the ID of the DB2 subsystem you want to monitor, and where \text{cccccccc} is the name of the Classic started task (in \text{rhilev.RKD2PAR}) that contains the desired collection keyword specifications.

The new collection specifications will take effect after the current collection interval has expired.
Event Collection Manager Parameters

The following lists parameters for the event collection manager running under the CI.

**DB2=parameter**

This parameter specifies ID(s) of the DB2 subsystem(s) to be started, modified, or stopped by the object and volume analysis collectors. If more than one DB2 ID is specified, the string must be enclosed in parentheses.

You can use this parameter with the Start, and Stop commands.

```
F cccccccc,F EVENTMGR,START DB2=(DSN,DB2)
F cccccccc,F EVENTMGR,STOP DB2=(DSN,DB2)
```

where cccccccc is the started task you specified for OMEGAMON II for DB2 using CICAT.

**INTERVAL=parameter**

This parameter specifies the interval for object and volume analysis collectors. Only one interval value may be specified, and the value must be between 1 and 1440. You can use this parameter with the Start (S) and Modify (F) commands:

```
F cccccccc,F EVENTMGR,START DB2=DSN,INTERVAL=15
```

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.

**THREAD=parameter**

This parameter specifies whether or not thread information is to be collected by object analysis collectors. You may specify either YES (default) or NO. You can use this parameter with the Start (S) and Modify (F) commands:

```
F cccccccc,F EVENTMGR,START DB2=DSN,THREAD=YES
```

where cccccccc is the started task name you specified for OMEGAMON II for DB2 using CICAT.
Introduction

This appendix provides a list of dataset authorization requirements for the following:

- Common Interface (CI)
- CUA interface
- Users

Appendix Contents

Dataset Authorization Requirements ........................................... 168
Dataset Authorization Requirements

This section provides information on dataset authorization requirements.

Common Interface authorization requirements

The Common Interface requires the following dataset authorization:

<table>
<thead>
<tr>
<th>Dataset Name</th>
<th>Dataset Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Read</td>
</tr>
<tr>
<td>rhilev.RKANMOD</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKO2HELP</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2PAR</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2PAR</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKO2PROC</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKO2PCSV</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKO2SNAP</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKO2PROF</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKOBBHELP</td>
<td>x</td>
</tr>
<tr>
<td>DB2 SYSUTIL</td>
<td>x</td>
</tr>
</tbody>
</table>

Note: The CI must have control access to the SYSUTIL dataset if an external security package such as RACF is used.

CUA interface authorization requirements

The CUA interface requires the following dataset authorization:

<table>
<thead>
<tr>
<th>Dataset Name</th>
<th>Dataset Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Read</td>
</tr>
<tr>
<td>rhilev.RKANCMN</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANHNU</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANMOD</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANMODL</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2PAR</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANPENU</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2NAM</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2TDB</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2VLOG</td>
<td>x</td>
</tr>
</tbody>
</table>
D2-TE interface requirements

The D2-TE interface requires the following dataset authorization:

<table>
<thead>
<tr>
<th>Dataset Name</th>
<th>Dataset Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB/TOOLS MSGS</td>
<td>x</td>
</tr>
<tr>
<td>DB/TOOLS LOAD</td>
<td>x</td>
</tr>
<tr>
<td>DB/EXPLAIN System PDS</td>
<td>x</td>
</tr>
<tr>
<td>DB/EXPLAIN Log PDS</td>
<td>x</td>
</tr>
<tr>
<td>DB/EXPLAIN User PDS</td>
<td>x</td>
</tr>
<tr>
<td>DB/TOOLS System Profile Dataset</td>
<td>x (optional)</td>
</tr>
<tr>
<td>DB/TOOLS Group (Auxiliary) Profile Dataset</td>
<td>x (optional)</td>
</tr>
</tbody>
</table>

User authorization requirements

The TSO/ISPF user requires the following dataset authorization:

<table>
<thead>
<tr>
<th>Dataset Name</th>
<th>Dataset Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>rhilev.RKANCHT</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANCLI</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANISP</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANMOD</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANTENU</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANSAMV</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKD2SAM</td>
<td>x</td>
</tr>
<tr>
<td>rhilev.RKANSAS</td>
<td>x</td>
</tr>
</tbody>
</table>
Dataset Authorization Requirements
Introduction

Based upon your input on the customization CLIST panels, OMEGAMON allocates runtime datasets and generates tailored customization jobs, sample jobs, and parameter specifications. The following sections describe this output in detail.

Note: You can go back and regenerate the CLIST specifications at any point during the customization process.

Appendix Contents

Job Tailoring. ................................................................. 172
Job Tailoring

The job tailoring portion of the customization CLIST generates tailored parameter specifications, customization jobs, or sample jobs, based on your input. Any JCL or parameter specification member that requires DB2-specific information will have the DB2 subsystem ID (cccc) specified for that customization CLIST invocation as a suffix in the member name.

Tailored parameters in RKD2PAR

Table 12: Tailored Parameters in RKD2PAR on page 172 describes the members created in rhilev.RKD2PAR.

Table 12. Tailored Parameters in RKD2PAR

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDEDcccc</td>
<td>Contains the command necessary to start OMEGAMON II in dedicated mode. This member will be created or replaced when the realtime component is customized if dedicated mode is selected. Variable cccc represents the unit address.</td>
</tr>
<tr>
<td>RVTMcccc</td>
<td>Contains the command necessary to start OMEGAMON II in VTAM mode. Variable cccc represents the DB2 subsystem ID. This member will be created or replaced when the realtime component is customized if VTAM mode is selected.</td>
</tr>
<tr>
<td>KD2COLLP</td>
<td>Contains the CUA interface collector session parameters. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INDM</td>
<td>Contains the CUA interface CT/Engine parameter. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INNAF</td>
<td>Contains the CUA interface CT/Engine parameter. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INNAM</td>
<td>Contains the CUA interface security facility options. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INTB</td>
<td>Contains the CUA interface table definition. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INVLG</td>
<td>Contains the CUA interface view log definition. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2INVPO</td>
<td>Contains the CUA interface CT/Engine parameter. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2SYSIN</td>
<td>Contains the CUA interface address space start-up parameters. This member will be created or replaced when runtime members are created.</td>
</tr>
</tbody>
</table>

Event Collection Manager
Table 12. Tailored Parameters in RKD2PAR

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMGRssss</td>
<td>Contains the command necessary to start the OMEGAMON II event collection manager. Variable ssss represents the MVS system ID. This member will be created or replaced when the event collection manager component is customized.</td>
</tr>
<tr>
<td>sssscccc</td>
<td>Contains the command necessary to start the object analysis collector. This member will be created or replaced when the event collection manager component is customized.</td>
</tr>
</tbody>
</table>

**Historical Component**

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COPTcccc</td>
<td>Collector options used by the online data collector. This member will be created or replaced when the historical component is customized and use of the online data collector is specified.</td>
</tr>
<tr>
<td>STRTcccc</td>
<td>Contains the command necessary to start the online data collector using the options specified during the customization process. This member will be created or replaced when the historical component is customized and use of the online data collector is specified.</td>
</tr>
<tr>
<td>ARCVcccc</td>
<td>The JCL the collector reads when the archive job is submitted. (The sample JCL is in RKD2SAM.)</td>
</tr>
</tbody>
</table>

**Tailored jobs in RKD2SAM**

Table 13: Tailored Jobs in RKD2SAM on page 173 contains the members created in rhilev.RKD2SAM and their descriptions.

Table 13. Tailored Jobs in RKD2SAM

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cccccc</td>
<td>Contains the CUA interface VTAM major node definition. This member name will be the major node name for VTAM specified using CICAT. This member must be moved to SYS1.VTAMLST. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>cccccc</td>
<td>Contains the sample OBTAM application major node definition. This member name will be the major node name for VTAM specified using CICAT. This member will be created or replaced when the realtime component is configured.</td>
</tr>
<tr>
<td>KO2ACF2A</td>
<td>Sample JCL to assemble and link-edit the OMEGAMON realtime monitor user security exit for ACF2 users. This member will be created only if it does not exist.</td>
</tr>
<tr>
<td>KO2ACF2X</td>
<td>Model of an ACF2 exit routine. (This member is provided only as an example.)</td>
</tr>
<tr>
<td>KO2JSNAP</td>
<td>Sample JCL to delete and reallocate the OMEGAMON SNAP dataset. This member will be created only if it does not exist.</td>
</tr>
</tbody>
</table>
Table 13. Tailored Jobs in RKD2SAM

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KO2RACFA</td>
<td>Sample JCL to assemble and link-edit the OMEGAMON realtime monitor user security exit for RACF users. This member will be created only if it does not exist.</td>
</tr>
<tr>
<td>KO2RACFX</td>
<td>Model of a RACF exit routine. (This member is provided only as an example.)</td>
</tr>
<tr>
<td>KO2SUPD</td>
<td>Sample JCL to run the OMEGAMON security table update program. This member will be created only if it does not exist.</td>
</tr>
<tr>
<td>ccccccccc</td>
<td>JCL used to start the D2-TE interface started task. The member name will be the name specified for the started task on the Classic Interface-DB/EXPLAIN customization panel.</td>
</tr>
<tr>
<td></td>
<td>This member must be moved to your installation’s JES procedure library for started tasks. The member will be either created or replaced every time the customization CLIST is run. Do not start this task. OMEGAMON II for DB2 starts this task on an as needed basis via an ASCRE. Note: When you are using an external security package such as RACF or ACF2, the authorization identifier is usually the same as the name of the OMEGAMON II started task. In such cases, the started task name must be set to 7 characters or less. External security packages can use authids other than the OMEGAMON II started task name. This happens when the security package alters the contents of the field ASXBUSER in the MVS control block ASXB. If you have questions about the authid, contact your security administrator.</td>
</tr>
<tr>
<td>USSTABLE</td>
<td>Contains the sample USSTABLE definition. This member will be created or replaced when the realtime component is customized if VTAM mode or TSO/ISPF mode is selected.</td>
</tr>
<tr>
<td>cccccccc</td>
<td>Contains the started task procedure for the D2 CUA. This member will be created or replaced when CUA configuration is selected.</td>
</tr>
<tr>
<td></td>
<td><strong>CUA Interface</strong></td>
</tr>
<tr>
<td></td>
<td>cccccccc</td>
</tr>
<tr>
<td></td>
<td><strong>Event Collection Manager and ATF</strong></td>
</tr>
<tr>
<td>ABNDccccc</td>
<td>JCL to bind the plan needed by ATF. This member will be created or replaced when ATF is customized.</td>
</tr>
<tr>
<td>ACRTccccc</td>
<td>JCL to create a VSAM dataset for use by the application trace facility (ATF). This member will be created or replaced when ATF is customized.</td>
</tr>
<tr>
<td>AGRTeccc</td>
<td>JCL to grant DB2 authority needed by ATF. This member will be created or replaced when ATF is customized.</td>
</tr>
<tr>
<td></td>
<td><strong>Historical Component</strong></td>
</tr>
<tr>
<td>EXTLccccc</td>
<td>Sample JCL used to invoke the data extractor in load mode (for example, data will be loaded into DB2 tables using the DB2 LOAD utility). This member will be created or replaced when the historical component is customized and DB2 tables are used to store historical data.</td>
</tr>
</tbody>
</table>
### Job Tailoring

#### Table 13. Tailored Jobs in RKD2SAM

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HBND</td>
<td>JCL to bind the plan needed for the historical functions. This member will be created or replaced when the historical component is customized.</td>
</tr>
<tr>
<td>HCRV</td>
<td>JCL used to create DB2 tables that will be used by the online collector, data extractor, or data summarizer. This member will be created or replaced when the historical component is customized, and DB2 tables are used to store historical data.</td>
</tr>
<tr>
<td>HGRT</td>
<td>JCL to grant DB2 authority needed for the historical functions. This member will be created or replaced when the historical component is customized.</td>
</tr>
<tr>
<td>EXTI</td>
<td>Sample JCL used to invoke the data extractor in insert mode (for example, data will be inserted into DB2 tables using SQL INSERT statements). This member will be created or replaced when the historical component is customized and DB2 tables are used to store historical data.</td>
</tr>
<tr>
<td>REPT</td>
<td>Sample JCL used to invoke the historical reporter. This member will be created or replaced when the historical component is customized.</td>
</tr>
<tr>
<td>LDCN</td>
<td>Load control statements needed to load tables using data extractor output. This member will be created or replaced when the historical component is customized and DB2 tables are used to store historical data.</td>
</tr>
<tr>
<td>SUMM</td>
<td>Sample JCL used to invoke the data summarizer. This member will be created or replaced when the historical component is customized.</td>
</tr>
</tbody>
</table>

**Miscellaneous**

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACRT</td>
<td>Contains JCL that can be modified to create a storage VSAM LDS dataset for ATF and historical components.</td>
</tr>
<tr>
<td>KO2JGTS</td>
<td>Contains JCL that can be modified to prepare a GTF dataset for input to the historical extractor.</td>
</tr>
<tr>
<td>KO2JSMFS</td>
<td>Contains JCL that can be modified to prepare an SMF dataset for input to the historical extractor.</td>
</tr>
</tbody>
</table>
Tailored parameters in RKANCMD

Table 14: Tailored Parameters in RKANCMD on page 176 contains the members created in `rhilev.RKANCMD` and their descriptions.

Table 14. Tailored Parameters in RKANCMD

<table>
<thead>
<tr>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CUA Interface</strong></td>
<td></td>
</tr>
<tr>
<td>KD2CASE</td>
<td>Contains the CUA interface display output fold option. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2CMDS</td>
<td>Contains the CUA interface startup commands. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2OPER</td>
<td>Contains the CUA interface command to start the operator facility. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2START</td>
<td>Contains the CUA interface command to initialize the CUA interface. This member will be created or replaced when runtime members are created.</td>
</tr>
<tr>
<td>KD2VTP</td>
<td>Contains the CUA virtual terminal definition. parameters. This member will be created or replaced when runtime members are created.</td>
</tr>
</tbody>
</table>
Introduction

You can store collected, extracted, and summarized data in VSAM datasets. This method requires that you define and initialize your datasets to store DB2 data. OMEGAMON generates sample JCL in rhilev.RKD2SAM(HCRTccc) to do this at customization time. In addition, rhilev.RKD2SAM(ACRTccc) contains sample JCL that you can use to define and initialize the datasets.

**Note:** You must have chosen VSAM as the storage type you want to use.

Appendix Contents

VSAM Dataset Allocation Guidelines .......................... 178
Table 15: VSAM Dataset Allocation Guidelines on page 178 provides guidelines for sizing your VSAM datasets, based on average record size.

Table 15. VSAM Dataset Allocation Guidelines

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Record Length</th>
<th>Avg.* No. of Rec./CI</th>
<th>3380 Device</th>
<th>3390 Device</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>No. of CI/Cyl.</td>
<td>Avg. No. of Rec./Cyl.</td>
<td>No. of CI/Cyl.</td>
</tr>
<tr>
<td>STATISTICS</td>
<td>1338</td>
<td>4</td>
<td>174</td>
<td>532</td>
</tr>
<tr>
<td>Buffer Pool</td>
<td>184</td>
<td>29</td>
<td>174</td>
<td>3870</td>
</tr>
<tr>
<td>Group Buffer Pool</td>
<td>124</td>
<td>43</td>
<td>174</td>
<td>5743</td>
</tr>
<tr>
<td>Package</td>
<td>256</td>
<td>21</td>
<td>174</td>
<td>2782</td>
</tr>
<tr>
<td>DDF</td>
<td>184</td>
<td>29</td>
<td>174</td>
<td>3870</td>
</tr>
<tr>
<td>†Average Record</td>
<td>2218</td>
<td>2</td>
<td>174</td>
<td>321</td>
</tr>
<tr>
<td>ACCOUNTING</td>
<td>1208</td>
<td>4</td>
<td>174</td>
<td>590</td>
</tr>
<tr>
<td>Buffer Pool</td>
<td>80</td>
<td>67</td>
<td>174</td>
<td>8902</td>
</tr>
<tr>
<td>Group Buffer Pool</td>
<td>44</td>
<td>121</td>
<td>174</td>
<td>16185</td>
</tr>
<tr>
<td>Package</td>
<td>256</td>
<td>21</td>
<td>174</td>
<td>2782</td>
</tr>
<tr>
<td>DDF</td>
<td>248</td>
<td>21</td>
<td>174</td>
<td>2872</td>
</tr>
<tr>
<td>†Average Record</td>
<td>1880</td>
<td>3</td>
<td>174</td>
<td>379</td>
</tr>
<tr>
<td>ACCOUNTING SUMMARY</td>
<td>1510</td>
<td>4</td>
<td>174</td>
<td>472</td>
</tr>
<tr>
<td>Buffer Pool</td>
<td>80</td>
<td>67</td>
<td>174</td>
<td>8902</td>
</tr>
<tr>
<td>Group Buffer Pool</td>
<td>44</td>
<td>121</td>
<td>174</td>
<td>16185</td>
</tr>
<tr>
<td>Package</td>
<td>256</td>
<td>21</td>
<td>174</td>
<td>2782</td>
</tr>
<tr>
<td>DDF</td>
<td>264</td>
<td>20</td>
<td>174</td>
<td>2698</td>
</tr>
<tr>
<td>†Average Record</td>
<td>2182</td>
<td>2</td>
<td>174</td>
<td>326</td>
</tr>
<tr>
<td>DSNZPARM</td>
<td>4202</td>
<td>1</td>
<td>174</td>
<td>169</td>
</tr>
<tr>
<td>Audit Fail</td>
<td>430</td>
<td>12</td>
<td>174</td>
<td>1656</td>
</tr>
<tr>
<td>Audit Control</td>
<td>380</td>
<td>14</td>
<td>174</td>
<td>1874</td>
</tr>
<tr>
<td>Audit DDL</td>
<td>406</td>
<td>13</td>
<td>174</td>
<td>1754</td>
</tr>
<tr>
<td>Audit DML</td>
<td>146</td>
<td>36</td>
<td>174</td>
<td>4878</td>
</tr>
</tbody>
</table>
### VSAM Dataset Allocation Guidelines

<table>
<thead>
<tr>
<th>Record Type</th>
<th>Record Length</th>
<th>Avg.* No. of Rec./CI</th>
<th>3380 Device</th>
<th>3390 Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit Bind</td>
<td>514</td>
<td>10</td>
<td>174</td>
<td>1385</td>
</tr>
<tr>
<td>Audit ChangeID</td>
<td>398</td>
<td>13</td>
<td>174</td>
<td>1789</td>
</tr>
<tr>
<td>Audit Utility</td>
<td>154</td>
<td>35</td>
<td>174</td>
<td>4624</td>
</tr>
</tbody>
</table>

† The average record is calculated using the values from a main section (Accounting/Statistics) with values from 2 buffer pool and 2 package sections.
* Assuming 30% data compression ratio.
**Introduction**

To provide support for OMEGAMON sessions under more than one TSO (or ISPF), you must install VTM1 in every VTAM domain that controls a TSO.

VTM1 uses a virtual terminal for each OMEGAMON session; VTPOOL defines this virtual terminal pool. Normally, each installation of VTM1 includes a VTPOOL definition. The following sections describe how multiple VTM1 installations can share a single VTPOOL definition.

**Chapter Contents**

Sharing VTPOOL .................................................. 182
Sharing VTPOOL

Sample Network

Assume that the network looks like this:

FIGURE 9. Sample Network

In this example, there are two VTAM domains: Host Subarea A (HSAA) and Host Subarea B (HSAB). Host Subarea A runs OMEGAMON and TSO (TSOA). Host Subarea B runs TSO (TSOB). Assume that OMEGAMON users who use ISPF or TSO mode must use the local TSO. This means that users whose terminals are controlled by VTAM domain HSAA must log onto TSOA, and users whose terminals are controlled by VTAM domain HSAB must log onto TSOB.
Defining the virtual terminal pool to VTM1

In the sample network described in Figure 9 on page 182, assume that a pool of 10 virtual terminals is required for each host subarea. Figure 10 on page 183 contains the $VTAPPL statement required to define this virtual terminal pool (VTPOOL) to VTM1.

FIGURE 10. VTPOOL Definition Statement

Defining the virtual terminal pool (VTPOOL) to VTAM

After defining VTPOOL to VTM1, you must define the virtual terminals in VTPOOL to each VTAM domain. To do so, take advantage of the capability to define the local name and the network name separately. The local name is defined by the ACBNAME keyword in the VTAM APPL definition statement; the network name is defined by the name field in the VTAM APPL definition statement. In the sample VTAM APPL definitions that follow, the HSAA network names differ from those of HSAB, but the local names for each virtual terminal are the same in both host subareas.

VTAM definition statements for host subarea A

Figure 11 on page 183 shows the definition statements for Host Subarea A that correspond to the $VTAPPL definition statement.

FIGURE 11. HSAA VTAM Definition Statements

<table>
<thead>
<tr>
<th>HSAAVTM1</th>
<th>VBUILD</th>
<th>TYPE=APPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSAAVT01</td>
<td>APPL</td>
<td>ACBNAME=OBVTM101,EAS=1</td>
</tr>
<tr>
<td>HSAAVT02</td>
<td>APPL</td>
<td>ACBNAME=OBVTM102,EAS=1</td>
</tr>
<tr>
<td>HSAAVT03</td>
<td>APPL</td>
<td>ACBNAME=OBVTM103,EAS=1</td>
</tr>
<tr>
<td>HSAAVT04</td>
<td>APPL</td>
<td>ACBNAME=OBVTM104,EAS=1</td>
</tr>
<tr>
<td>HSAAVT05</td>
<td>APPL</td>
<td>ACBNAME=OBVTM105,EAS=1</td>
</tr>
<tr>
<td>HSAAVT06</td>
<td>APPL</td>
<td>ACBNAME=OBVTM106,EAS=1</td>
</tr>
<tr>
<td>HSAAVT07</td>
<td>APPL</td>
<td>ACBNAME=OBVTM107,EAS=1</td>
</tr>
<tr>
<td>HSAAVT08</td>
<td>APPL</td>
<td>ACBNAME=OBVTM108,EAS=1</td>
</tr>
<tr>
<td>HSAAVT09</td>
<td>APPL</td>
<td>ACBNAME=OBVTM109,EAS=1</td>
</tr>
<tr>
<td>HSAAVT10</td>
<td>APPL</td>
<td>ACBNAME=OBVTM110,EAS=1</td>
</tr>
</tbody>
</table>

VTAM definition statements for host subarea B

Figure 12 on page 184 shows the definition statements for Host Subarea B that correspond to the $VTAPPL definition statement.
Providing access to VTPOOL

Once you have defined VTPOOL and the virtual terminals in VTPOOL to VTAM, you must assemble and link-edit the VTPOOL definition statements to produce the module shilev.TKANSAM(KOBVTPL). shilev.TKANSAM(KOBVTPL) is used by VTM1 at run time to select a virtual terminal for use prior to starting an OMEGAMON session. Given the sample network described in this section, you must install VTM1 execution-time modules, including the linked edit shilev.TKANSAM(KOBVTPL) module, so that they are available to TSOA and TSOB users. The most convenient method is to place the modules in a library on DASD shared by both host subareas. If this is not possible, you must use separate libraries with identical modules for both systems. You can still perform VTPOOL maintenance from a single master library.

Modifying Virtual Terminal Pool Definitions for TSO/ISPF Mode

If you use TSO or ISPF mode and your runtime environment (RTE) is not sharing libraries with other RTEs or with SMP/E, perform this procedure if you want to modify virtual terminal pool definitions.

1. Define your virtual terminals and LOGMODE names to the VTM1 program by updating RKANSAM dataset member KOBVTPL.

2. Assemble and link the KOBVTPL source using the JCL in RKANSAM dataset member KOBVTPLX. The resulting KOBVTPL load module is stored in the RKANMOD dataset.

3. If you modified the terminal names or the number of terminals:
   - Update the VTAM node list member, KOBVT1AP, in the RKANSAM dataset.
   - Update your VTAMLST controls accordingly.
Introduction

This appendix contains information about the parameters in KD2SYSIN. The parameters are listed in alphabetical order by name.

Appendix Contents

Parameters in KD2SYSIN .................................................. 186
Parameters in KD2SYSIN

AMODE31

AMODE31(Y | N)

Determines whether CT/Engine runs in 31-bit addressing mode. The default is AMODE31(Y).

If MVS/XA is running under VTAM prior to Version 3.1.1 or DFP prior to Version 1.2, code AMODE31(N).

CONFIRM

CONFIRM(15 | n)

Sets the maximum number of seconds between two successive SHUTDOWN commands or MVS STOP (P) commands to terminate the CT/Engine address space.

CONFIRM(0) allows CT/Engine shutdown to begin immediately without an additional, confirming SHUTDOWN command. The default value is CONFIRM(0).

CONFIRM(n) prevents accidental shutdowns by requiring you to confirm the command by entering it a second time within the specified number of seconds.

For example, the default (15 | n) requires you enter SHUTDOWN twice within 15 seconds to terminate the CT/Engine address space.

FRAME

FRAME(512 | n)

Specifies the size, in bytes, of a save area stack. The default is FRAME(900).

CT/Engine preallocates a contiguous area of extended storage that is used as a stack. When a module is called, a portion of the stack is used for the module’s save area and local working storage. If the area fills up, CT/Engine allocates another save area stack. Specifying larger FRAME values enable you to

- produce larger preallocated save area stacks
- avoid dynamic GET and FREE overhead
- leave some storage idle

FRAME(1) turns off all preallocation. In general, FRAME(180) conserves virtual storage without unduly affecting performance. FRAME(900) is the largest useful value.

INITIAL

INITIAL(membername)

Identifies the member that contains CT/Engine initialization commands. Membername is a member in the RKANCMD library. The default is INITIAL(KD2CMDS).
RKANCMD may contain a member named $smfid$, where $smfid$ is the SMF ID of the CPU executing CT/Engine. If this member exists, it is automatically invoked before the command list specified in the INITIAL parameter.

**INITLIST**

**INITLIST**(membername)

Identifies the member of RKANPAR which contains initialization member name overrides. The default is INITLIST(KD2INIT).

Normally, CT/Engine initialization modules read members of RKANPAR whose names are the same as the modules. The member identified by the INITLIST keyword contains statements such as:

```
startup-module-name=override-name
```

Each line of the INITLIST member names a particular startup module and identifies the member of RKANPAR which should be read by that module for its parameters. The valid startup module names vary with the CT/Engine-based products that are installed. The number of startup modules may also change as new features are added or new products become available. As a result, it is not possible to provide a comprehensive list of startup module names.

No wild-card characters are accepted during INITLIST processing. Each override must be coded in full. Override statements must be coded one per line. The statements may appear in any column. Input will be converted to upper case prior to processing. Comment lines are indicated by an asterisk as the first non-blank character. On any line, anything following an asterisk is ignored.

Here is a sample input line:

```
KLVINNAM=NAMINIT0  * Override KLVINNAM parameter member name
```

If multiple override statements for the same startup module are present, the last one will determine the member name to be used. No messages will be issued.

RKANPAR may contain a member named $smfid$, where $smfid$ is the SMF ID of the CPU executing CT/Engine. If this member exists, it will be used as the list of initialization member name overrides if the INITLIST keyword is not coded. When INITLIST is coded, a member of RKANPAR named $smfid$ will not be read even if present in the library.

**LIMIT**

```important```

Do not modify this parameter except under the guidance of a Candle support representative.

```important```

**LIMIT**(16,P | n,P)

**LIMIT**(16,X | n,X)

Specifies the largest block of primary (P) or extended (X) storage that can be allocated. This value is specified in bytes, as a power of 2. For example, if $n$ is 16, the largest block that can be allocated is 65,536 bytes. The default values are:
Parameters in KD2SYSIN

- LIMIT(16,P)
- LIMIT(17,X)

Primary storage is below the 16-megabyte line; extended storage is above the line. To specify values for both primary and extended storage, include the LIMIT parameter twice in RKLVIN:

```
LIMIT(16,X)
LIMIT(17,P)
```

If the LIMIT value is too small and a process in CT/Engine attempts to allocate a block of storage larger than LIMIT specifies, program interruption U0100 or U0200 results. Too large a LIMIT value may waste storage and increase processing overhead.

LGSA

LGSA(Y)

When the LGSA parameter is set to Y, the program issues an error message if the STEPLIB is not APF-authorized. If the RKANMODn libraries in the STEPLIB concatenation are not authorized and LGSA(Y), the program issues the following message.

```
KLVST038 STEPLIB AUTHORIZATION REQUIRED
```

The default value is LGSA(Y).

LOADLIST

LOADLIST(membername)

LOADLIST specifies one or more RKANPAR members containing generic module names, which limits module loading at startup. The default is LOADLIST(KD2LLIST).

LSRPOOL

LSRPOOL(size,count [,0 | hiper])

You must code at least one LSRPOOL parameter in order for CT/Engine to start. The default values are:

- LSRPOOL(2048,8)
- LSRPOOL(4096,32)
- LSRPOOL(32768,3)

It corresponds to the BUFFERS parameter of the BLDVRP macro instruction, and specifies the number of buffers to be made available for each VSAM dataset used by CT/Engine.

size is a buffer (VSAM control interval) size. Valid sizes are 512, 1024, 2048, 4096, 8192, 12288, 16384, 20480, 24576, 28672, and 32768. count is the number of virtual storage buffers of size to be allocated. The minimum is 3. The maximum is 65535, although this may be less, depending on the amount of available virtual storage in the CT/Engine
address space. Buffers will be allocated from extended storage if AMODE31(Y) was coded or defaulted to; from primary storage otherwise.

hiper is the number of hiperspace buffers of size to be allocated. The minimum is 0 (no hiperspace buffers). The maximum is 16777215, although this may be less, depending on the hiperspace storage available to IBM VSAM services. Ensure the CT/Engine address space is non-swappable when you are using hiperspace buffers.

**Note:** IBM restricts hiperspace buffers to multiples of 4K. Do not code hiper for the 512, 1024, or 2048 sizes; an error message will be issued and CT/Engine startup will terminate.

For best storage use, code an LSRPOOL parameter for each different VSAM control interval size that CT/Engine uses:
- index buffer for NAM and table database
- data buffer for NAM and table database
- data buffer for VIEWLOG

Candle distributes the recommended values in the RKLVIN dataset.

**Note:** You must enter LSRPOOLs individually; you cannot string them.

If you receive many KLVVS026 messages identifying buffer contention, increase the number of buffers allocated to the dataset identified in the associated KLVVS021 messages.

### MAXIMUM

```plaintext
MAXIMUM(8192,P | n,P)

MAXIMUM(n,X)
```

MAXIMUM is a storage throttle used to prevent GETMAINs from overallocating and occupying the page dataset with rarely referenced frames. The default values are:
- MAXIMUM(4096,P)
- MAXIMUM(32768,X)

The variable `n` represents the maximum amount (in kilobytes) of primary or extended storage that can be allocated. `X` stands for extended storage (above the 16-megabyte line), and `P` stands for primary storage (below the line).

Set your MAXIMUM value to a value that will allow CT/Engine to continue running without overloading your page volumes when the steady-state MINIMUM value is exceeded.

To use extended storage, you must do both of the following:
- Code the MINIMUM parameter.
- Make sure that MAXIMUM is equal to or greater than MINIMUM + RESERVE.

If MAXIMUM is too large and RESERVE is not large enough to meet your requirements, the address space may run out of virtual storage.
If the value of MAXIMUM is greater than that of MINIMUM, CT/Engine attempts a conditional GETMAIN for the MAXIMUM value minus the RESERVE value (RESERVE defaults to 512,P). If the MAXIMUM value is not satisfied, CT/Engine accepts the amount of storage acquired by the GETMAIN.

**MINIMUM**

MINIMUM(1024,P | n,P)
MINIMUM(8192,X | n,X)

The default values are:
- MINIMUM(521,P)
- MINIMUM(8192,X)

n represents the minimum amount (in kilobytes) of primary or extended storage that can be allocated.

For example, to specify a 16-megabyte above-the-line region, code

MINIMUM(16384,X)

To specify a 32-megabyte above-the-line region, code

MINIMUM(32768,X)

To use extended storage, you must do both of the following:
- Code the MINIMUM parameter.
- Make sure that MINIMUM + RESERVE is less than or equal to MAXIMUM.

Note the following about the default above-the-line region:
- Specified in the IEFUSI and IEALIMIT MVS modules.
- Distributed by IBM as 32 megabytes.
- If smaller than the amount specified for the MINIMUM parameter, do one of the following:
  - Alter the default.
  - Use the REGION parameter as follows:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0K or 0M</td>
<td>All primary and extended storage is available for GETMAIN.</td>
</tr>
<tr>
<td>Up to 16M</td>
<td>Primary region equals the specified value; extended region equals the default.</td>
</tr>
<tr>
<td>Up to 32M</td>
<td>All available region goes to primary storage; extended region equals the default.</td>
</tr>
<tr>
<td>Above 32M</td>
<td>All available region goes to primary storage; specified value goes to extended storage.</td>
</tr>
</tbody>
</table>

In general, Candle recommends REGION=0M.
OPSTART

OPSTART(command)

Specifies an initial CT/Engine command or CLIST to be issued after an operator logs on. Command is a member in the RKANCMD library. The default is OPSTART(KD2OPST).

STGMON

STGMON(15 | n)

Specifies the number of minutes between storage quiesce mode message displays. The default is STGMON(0). Any value between 0 and 120 is valid. A value of 0 results in messages being issued only when a short-on-storage condition is detected or relieved (for example, when the quiesce mode state changes).

WTO

WTO(Y | N)

Determines whether or not CT/Engine issues WTOs. The default is WTO(N).

WTOs write information and exception condition messages to the operator consoles. ALERT messages are always written to the consoles.

Note: WTO(N) will suppress messages written with the WTO SSPL dialog function.

WTORC

WTORC(ALERT,1,11 | type,code,code,...)

Specifies WTO route codes for CT/Engine message types. Specify the WTORC parameter for each CT/Engine message type. The default is WTORC(ALERT,11).

For definitions of the route codes, see IBM’s Supervisor Services and Macro Instructions manual. For information on message types, see Supervisor Services and Macro Instructions Manual.
Symbols

154, 165
$UCHECK macro 121
./PWD command 131
.DSA, authorized command 112
/PWD command 128, 131, 142
  accessing authorized commands 112
  locking to maintain security 142
  relogon feature 116

A

ACF2
  auditing 141
  deleting the realtime monitor to 119
  locking facility 143
  macro library 119
  modifying rules for the realtime monitor 119
  resource class 119
  restricting a command 119
  rule compiler 119
  security 120
ACF2 CA-ACF2 143
  address space
    stopping OMEGAMON
      CUA address space 83
Adobe portable document format 14
AMODE3 1 parameter 186
APF authorization requirements 58
application trace facility (ATF) 87
  verifying customization 87
application trace facility (ATF) ATF 87
applids
  OMEGAMON II VTAM security 149
asterisk (*)
  comment line 126
  disabled commands 139
ATTN key 113
audit
  ACF2 ACFRPT 141
  RACF Report Writer 141
  SMF 129, 140
  supplementation 141
  suppression 141
WTO 128
  AUDIT keyword 128
  AUTHLIB control statements 127
    format 127
  authorization levels 112
  authorization requirements 167
  authorized commands
    access levels 112–??
    list 112
    removing access 113
  authorized screen library 125, 127
  automatic startup 155

C

CA-ACF2
  for CUA interface 148
calling
  conventions 121
  flow 122
Candle Web site 16
Candle-supplied profile 98
CICAT
  help 46
CL/Engine 30
CLIST
  KD2SPF 76, 77, 92
CLIST dataset 62
COMMAND control statements 128
  format 128
commands
  auditing 128
  EXEC 160
  HELP 157
  IF 161
  in ACF2 119
  in RACF 118
  LIST 162
  LISTH2DS 163
  LOG 162
  MODIFY 157
  protection 128, 130
  START 157, 159
  STOP 157, 159
SWITCH 163
  types 139
VARY 164
commands commands 165
comment(*), CI command 158
common component  
  CT/Engine 30
Common Interface  
  started task name 42
  starting and stopping 72
Common Interface (CI) 31, 157, 159, 160, 161, 162, 163, 164
  authorization requirements 168
  automatic startup 155
  dispatching priority 42
  operation 155
Common Interface (CI) CI 165
Common Interface (CI),commands  
  DISPLAY 160
Common Interface (CI);commands comment (*), 158
completing the configuration  
  outside of CICAT 69
configuration  
  CUA interface 65
  process overview 27
  verification 78
configuration with CICAT 47
CONFIRM parameter 186
CONS, authorized command 112
control statements  
  edit listing 137
control statement  
  MINOR 130
control statement edit listing  
  codes 138
control statements 110, 124, 127–133, 138
  AUTHLIB 127
  COMMAND 128
  edit listing 128
  example 133
  format rules 127–??
  LIST 129
  MODULE 130
  PASSWORD 131
  RESET 132
  SMFNUM 133
  UPDATE 133
  control statements edit  
  codes 138
  copying OMEGAMON II across subsystems 161
CT/Engine 30
CUA 81
  dataset considerations 65
CUA interface  
  authorization requirements 168
  CA-ACF2 148
  CA-TOP SECRET 150
  component configuration 65
  migrating from classic 43
  NAM 146
  RACF 149
  starting 81
CUA interface address space  
  starting 81
customization  
  realtime monitor profile 105
customization  
  application trace facility 87
  ATF 85
  Event Manager 85
  Historical 85
  historical component 90
  screen spaces 105
  customizing  
  requirements for each mode 37

D

D2 CUA started task 146
data extractor 33
datasets  
  rhilev.RKANMODL 145, 150
  rhilev.RKD2NAM 149
  rhilev.RKD2PAR 146, 148, 150
  rhilev.RKD2SAM 145, 148, 150
  rhilev.RKD2TDB 65
  rhilev.RKD2VLOG 149
DB/Tools  
  access requirements 61
DB2, determining the ID 143
DCMD, authorized command 112
DCNS, authorized command 112
dedicated mode 37
  logging onto external security 116
  starting OMEGAMON 75
  stopping OMEGAMON 79
Dedicated operating mode 37
defaults  
  exception thresholds 102
delete
  installation-defined profile 108
Delete New/Altered Installation Profile screen 107
DISABLE parameter 128
disk space requirements 167
  customization considerations 61
dispatching priority
    Common Interface 42
DISPLAY command 160

e
edit listing
  codes 138
error messages, custom 141
event collection manager (EVENTMGR)
  starting 88
  stopping 89
exception
  threshold setting 102
exceptions
  changing session options 106
EXEC command 160
Exit Confirmation panel 82
exit routine 110, 120
  calling conventions 121
  calling flow 122
  calling sequence 121
  creating 120
  external security checks 114
  module 130
exiting OMEGAMON II 82
external
  CA-ACF2 148
  CA-TOP SECRET 150
  RACF 149
EXTERNAL keyword 114, 128
external security 112–116
  checking 128
  creating exit routine 120
  exit routine 125, 130
EXTERNAL keyword 128
  implementation procedure 117
interface 121
KO2SUPDI 125
locking feature 142
mapping 121
modifying ACF2 rules 119
modifying RACF rules 118
MODULE 130

options 141
relogon feature 116
external security
  mapping 121

F
FRAME parameter 186

G
GETMAIN 121

H
HELP command 157
help, CICAT 46
high-level qualifier, definition 12
-HILEV-, definition 12
hilev, definition 12
historical component customization 90
historical data collection 32
  changing collection specifications 93
  sort requirements using VSAM 34
  starting the collector 93
  stopping a session 92
  stopping the collector 93
  verifying operation 94
historical data extraction 32
historical data summarization 32

I
ICHERCDE macro call 118
identifier
  determining the DB2 subsystem ID 143
IEBGENER 12
IF command 161
INITIAL parameter 186
INITIAL resources 142
INITLIST parameter 187
installation
  copying OMEGAMON to multiple DB2 subsystems 161
installation-define profile
  create 105
installation-defined profile
  delete 108
  save 107
internal security 112–113
ISPF 74, 79
ISPF mode 37
ISPF operating mode 37
J

JCL
bind ATF plan 87
bind historical plan 90
clear snap dataset 12
grant ATF access 87
grant historical access 90
security update 135
security updates 110
tailoring customization 172

K

KD2SPF 61
KO2ACF2X 120
KO2PRSUPD 110
KO2PRSUPDI 110
KO2RACFX 120
KO2SUPD 125
example 135
KO2SUPDI 124
example 133
KOBSPORT 110, 125
KOBVTAM
load module 114
logon panel 115

L

level
change 125
reset 113
LEVEL keyword 128, 131
LGSA parameter 188
LIMIT parameter 187
LIST command 162
LIST control statement 126, 139
format 129
LIST control statements 129
LISTH2DS command 163
listing 138, 140
control statements edit 137
security file listing 139
security update program 136
LOADLIST parameter 188
locking
use of /PWD command 142
LOG command 162
log off
CUA procedure 82
logging off 82

logging on
external security 116
from OMEGAVIEW 81
KOBVTAM logon panel 115
starting the historical data collector 93
to a TSO/ISPF session 76
to the classic interface 76
to the CUA interface 81
to the historical component 92
using external security 115
verifying configuration 78
LSRPOOL parameter 188

M

macro libraries in security routine 119
MAXIMUM parameter 189
MCHN, authorized command 112
menus
Profile Maintenance 107
migration considerations 90
migrating classic profiles to CUA 43
MINIMUM parameter 190
minor commands
listing 139
protecting 130
MINOR control statement 130
format 130
MLST, authorized command 112
mode
starting OMEGAMON 74
stopping OMEGAMON 79, 80
modes of operation 35
MODIFY
CI command 157
MODULE control statement 130
format 130
MSCN, authorized command 112
multiple user single address space system (MUSASS) 148
MUSASS attribute for ACF2 119
MVS
MODIFY 75, 155
STOP command 72
MVS commands 165
MZAP, authorized command 112

O

O2CMDEX module 136
object analysis 88
Index 197

OCMD, authorized command 112
OMEGAMON reporter 34
online data collector 33
operating mode requirements 37
operating modes 35
operation modes 35
OPSTART parameter 191
OSPC, authorized command 112

P

P (MVS STOP command 72
PAI key 113
panels
Exit Confirmation 82
PARM=in security update 135
password
   access levels 112
   change 125
   updating 141
PASSWORD control statement 131
   format 131
passwords
   using to relogon 116
PDF files, adding annotations 15
PEEK, authorized command 112
PERMIT command 118
prefix definition 12
printing problems 14
product components 26
profiles
   changing exception thresholds 106
   migrating classic profiles to CUA 43
   search order 99
   storage 99
   suffix 98
   types 98
PURGE (MVS STOP command 72

Q

QMF sample reports 91
qualifiers
   high-level, definition 12

R

RACF
   locking /PWD 142
   locking facility 142
   modifying rules for the realtime monitor 118
   Reporting Writer 141
security 120
RDEFINE command 118
realtime
   monitor customization 97
   operating modes overview 35
realtime monitor
   configuration 57
realtime monitor profile customization 105
relogon feature 116
required space requirements 169
requirements
   dataset authorization requirements 169
   resource class (ACF2) 119
   return codes in security 122
RKANMOD
   KOBSSUPD 110
RKD2SAM 120
   KO2ACF2X 120
   KO2RACF 120
   KO2SUPD 125
   KO2SUPDI 110, 124, 125
   KOBSMFRP 140
RKD2TDB 65
RKO2SNAP dataset 12

S

save
   installation-defined profile 107
Save New/Altered Installation Profile Screen 107
SCHN, authorized command 112
screen space customization 105
screens
   authorized 127
security 113, 124, 125, 129, 135, 136, 140, 146, 148, 149, 150
   access levels 112–??
ACF2 rules 119
   audit 125, 128, 133, 140
   considerations 108
   control statements 127–??
   customization choices 145
default 111
   exit for ACF2 validation 148
   exit for CA-TOP SECRET validation 150
   exit module 125
exit routine 110, 120, 125
external 125
file listing 139
interfaces 121
internal 112–??
JCL 110
level 122
listing 125, 129
logging on 110
OMEGAMON II applid 149
RACF 118
resetting defaults 132
sequence of events 110
SMF record 140
table 110
terms 110
TOP SECRET 120
update 133
update job 125
update procedure 124
update program 110
security update program
trace 140
see ACF2. security 151
see control statements. security 151
see RACF. security 151
see Security. exit routine 151
see Security. external security 151
see Security. protection 151
see Security.internal security 151
see ’JCL’.jobs 151
session options, selecting 106
shutdown 83
signing off 82
SLST, authorized command 112
SMF
audit record 129, 133, 140
Record Exits (IEFUS83/84) 140
record number 129, 133
system parameters 140
SMFNUM control statement 129
format 133
SMFNUM control statements 133
snap dataset 12
sort requirements using VSAM for historical reporting 34
SSCN, authorized command 112
START
CI command 157, 159
started task
ACF2 119
D2 CUA 146
RACF 118
started task name 72
starting OMEGAMON
CUA interface 81
dedicated mode 75
event collection manager 88
historical data collector 93
historical reporter 92
ISPF mode 74
TSO mode 74
VTAM mode 73
starting the Common Interface
automatic startup 155
STEPLIB in security update 135
STGMON parameter 191
STOP
CI command 157, 159
MVS PURGE command 72
stopping OMEGAMON
CUA interface address space 83
event collection manager (EVENTMGR) 89
historical data collector 93
in dedicated mode 79
in ISPF or TSO mode 79
VTAM mode 80
stopping OMEGAMON session shutdown 83
stopping OMEGAMON session terminating
OMEGAMON 83
stopping OMEGAMON shutdown 83
stopping OMEGAMON terminating
OMEGAMON 83
stopping OMEGAMON terminating OMEGAMON 83
subsystem ID, determining 143
SWITCH command 163
SYS1.PARMLIB 155
SYSIN in security update 135
SYSLIB in K02SUPD
error 136
SYSLIB in security update 135
SYSPRINT in security update 135
SYSPROC 61
SYSPROC concatenation 62
SYSPROC Concatenation Considerations 61
system access validation 148
SZAP, authorized command 112
T

  table
    listing 129
    modifying 124
tape contents 167
tape contents dataset names 169
tape contents distribution tape 169
tape contents product tape 169
TCMD, authorized command 112
thresholds, exception 102
TKANMAC 119, 121
TO2PRINST
  KO2PRSUPD 110
TOP SECRET 120
trace, active 43
TSO 74, 79
TSO mode 37
TSO operating mode 37

U

U#CHMSG field 122
U#CHTYP field 122
update
  JCL 135
UPDATE control statement 126, 133
    format 133
    with LIST 129
update program 110, 138
  control statement edit listing 137
  header 136
  listing 136
  security file listing 129, 139
  trace 140
user ID termination 123
USER profile option 99
user profiles 98
  saving and deleting 108
user security 142

V

VARY
  CI command 164
  MVS command 73
verifying configuration 78
virtual terminal pool
  modifying definitions 184
  RKANMOD 184
  RKANSAM dataset 184
VOL keyword 127

VSAM datasets 177
  for ATF capture 87
  migrating 44
  preparing datasets 177
  sort requirements for historical reporting 34
VTAM 80, 149
  definition statements 183
  mode 37
  multiple domains 181
VTAM operating mode 37
VTAM,mode
  starting OMEGAMON 73
VTM1
  in multiple VTAM domains 181
VTPool
  defined to VTAM 183
  definition statement 183
  in multiple VTAM domains 181
  OMEGAMON access to 184

W

Web site, Candle 16
WTO audit 128
WTO parameter 191
WTORC parameters 191

X

XMCH, authorized command 112
XMLS, authorized command 112
XMSC, authorized command 112
XMZP, authorized command 112