## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of Figures</td>
<td>7</td>
</tr>
<tr>
<td>List of Tables</td>
<td>9</td>
</tr>
<tr>
<td>Preface</td>
<td>11</td>
</tr>
<tr>
<td>About This Book</td>
<td>12</td>
</tr>
<tr>
<td>Adobe Portable Document Format</td>
<td>14</td>
</tr>
<tr>
<td>Documentation Conventions</td>
<td>15</td>
</tr>
<tr>
<td>What's New</td>
<td>17</td>
</tr>
<tr>
<td>Application Trace Facility</td>
<td>18</td>
</tr>
<tr>
<td>New OMEGAMON II Functions</td>
<td>19</td>
</tr>
<tr>
<td>Documentation Changes</td>
<td>20</td>
</tr>
<tr>
<td>Chapter 1: Operational Essentials</td>
<td>21</td>
</tr>
<tr>
<td>OMEGAMON II User Interfaces and Components</td>
<td>22</td>
</tr>
<tr>
<td>Software Requirements</td>
<td>25</td>
</tr>
<tr>
<td>Modes of Operation</td>
<td>26</td>
</tr>
<tr>
<td>INFO-line Format</td>
<td>28</td>
</tr>
<tr>
<td>Entering Commands</td>
<td>30</td>
</tr>
<tr>
<td>Command Help</td>
<td>33</td>
</tr>
<tr>
<td>Chapter 2: Operational Commands</td>
<td>37</td>
</tr>
<tr>
<td>Command Modifiers and Help Symbols</td>
<td>38</td>
</tr>
<tr>
<td>Operational INFO-line and Immediate Commands</td>
<td>40</td>
</tr>
<tr>
<td>Chapter 3: User Profile Facility</td>
<td>81</td>
</tr>
<tr>
<td>Summary of Profile Definition Commands</td>
<td>82</td>
</tr>
<tr>
<td>Multi-line Input</td>
<td>85</td>
</tr>
<tr>
<td>Profile Maintenance Commands</td>
<td>86</td>
</tr>
<tr>
<td>Controlling Session and Display Options</td>
<td>87</td>
</tr>
<tr>
<td>Setting Print Output Options</td>
<td>89</td>
</tr>
<tr>
<td>Setting Color Options</td>
<td>95</td>
</tr>
<tr>
<td>Setting Operational Parameters</td>
<td>101</td>
</tr>
<tr>
<td>Setting Installation Performance Options</td>
<td>103</td>
</tr>
<tr>
<td>Chapter 4: Commands for Customizing and Creating Exceptions</td>
<td>105</td>
</tr>
<tr>
<td>Customizing and Creating Exceptions for User Profiles</td>
<td>106</td>
</tr>
<tr>
<td>Customizing Profiles for Users Who Log On through the CUA Interface</td>
<td>107</td>
</tr>
<tr>
<td>Commands for Controlling Exception Analysis</td>
<td>108</td>
</tr>
<tr>
<td>Resource Contention Exception Commands</td>
<td>123</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>System Information</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Real Memory Information</td>
<td>128</td>
</tr>
<tr>
<td>Common Storage Area Virtual Memory Analysis</td>
<td>129</td>
</tr>
<tr>
<td>CSA Storage Isolation Information</td>
<td>131</td>
</tr>
<tr>
<td>IMS Control Block Display Commands</td>
<td>132</td>
</tr>
<tr>
<td>Dynamic Control Block Table Analysis</td>
<td>138</td>
</tr>
<tr>
<td>Dynamic Control Block Table Minor Commands</td>
<td>139</td>
</tr>
<tr>
<td>Display Checkpoint ID Information</td>
<td>141</td>
</tr>
<tr>
<td>Locate IMS Modules in Virtual Memory</td>
<td>142</td>
</tr>
<tr>
<td>IMS Startup Parameters and Overrides</td>
<td>143</td>
</tr>
<tr>
<td>IRLM Startup Parameters</td>
<td>145</td>
</tr>
<tr>
<td>IMS Control Region Virtual Storage Analysis</td>
<td>148</td>
</tr>
<tr>
<td>General IMS Information</td>
<td>150</td>
</tr>
<tr>
<td>IMS Online Change Analysis</td>
<td>153</td>
</tr>
<tr>
<td>IMS Dataset Information</td>
<td>155</td>
</tr>
<tr>
<td>IMS Trace Table Entries</td>
<td>160</td>
</tr>
<tr>
<td>Plotting System Information</td>
<td>161</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 6</th>
<th>Device Information</th>
<th>165</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Listing Commands</td>
<td>166</td>
<td></td>
</tr>
<tr>
<td>Disk Information</td>
<td>168</td>
<td></td>
</tr>
<tr>
<td>Long-Term Device Utilization</td>
<td>177</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 7</th>
<th>PSB Group Definition and Activation</th>
<th>183</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defining PSB Groups</td>
<td>184</td>
<td></td>
</tr>
<tr>
<td>Specifying Maximum Number of PSB Groups</td>
<td>187</td>
<td></td>
</tr>
<tr>
<td>Activating New PSB Groups</td>
<td>188</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 8</th>
<th>DBCTL Regions</th>
<th>189</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBCTL Region Immediate Commands</td>
<td>190</td>
<td></td>
</tr>
<tr>
<td>DBCTL Region Major Commands</td>
<td>194</td>
<td></td>
</tr>
<tr>
<td>DBCTL Region Minor Commands</td>
<td>198</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 9</th>
<th>IMS Resource Information</th>
<th>217</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Management Block Commands</td>
<td>218</td>
<td></td>
</tr>
<tr>
<td>PSB Commands</td>
<td>228</td>
<td></td>
</tr>
<tr>
<td>Pool Commands</td>
<td>233</td>
<td></td>
</tr>
<tr>
<td>IMS Logging Analysis</td>
<td>254</td>
<td></td>
</tr>
<tr>
<td>Fast Path Resource Commands</td>
<td>260</td>
<td></td>
</tr>
<tr>
<td>Extended Recovery Facility (XRF) Support</td>
<td>266</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 10</th>
<th>IMS/DB2 Attach Information</th>
<th>269</th>
</tr>
</thead>
<tbody>
<tr>
<td>Major Commands</td>
<td>270</td>
<td></td>
</tr>
<tr>
<td>Minor Commands</td>
<td>271</td>
<td></td>
</tr>
</tbody>
</table>
List of Figures

FIGURE 1. OMEGAMON II user interfaces and data components .................................. 22
FIGURE 2. INFO-line command help ................................................................. 35
FIGURE 3. IMSCTL region virtual storage areas ............................................... 148
FIGURE 4. Information display for IMS without Remote Site Recovery .................... 150
FIGURE 5. Information display for IMS with Remote Site Recovery ......................... 151
FIGURE 6. Typical MSYS system information display ........................................ 151
FIGURE 7. Sample PLOT command results ....................................................... 163
FIGURE 8. The DPLT major command with the DSKQ minor command ................. 173
FIGURE 9. PDSK under IDEV display .............................................................. 180
FIGURE 10. PDSK display of devices allocated to the DLS address space ............... 180
FIGURE 11. Typical analysis of OLDS activity using IDEV .................................. 181
FIGURE 12. Typical analysis of WADS activity using IDEV .................................. 182
FIGURE 13. RGN command used with IDEV .................................................... 194
FIGURE 14. LOCK command display ............................................................... 202
FIGURE 15. Region major command with several minor commands ................. 215
FIGURE 16. Region major command with service unit and page rate minor commands ... 215
FIGURE 17. Typical AIOP major and SIZE minor output .................................. 233
FIGURE 18. DMPL major and SIZE minor commands ...................................... 238
FIGURE 19. Display VSAM buffer subpool statistics ........................................ 239
FIGURE 20. ISAM/OSAM buffer pool statistics ................................................ 242
FIGURE 21. FPDB major and BSTS and SIZE minor commands ....................... 247
FIGURE 22. Typical EPCB major and SIZE minor output .................................. 249
FIGURE 23. IMS PSBW pool information ......................................................... 250
FIGURE 24. DSIZ and SIZE minors of PSPL ................................................... 251
FIGURE 25. LSYS and its minor commands ...................................................... 255
FIGURE 26. (Page 2 of 2) LSYS and its minor commands .................................. 256
FIGURE 27. XRFS command executed on the active IMS system ....................... 266
FIGURE 28. XRFS command executed on the standby IMS system ...................... 266
FIGURE 29. XRSV command output (RDS mode inactive) .................................. 267
FIGURE 30. XRSV command output (LINK surveillance mode not used) .......... 268
FIGURE 31. Typical output for ILST ................................................................. 283
FIGURE 32. XLMS output in character format ................................................. 285
FIGURE 33. XLMS output in hex and character formats .................................... 285
FIGURE 34. Output from XMLS command with an action character ................. 285
FIGURE 35. ICNS command with the ICMD authorized command .................... 303
FIGURE 36. OCMD and CONS output ............................................................... 305
FIGURE 37. AMAP Display Format (XA and ESA) ............................................ 309
FIGURE 38. Example of TCB structure .............................................................. 318
FIGURE 39. Typical JCL for external writer procedure ..................................... 320
FIGURE 40. DEX02 screen space .................................................................... 342
List of Tables

Table 1. OMEGAMON II for DBCTL Documentation Set .................................................. 12
Table 2. Symbols in Command Syntax .............................................................................. 16
Table 3. Profile Definition Commands ........................................................................... 82
Table 4. OUTP Minor Commands with Initial Settings ...................................................... 90
Table 5. Color/Highlighting Settings in .SCC ................................................................. 100
Table 6. Exception Analysis Control Commands ............................................................. 108
Table 7. Commands that require the action character ...................................................... 275
Table 8. Partial list of valid control block identifiers ....................................................... 280
Introduction

This manual contains reference information on the command user interface for the realtime performance component (OMEGAMON®) of OMEGAMON II® for DBCTL, Version 500. The manual describes each command in detail, and groups commands within chapters based on the type of information the commands display. For example, there are chapters on the commands that give information about IMS devices, IMS regions, and IMS internal and external resources. In addition, the opening chapter covers operational essentials.

This manual does not include information about the commands in the bottleneck analysis (DEXAN) or historical (EPILOG) components of OMEGAMON II for DBCTL. For descriptions of these commands, see the OMEGAMON II for DBCTL Bottleneck Analysis (DEXAN) Reference Manual, and OMEGAMON II for DBCTL Historical Component (EPILOG) Reference Manual.

Unless otherwise specified, OMEGAMON II refers to the OMEGAMON II for DBCTL product, and OMEGAMON refers to the command interface to the realtime performance component of OMEGAMON II for DBCTL.
About This Book

Who should use this guide

This manual is for users who are familiar with performance monitoring software and IMS. For introductory, step-by-step instructions on how to use OMEGAMON II’s command interface, as well as the primary CUA™ interface and menu interface, see the OMEGAMON II for DBCTL User’s Guide.

Documentation set information

The documentation listed in the following table is available for the Candle IMS Products. To order additional product manuals, contact your Candle Support Services representative. Where to look for more information

<table>
<thead>
<tr>
<th>Document Number</th>
<th>Document Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC51-6057</td>
<td>Installing Candle Product on MVS</td>
<td>Provides installation instructions and other installation considerations.</td>
</tr>
<tr>
<td>ID53-6341</td>
<td>OMEGAMON II for DBCTL Realtime Commands Reference Manual</td>
<td>Describes in detail all of the features of the OMEGAMON II for DBCTL command interface.</td>
</tr>
<tr>
<td>ID53-6344</td>
<td>OMEGAMON II for DBCTL Bottleneck Analysis (DEXAN) Reference Manual</td>
<td>Provides reference information and descriptions of the features of the bottleneck analysis component.</td>
</tr>
<tr>
<td>ID53-6345</td>
<td>OMEGAMON II for DBCTL Historical Component (EPILOG) Reference Manual</td>
<td>Provides a comprehensive description of the features of the historical component (EPILOG).</td>
</tr>
<tr>
<td>ID53-6346</td>
<td>OMEGAMON II for DBCTL Historical Component (EPILOG) User’s Guide</td>
<td>Teaches you, step-by-step, how to operate the historical component (EPILOG) reporter after installation.</td>
</tr>
<tr>
<td>I251-6317</td>
<td>OMEGAMON II for IMS/DBCTL Configuration and Customization Guide</td>
<td>Explains how to configure and customize OMEGAMON II and its user interfaces and components.</td>
</tr>
</tbody>
</table>
Table 1. OMEGAMON II for DBCTL Documentation Set

<table>
<thead>
<tr>
<th>Document Number</th>
<th>Document Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I299-6303</td>
<td>Application Trace Facility for OMEGAMON II for IMS and DBCTL</td>
<td>Provides user and reference information about the features of the Application Trace Facility (ATF) component.</td>
</tr>
<tr>
<td>I299-6338</td>
<td>Transaction Reporting Facility for OMEGAMON II for IMS and DBCTL</td>
<td>Provides user and reference information about the features of the Transaction Reporting Facility (TRF) component.</td>
</tr>
<tr>
<td>I299-6339</td>
<td>IMS Console Facility for OMEGAMON II for IMS and DBCTL</td>
<td>Provides a comprehensive description of the features of the IMS Console Facility (ICF) component.</td>
</tr>
<tr>
<td>W052-6238</td>
<td>Candle Product Messages Manual Volumes 1, 2 and 3</td>
<td>Provides reference summary information for all Candle product messages.</td>
</tr>
</tbody>
</table>

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 and the other related products.

**Ordering additional product documentation**

To order additional product manuals, contact your Candle Customer Support representative.

**We would like to hear from you**

Candle welcomes your comments and suggestions for changes or additions to the documentation set. A user comment form, located at the back of each manual, provides simple instructions for communicating with the Candle Information Development department.

You can also send email to UserDoc@candle.com. Please include “OMEGAMON II for DBCTL Realtime Commands Reference Manual Version 510” in the subject line.
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.
Documentation Conventions

Introduction
Candle documentation adheres to accepted typographical conventions for command syntax. Conventions specific to Candle documentation are discussed in the following sections.

Panels and figures
The panels and figures in this document are representations. Actual product panels may differ.

Revision bars
Revision bars (||) may appear in the left margin to identify new or updated material.

Variables and literals
In examples of command syntax, uppercase letters are actual values (literals) that the user should type; lowercase letters are used for variables that represent data supplied by the user. Default values are underscored.

LOGON APPLID (cccccccc)
In the above example, you type LOGON APPLID followed by an application identifier (represented by cccccccc) within parentheses.

Note: In ordinary text, variable names appear in italics.
Symbols

The following symbols may appear in command syntax:

<table>
<thead>
<tr>
<th></th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The “or” symbol is used to denote a choice. Either the argument</td>
</tr>
<tr>
<td></td>
<td>on the left or the argument on the right may be used. Example:</td>
</tr>
<tr>
<td></td>
<td>YES</td>
</tr>
<tr>
<td></td>
<td>In this example, YES or NO may be specified.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Denotes optional arguments. Those arguments not enclosed in</td>
</tr>
<tr>
<td></td>
<td>square brackets are required. Example:</td>
</tr>
<tr>
<td></td>
<td>APPLDEST DEST [ALTDEST]</td>
</tr>
<tr>
<td></td>
<td>In this example, DEST is a required argument and ALTDEST is</td>
</tr>
<tr>
<td></td>
<td>optional.</td>
</tr>
<tr>
<td>{ }</td>
<td>Some documents use braces to denote required arguments, or to</td>
</tr>
<tr>
<td></td>
<td>group arguments for clarity. Example:</td>
</tr>
<tr>
<td></td>
<td>COMPARE {workload} -</td>
</tr>
<tr>
<td></td>
<td>REPORT={SUMMARY</td>
</tr>
<tr>
<td></td>
<td>The workload variable is required. The REPORT keyword must be</td>
</tr>
<tr>
<td></td>
<td>specified with a value of SUMMARY or HISTOGRAM.</td>
</tr>
<tr>
<td>_</td>
<td>Default values are underscored. Example:</td>
</tr>
<tr>
<td></td>
<td>COPY infile outfile - [COMPRESS={YES</td>
</tr>
<tr>
<td></td>
<td>In this example, the COMPRESS keyword is optional. If specified,</td>
</tr>
<tr>
<td></td>
<td>the only valid values are YES or NO. If omitted, the default is YES</td>
</tr>
</tbody>
</table>

Table 2. Symbols in Command Syntax
What’s New

Chapter overview

Version 510 of OMEGAMON II for IMS and OMEGAMON II for DBCTL significantly enhanced the Application Trace Facility. This version also provides several new functions, which broaden the overall functionality of OMEGAMON II for IMS and OMEGAMON II for DBCTL.

Chapter contents

Application Trace Facility ................................................. 18
New OMEGAMON II Functions ........................................... 19
Documentation Changes .................................................... 20
Application Trace Facility

Application Trace Facility (ATF) is a monitoring agent in OMEGAMON II for IMS and OMEGAMON II for DBCTL. In Version 510, ATF was significantly enhanced so that:

- Multiple ATF OMEGAMON Classic address space sessions can monitor the same IMS
- The IMS Monitor can run concurrently with these ATF sessions
- All environments for IMS, IMS DB/DC, IMS DC and IMS DBCTL are supported
- A site has external control of its operations
- IMS Version 7 DC Monitor is supported
- Concurrent Online TRF display and ATF display functions are supported

In the previous Version 500, ATF had a DETAIL parameter that could be set to LOW or HIGH. In Version 510, this parameter was removed and the function was separated to display this information on separate sets of panels:

- What used to be DETAIL=LOW in ATF V500 is now the Online TRF Display
- What used to be DETAIL=HIGH in ATF V500 is now new ATF panels

The changes made to ATF in this release are explained in detail in the Application Trace Facility Manual for OMEGAMON II for IMS and DBCTL 510. ATF’s online help has been upgraded to reflect these new features.
New OMEGAMON II Functions

Several new functions were added to OMEGAMON II for IMS and OMEGAMON II for DBCTL. These functions are:

- Expanded generic IMS command support
- Enhanced VSAM buffer pool statistics
- Enhanced fast path buffer pool statistics
- Enhanced fast path statistics
- Enhanced operator assistance for fast path areas
- Additional data and sorting on IMS Message region fields
Online documentation

With version 510, Candle Corporation has moved OMEGAMON II for IMS 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 IMS 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.

The documentation CD being provided with this release has robust and easy-to-use search capabilities. You can search for information in multiple volumes, multiple versions, and across products. The CD also provides easy setup of search indexes with a single click of the mouse.

If you want to order printed copies of the documentation, please contact your Candle Support Services representative.
Chapter overview

OMEGAMON II is a comprehensive software performance management tool for the DBCTL environment. It provides you with realtime and historical performance data and tools to enable you to manage your DBCTL environment.

Chapter contents

OMEGAMON II User Interfaces and Components ................................. 22
Software Requirements .............................................................. 25
Modes of Operation ................................................................. 26
Entering Commands ................................................................. 30
Command Help ....................................................................... 33
Introduction

This manual focuses on the command interface to the realtime performance component (OMEGAMON) of OMEGAMON II. To put this focus in context, the following figure illustrates the OMEGAMON II user interfaces and components:

FIGURE 1. OMEGAMON II user interfaces and data components
User interfaces

OMEGAMON II for DB2 has the following user interfaces:

- **CUA interface**
  OMEGAMON II’s primary user interface is a graphical, object-oriented interface. It follows the guidelines of the IBM® SAA®/CUA (Systems Application Architecture®/Common User Access) model for consistent graphical user interfaces across products. It gives the user access to OMEGAMON II’s key realtime status information, as well as an operator assist feature that simplifies the issuing of IMS commands. From the CUA interface, you can also zoom in to OMEGAMON II’s menu and command interfaces for additional information. The CUA interface is accessed through OMEGAVIEW®. For details on using the CUA interface, refer to the OMEGAMON II for DBCTL User’s Guide.

- **Menu interface**
  For users requiring more detailed information, OMEGAMON II has a menu interface. The menu interface enables the user to access realtime data via an easy-to-use menu system. Each menu option leads to a panel displaying appropriate OMEGAMON II commands and output. For details on using the menu interface, refer to the OMEGAMON II for DBCTL User’s Guide.

- **Command interface**
  The most detailed information is available through the command interface. This interface allows the user to enter extensive commands in any order or combination covering almost every aspect of the DBCTL environment in realtime. Commands can also be saved in screen spaces for easy access. This reference manual provides detailed information on OMEGAMON commands. For information on how to access the command interface and enter commands, refer to the OMEGAMON II for DBCTL User’s Guide.

- **Historical information interfaces**
  For historical information, users request printed reports via a batch report generator, or make interactive queries through a series of TSO or ISPF panels.

Components

OMEGAMON II is composed of the following components:

- **Realtime Performance component (OMEGAMON)**
  The realtime performance component (OMEGAMON) provides basic realtime data about the DBCTL environment. One of its most unique features is exception analysis, which provides automatic alerts when system problems occur or service levels are not being met. This information is available through the CUA interface’s graphical status
displays, or in the form of tables and graphs accessed in OMEGAMON II’s command and menu interfaces. For information on using the realtime performance component, refer to the OMEGAMON II for DBCTL User’s Guide. In addition, this reference manual provides in-depth information on OMEGAMON operation and commands.

- Bottleneck Analysis component (DEXAN)

  The bottleneck analysis component (DEXAN) helps a system tuner perform degradation or bottleneck analysis, which is an analysis technique that focuses on workloads rather than resources. DEXAN breaks down DBCTL thread response time into times spent in various executing states: CPU usage, IMS waits, database I/O waits, and MVS waits. This information is available through the CUA interface’s graphical status displays, or in the form of tables and graphs accessed from OMEGAMON II’s command and menu interfaces. For details on using the bottleneck analysis component, see the OMEGAMON II for DBCTL Bottleneck Analysis (DEXAN) Reference Manual.

- Historical component (EPILOG)

  The historical component (EPILOG) provides historical information about the DBCTL environment. It collects, analyzes, and reports on resource information and bottlenecks for substantial periods of time, such as hours or days. For details on using the historical component, see the OMEGAMON II for DBCTL Historical Component (EPILOG) User’s Guide and the OMEGAMON II for DBCTL Historical Component Reference Manual.
Software Requirements

Requirements

OMEGAMON II is available for IMS and DB Control systems running under MVS, and requires MVS/ESA™ SP5 or above. OMEGAMON II also requires both of the following:

It also requires both of the following:

- IMS 6 or higher
- SMP/E Release 5 or above
Modes of Operation

Introduction

OMEGAMON II runs as a separate MVS started task. It has no dependencies on IMS facilities. This high availability design lets OMEGAMON II diagnose even the most severe problems.

OMEGAMON II’s primary mode of operation is through OMEGAVIEW using VTAM®. This mode gives you access to the primary CUA user interface, and also lets you zoom in to the menu and command interfaces. If you prefer to log on to the command or menu interface directly, you also have the option of using any of the following modes:

- VTAM automatic update mode (uses a VTAM-owned terminal logged on to OMEGAMON II)
- ISPF mode (uses your TSO/ISPF terminal and allows split screen mode)
- TSO mode (uses your TSO terminal from the TSO READY state)
- dedicated mode

For more information, refer to the Configuration and Customization Guide.

VTAM mode

VTAM mode lets you run OMEGAMON II sessions from a VTAM terminal directly, without the intervention of an intermediate online application such as TSO. An optional security logon feature for OMEGAMON II is also provided in VTAM mode. Systems programmers or performance analysts can use VTAM mode to analyze IMS performance in real time when the extra availability of dedicated mode is not required, and when TSO mode would be undesirable. The advantages of VTAM mode over TSO mode include:

- You can use OMEGAMON II on remotely attached display terminals without the overhead and limitations of TSO.
- You can set OMEGAMON II to automatic update mode so that the screen refreshes as in dedicated mode.
- You can use OMEGAMON II without using TSO, which is useful for sites that want to restrict employees from TSO but not from OMEGAMON.
- You can use VTAM automatic update mode from a remote site to process data with the event- and time-driven features (exception logging facility, timed screen facility, and automatic screen facility). TSO mode does not support the event- and time-driven features.

ISPF mode

If you run with TSO and ISPF Version 2, OMEGAMON II has an ISPF split-screen mode that allows you to swap back and forth between OMEGAMON II and another ISPF application.
You can request both basic and extended color options in ISPF mode if your terminal supports them. However, ISPF does not support the following extended highlighting features: blinking, reverse video, and high intensity.

**TSO mode**

In TSO mode, OMEGAMON II simulates logging on to the OMEGAMON II VTAM application ID from your TSO session. Systems programmers or performance analysts can use this mode to access OMEGAMON II if ISPF is not available. In TSO mode, the screen refreshes when you press Enter.

**Dedicated mode**

Dedicated mode offers the highest OMEGAMON II availability. In dedicated mode, OMEGAMON II does not use any telecommunications access. By operating this way, OMEGAMON II can report hardware and software problems so severe that they disable other mechanisms, including MVS system consoles. In dedicated mode, OMEGAMON II refreshes the screen automatically every few seconds without operator intervention. The default cycle time in dedicated mode is 5 seconds; however, you can change this interval (see the .SET command).

You can also establish optional secondary consoles to echo the output of the dedicated terminal. The dedicated terminal and its optional secondary consoles must be local non-SNA devices.
INFO-line Format

Introduction

The top line of an OMEGAMON screen display is called the INFO-line and looks similar to this:

```
/PRINT _ _ _ _ #01  VTM DBCTL IS1C____ + 01/02/97 17:03:37 5 B
```

The INFO-line accepts input (such as the /PRINT command) over the underscores, and displays status information about your session.

The last eight positions of the input line may be reserved by many commands and may not be used to input data.

This section explains the components of the INFO-line using the variables shown in the following figure:

```
<input> _ _ _ _ cccccccc  mode LOG pr cc sysid mm:dd:yy hh:mm:ss nnn Ac
```

- **input area** Accepts INFO-line commands or screen space names. A screen space is a set of commands that is saved under a unique name. Each of the commands on the pre-defined screen is invoked when the screen space name is entered. OMEGAMON distinguishes screen space names from INFO-line commands because INFO-line commands always start with a slash (/). They are explained in the next section.

- **cccccccc** Shows the name of the screen space currently in use, if any.

- **mode** Shows the type of session or mode being used to access OMEGAMON. The possible values are:
  - **DED** A dedicated mode session
  - **DIR** A director segment running dedicated in a cross memory or cross system mode session
  - **DSK** A collector segment running in a cross system mode session
  - **VTD** A director segment running under VTAM in a cross memory or cross system mode session
  - **VTM** A VTAM mode session
  - **VTS** An ISPF mode session running under VTAM
  - **VTT** A TSO mode session running under VTAM
  - **XMM** A collector segment running in a cross memory mode session
LOG If the word LOG appears, the screen is being copied to the REPORT log file. If logging is off, the field is blank.

pr Displays a code or multiple codes for the product(s) running.

cc Identifies the user profile in use.

sysid Indicates the IMS system ID value. This value identifies which IMS system OMEGAMON is monitoring.

mm:dd:yy Indicates the date the screen was last refreshed.

hh:mm:ss Indicates the time the screen was last refreshed.

nnn Indicates the scrolling depth.

A Indicates whether the automatic screen facility is currently active. If it is, an A appears. If it is not, the field is blank.

c The variable c can be one of these values:

S Screen spaces are currently stacked.

B The terminal bell has been activated and will sound if exceptions occur.

Note: If both screen stacking and the bell feature are active, only an S will appear.
Entering Commands

If you want to operate the command interface, you need to know the types of OMEGAMON commands and the appropriate time and place to enter each type.

Command types

The next figure shows the four types of commands used in OMEGAMON along with some sample output.

```
/PRINT_ _ _ _ _ #01 VTM DBCTL I51C_ _ _ + 01/02/97 17:03:37 5 B
DISK  VMXA04 VMXA05 VMSP50 VMHP02 OMONVM DOSTST DP215R DOSRES +
dadr   1A0    1A1   1B0    1B1   2A7   2B0   4F1   4F2
.MIN   DADR   DALT DIO  DIOQ DOPN DRES DSTA DTYP DUSR DVMP +
```

The types are:

**INFO-line**

These commands perform control functions such as printing a screen (/PRINT) and stopping your OMEGAMON session (/STOP). INFO-line commands are executed first and, unlike the other types of commands, disappear as soon as they execute. Therefore, you cannot save them in a screen space. INFO-line commands always begin with a slash (/), and must be entered on the top line starting in column 2. If you are running in an automatic update mode, placing the cursor in column 1 on this line pauses updating until you move the cursor.

**Major**

These commands select general categories for display, such as system information, resource utilization, or storage utilization. In the example, the major command DISK produces a list of online disks. You can enter major commands on any line below the INFO-line.

**Minor**

Minor commands display detailed information about the category that the major selects. In the example, the minor command DADR displays the unit addresses of the devices listed with DISK. You can enter minor commands on any line below the INFO-line, but they will not execute unless they are preceded by the appropriate major command.
Immediate commands serve various functions. Some are system monitoring commands, while others give you information about your session or about OMEGAMON. In the example, .MIN produces a list of all the minors of DISK. Immediate commands can also provide screen and session controls, and often perform the same or similar functions as INFO-line commands of the same name. This enables you to include control commands as part of a screen space. Enter immediate commands on any line below the INFO-line. (It is permissible to enter immediate commands between a major command and one of its minors.)

**Command format**

Here is another example of OMEGAMON commands and their output. Notice that command fields are four characters, but commands can be entered with labels or arguments.

```
.MJ DS DISK DLST DSKB DSKC DSKK DSKM DSKN DSKP DSKQ DSKR DSKS +
DISK VMXA04 VMXA05 VMSP50 VMHP02 OMONVM DOSTST DP21SR DOSRES +
dadr 1A0 1A1 1B0 1B1 2A7 2B0 4F1 4F2 2GDEV SYSDA
MIS003 HSM001 WORKB1 PDB002 MIS002 MIS003 SYSB24 SYSD22 PROD02 +
```

The command entry area is:

**ccccnn**

where:
- The command name is columns 2 through 5.
- The argument field is columns 6 and 7.
- The extended argument or keyword field is columns 8 through 72.

**label field**

Many commands accept a character in this field that alters the type of output displayed. The field is also used for special command modifiers, such as those that request online help text for commands. In the example (since the GDEV command produces output that extends for several lines), the numeral 2 in the label field skips the first two rows of available output.

**command name**

This field contains the command name and is four characters in length. Although the command name field is columns 2–5, OMEGAMON will recognize most commands begun in column 1 and will automatically shift the command one space to the right.
**argument field**  Many commands accept arguments that modify their function or specify output options. In the example, the .MJ command produces a list of major commands. It has been entered with an argument of DS, a group code which specifies that only disk major commands be listed.

**extended argument**  Arguments are entered in columns 8–72. In the example, the GDEV command has been entered with an argument of SYSDA, which is a generic device name in the system. Thus, only devices in the SYSDA category are displayed. Many commands also accept keywords and parameters that can extend to column 72. Note that a parameter string cannot be extended onto a second line. The command (along with keywords) must be re-entered on succeeding lines.
Command Help

Introduction
OMEGAMON provides online helps for every command. This section describes how to request these explanations for different kinds of commands.

Major, minor, and immediate command help
You can use one of three symbols in column 1 in front of a major, minor, or immediate command to request an explanation of the command function. All commands offer a brief, one-line help. Many commands also have an extended help, which gives you more information about the command or displays the command operands. A continuation character (+) to the right of the one-line help indicates that more (extended) help is available.

Help Symbols

? Displays a one-line help that stays on the screen until you clear it.
/
You can clear the help text with the .DCL command discussed later in this guide.
;
Displays an extended help that disappears from the screen on the next cycle.

Examples
Examples of these three symbols used with the .WAT command follow.

This entry for the .WAT immediate command

?.WAT
produces a one-line help as shown below.

When you use a slash instead of a question mark,

/.WAT
an extended help is displayed as shown below.
The entry

;.WAT

produces the same output as /.WAT, except that plus signs display in column 1 below the one-line help. The plus signs are continuation symbols and indicate that the extended help text will disappear on the next cycle.

You can obtain help for a minor command by using the symbols previously described, without knowing the associated major command name. In some cases, however, the same minor name is used for multiple majors. If the function of the minor command is the same for all of its associated majors, OMEGAMON displays a help screen similar to the following one.

In cases where the same minor command name is used for multiple major commands, but the function and help text differs, OMEGAMON lists the possibilities and instructs you to place your cursor under the desired major command name. When you press ENTER, the desired help appears.

If you know that the minor command is shared among several major commands, you can type the desired major command name on the same line after the help request. For example, the entry

/dio DISK

produces the help text for the DIO minor specific to the DISK major.
INFO-line command help

INFO-line command help is obtained with the .ILC immediate command rather than with a symbol in column 1. Enter the name of the INFO-line command (cccccc) in this format (the slash is not needed):

```
.Call cc
```

For example, to get help for the /DEF command, enter:
```
.Call DEF
```

You will see a help like the following.

FIGURE 2. INFO-line command help

```
.Call /DEF
Set definition mode.

Type of Command: INFO-line

Command Format: /DEF cccc

1 | 2 Definition of operands:

/DEF <ccc> ON Turns on definition mode. Definition mode

Off suspends command execution (except for the
type of Command: INFO-line
functions) so that you can define a screen
space without executing commands. Once you
set definition mode with /DEF ON, it remains
in effect until you issue /DEF OFF or the
screen space is saved or replaced.

OFF Restores normal command execution (cancels
/DEF ON or /DEF HOLD).

HOLD Same as ON argument, but definition mode
remains in effect after you save a screen
space. It is only cancelled when you issue
/DEF OFF. Use this option when you want to
```
Chapter overview

This chapter describes some INFO-line and immediate commands that control OMEGAMON, as well as some generic operands. Other operational commands that can be used in customizing your OMEGAMON profile are described in “User Profile Facility” on page 81.

Chapter contents

Command Modifiers and Help Symbols .......................... 38
Operational INFO-line and Immediate Commands ................. 40
Command Modifiers and Help Symbols

Introduction

This section explains command modifiers that can be used generically with all commands of the specified types.

- `[n cccc]`  
  Controls the display of major command output. When a major command selects a list of items (for example, all online disks), the list may extend to more than one line. The value of `n` is a number from 1 to 9 or a letter from A to Z (representing 10 to 35) that specifies the number of lines to skip from the last line displayed. For example, entering 3DISK on the first row of output from the DISK major command displays the 4th row of output.

- `< cccc>`  
  Causes a major command to display a complete list of selected items, even if the list extends to more than one line.

- `[# cccc]`  
  Displays the number of items available for a major command to select.

- `[> cccc]`  
  Indicates a comment line. OMEGAMON generates this character in front of some commands after they execute to prevent them from being executed again on succeeding cycles. It is also displayed in front of help text. You can use it when creating screen spaces to designate the line as comment text. The `_` (underscore) is an alias of `>`.  

- `[? cccc]`  
  Provides a one-line explanation of the command. If a plus sign (+) appears in column 79 of the one-line explanation, it indicates that a more detailed explanation is available by using the slash or semicolon.

- `[/ cccc]`  
  Provides a detailed explanation of the command and its syntax. The help stays on the screen until you clear it.

- `[; cccc]`  
  Provides a detailed explanation of the command and its syntax. The expanded help disappears from the screen on the next cycle.

Rate and difference arguments

For any minor OMEGAMON command that normally displays a numeric value, you can add an argument that calculates the rate of a system event during an OMEGAMON interval or the difference from one interval to the next. Statistics that show rates or differences are often more useful that the original output of the command. For example, the I/O rate is more meaningful to performance analysis than the raw number of I/Os processed.
For commands that normally display time values, such as CPU time, the rate calculated represents a scaled percentage of utilization (expressed as a decimal value) where the real time of the interval equals 100%.

OMEGAMON requires data from 2 cycles to calculate a rate or a difference. On the first cycle (the initialization cycle), a row of 8 periods (........) appears. The rate appears on the next cycle.

Be aware that the rate and difference arguments are not effective in the following cases.

- If a major command selects different items from cycle to cycle. For example, the DSKB major command may select different busy disks at each screen update. When a rate argument is used with a minor of DSKB, the periods may appear after each update, indicating that OMEGAMON is initializing each time.

- If a command displays a value that has been reset to zero by the system between OMEGAMON intervals.

The arguments shown below must be entered in columns 6 and 7 after a minor command that displays a numeric value.

```
```

- **.D** Causes OMEGAMON to calculate the difference between the two most recent values of an event monitored by the specified command (ccccc).
- **.R** Causes OMEGAMON to calculate the rate per second of an event monitored by the specified command (ccccc) during the last OMEGAMON interval.
- **.H** Displays a rate per hour.
- **.M** Displays a rate per minute.
- **.S** Displays a rate per second.
Introduction

This section describes the commands that control OMEGAMON. The commands appear in alphabetical order, starting with special characters.

**.bb**

Clears rest of screen (b indicates a blank).

**Type:** Immediate

This command clears the entire display down to the end of the logical screen.

**====**

Draws a separator line across the screen.

**Type:** Immediate

**Format:** c====aa

- **c** For terminals that support an extended data stream, specifies the color of the separator line. Replace the variable with the first letter of one of the seven extended colors (Red, Blue, White, Green, Pink, Yellow, or Turquoise). For four- or non-color terminals, OMEGAMON translates an entry of G or B to low intensity. All other color codes translate to high intensity.

- **aa** Any two characters you want to use for the separator line. When you invoke this command, the specified characters are repeated across the screen.

For example, this command

**R====++**

creates a line of red pluses across the screen.

**/ABORT**

Stops processing for a cross memory (XMF) collector immediately.

**Type:** INFO-line

If the collector is hung and cannot process a STOP command, you can enter the /ABORT command on the INFO-line of the collector.
This command also frees up any resources the director has associated with it.

**Caution**

Use this command only in an emergency, as it does not bring the collector down cleanly. (If the collector later starts up after an /ABORT, it may crash with an unexpected program check.)

See the discussion of XMF in the *Configuration and Customization Guide.*

<table>
<thead>
<tr>
<th>/ATTACH</th>
<th>Attaches the specified cross system session (alias is /A).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>/ATTACH cccc</td>
</tr>
</tbody>
</table>

The variable `cccc` is the 4-character system ID of the cross system collector. For example, the following command attaches cross system collector A083:

/ATTACH A083

For information about the cross system collector, see the *Configuration and Customization Guide.*

<table>
<thead>
<tr>
<th>/ATTN</th>
<th>Emulates the PA1 (program attention) key.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

/ATTN clears the screen, resets your internal security authorization to the lowest level, returns to default basic color settings, and turns off extended color.

<table>
<thead>
<tr>
<th>/AUPON</th>
<th>Enables automatic update mode.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

This command is valid only in VTAM mode. Automatic update mode is similar to running OMEGAMON in dedicated mode, since the screen updates at regular intervals without pressing ENTER. You can check the current default interval or change it with the .SET command.

While OMEGAMON is automatically updating in VTAM mode, you can continue to enter commands. OMEGAMON delays processing to avoid
executing half-entered input. The length of the delay is determined by the IODELAY keyword of the .SET command.

**Important**

Some network programs do not support automatic update mode (for example, a program that emulates a terminal for your OMEGAMON VTAM mode session).

.AUP is the equivalent immediate command.

<table>
<thead>
<tr>
<th>/AUPOFF</th>
<th>Enables automatic update mode.</th>
</tr>
</thead>
</table>

**Type:** INFO-line

This command will turn the automatic update mode off.

See /AUPON for additional information on automatic update mode.

**Important**

Some network programs do not support automatic update mode (for example, a program that emulates a terminal for your OMEGAMON VTAM mode session).

.AUP is the equivalent immediate command.

<table>
<thead>
<tr>
<th>.AUP</th>
<th>Controls automatic update mode.</th>
</tr>
</thead>
</table>

**Type:** Immediate

See /AUPON, the equivalent INFO-line command, for the command description.

If .AUP is entered without an argument, OMEGAMON displays whether automatic update mode is ON or OFF.

<table>
<thead>
<tr>
<th>.CN</th>
<th>Controls the specified secondary console in dedicated mode.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** .CNxxxx or .CN cc

The variable xxxx is the hexadecimal address of the secondary console.

In dedicated mode, you can set up a secondary OMEGAMON console to be used for output only. The secondary console is a repeater console; it echoes everything that appears on the primary console, but accepts no commands or input of any kind.
After you set the address of the secondary console with .CNxxx, you can manipulate it with the following arguments:

- **.CN OP**: Allocate (open) a secondary console for OMEGAMON output display.
- **.CN CL**: Deallocate (close) a secondary console.
- **.CN SW**: Swap primary and secondary console functions.

The secondary console must be the same terminal type as the primary console.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.D</td>
<td>Deletes the specified number of lines on the physical screen.</td>
</tr>
<tr>
<td>/DCL</td>
<td>Deletes all comment lines on the screen.</td>
</tr>
<tr>
<td>.DCL</td>
<td>Deletes all comment lines below its entry line.</td>
</tr>
<tr>
<td>.DDb</td>
<td>Deletes a block of data.</td>
</tr>
</tbody>
</table>

**Type:** Immediate

**Format:** .Dnn

This command deletes lines beginning with the current line. For example, the following command deletes 5 lines on the physical screen.

```
.D5
```

**Type:** INFO-line

If you want to delete only those comment lines below a certain point on the screen, use the equivalent immediate command instead.

```
.DCL
```

**Type:** Immediate

If you want to delete all comment lines on the screen, use the equivalent INFO-line command instead. Unlike most other immediate commands, .DCL disappears after it executes.

```
.DDb
```

**Type:** Immediate

To delete a block of data from the physical screen, enter .DD on the first line of the block and .DD on the last line.
For example, the following command deletes the line with the first .DD command and the succeeding 3 lines.

```
.DDb
DISK SYSB24 TSO021 SYSB21 MVSA21
DSKB MVSA21 PROD05 SYSA24
.DDb 150 334 D8B
```

This command allows you to define a screen space to include commands that comment themselves out or otherwise change form after execution.

**Type:** INFO-line

**Format:** `/DEF [ON|HOLD|OFF]`

- **ON**  
  `/DEF ON` inhibits automatic updating during a dedicated mode session or a VTAM mode session with automatic updating activated (see the `/AUPON`, `/AUPOFF`, or `.AUP` command). Once you set definition mode with `/DEF ON`, it remains in effect until you issue `/DEF OFF`, or save or replace the screen space.

- **HOLD**  
  Same as ON argument, but definition mode remains in effect after you save a screen space. It is only cancelled when you issue `/DEF OFF`. Use this option when you want to save two or more screens in a row without reactivating definition mode each time.

- **OFF**  
  Restores normal screen updating (cancels the effect of `/DEF ON` or `/DEF HOLD`).

`.DEF` is the equivalent immediate command.

**Type:** Immediate

See `/DEF`, the equivalent INFO-line command, for the command description.

If `.DEF` is entered without an argument, OMEGAMON displays the current definition mode status (ON, OFF, or HOLD).
DELTA does not delete screen spaces from the Candle-supplied screen space library, which is referenced by the RKOIPCSV DD statement.

**Type:** Immediate  
**Format:** DELTc aaaaaaaaa

<table>
<thead>
<tr>
<th>c</th>
<th>One of the following arguments that specifies the location of the screen space. Enter it in column 6:</th>
</tr>
</thead>
<tbody>
<tr>
<td>B or b</td>
<td>Deletes from both main storage and RKOIPCSV (default).</td>
</tr>
<tr>
<td>I</td>
<td>Deletes from main storage (in-storage) only.</td>
</tr>
<tr>
<td>D</td>
<td>Deletes from RKOIPCSV only.</td>
</tr>
</tbody>
</table>

aaaaaaa  
The screen space name you want to delete. Specify the name starting in column 8.

For example, the following command deletes screen space SAMPLE from main storage.

**DELTI SAMPLE**

**DING**  
Forces the terminal bell (audible signal) to sound.

**Type:** Immediate

The bell must be activated with the BELL=YES option of the OPTN immediate command. You can set the BELL=YES option through the menu system and save your setting in a user profile.

**.DIR**  
Executes a cross memory (XMF) or cross system (XSF) director command.

**Type:** Immediate  
**Format:** .DIR ccccccc

.DIR lets you issue commands (cccccc) that control director and collector functions. This capability allows you to execute these commands from a screen space.
The commands are:

- .DIR ABORT (see /ABORT)
- .DIR ATTACH cccc (see /ATTACH)
- .DIR GIVE nn cccc (see /GIVE)
- .DIR TAKE nn cccc (see /TAKE)

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/DOWN</td>
<td>Scrolls down the specified number of lines (alias is /D).</td>
</tr>
</tbody>
</table>

**Type:** INFO-line  
**Format:** /DOWN cccc

cccc can be:

- nnn: Scrolls nnn lines (from 1 to 999).
- BOT: Scrolls to the last logical row.
- CSR: Scrolls according to the current location of the cursor. If the cursor is on the INFO-line, the scroll amount is a page.
- MAX: Scrolls down the number of LROWS defined for your terminal.
- PAGE: Scrolls down so that the current cursor position is at the bottom of the physical screen. This is the default.

/DOWN works only if the number of logical rows is defined to a number greater than the number of physical rows on this terminal. This definition can be changed with the LROWS startup parameter.

If you assign the /DOWN command to a PF key (the default is PF20), you can type any of the optional arguments on the INFO-line before you press the PF key. OMEGAMON will interpret the entry as if you had typed the command plus the argument.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.DSE</td>
<td>Displays the status of stacked screens.</td>
</tr>
</tbody>
</table>

**Type:** Immediate

The .DSE command displays the status of screens stacked with the /STK INFO-line command. The information includes the screen space name, the GETMAINed size in bytes of each screen space, a time stamp that indicates when you stacked the screen, the total amount of storage allocated for all stacked screens, and the relative position of the current stack entry pointer.

The current stack entry pointer is the arrow that is labelled current in the .DSE display. The entry pointer indicates which screen space in the stack has most recently been referenced with a /STK INFO-line command. If you issue
the /STK command with an up or down argument, the pointer moves to the entry above or below the current entry.

Following is a typical .DSE immediate command display.

<table>
<thead>
<tr>
<th>.DSE</th>
<th>Entry</th>
<th>Screen</th>
<th>Size (bytes)</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>1</td>
<td>#01</td>
<td>17987</td>
<td>10:27:14</td>
</tr>
<tr>
<td>+</td>
<td>current --&gt; 2</td>
<td>SYSLOAD</td>
<td>17987</td>
<td>11:08:30</td>
</tr>
<tr>
<td>+</td>
<td>3</td>
<td>OIINITZZ</td>
<td>17987</td>
<td>11:56:00</td>
</tr>
<tr>
<td>+</td>
<td>Total stack size:</td>
<td>53961</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### .EXM

- **Type:** Immediate
- **Format:** `.EXM [nn|c1 c2]`

- **(blank)** Without operands, the .EXM command lists and executes all the minors.
- **nn** Skips nn minor commands and executes the rest.
- **c1 c2** Executes all the minor commands that begin with the specified character string or are in the range specified (c1 to c2). A character string can be 1 to 4 characters long.

This command applies only to the major command that immediately precedes it. The .EXM command executes the minors in alphabetical order and shows the number of minors it has executed. You can use operands to limit the execution to specified minors.

For example, the following .EXM command executes minors of the DISK command that have names starting with A through F.

```
DISK
.EXM A F
```

The .EXM command comments itself out after it executes.

### .EXP

- **Type:** Immediate

The .EXP command displays the expiration date after which OMEGAMON will not function.

Product updates contain new features, support for new IBM releases, enhanced operations, and maintenance. To benefit from improvements, install the product each time it is updated.
The `.FGO` (Fast GO) command is used when creating screen spaces to fetch the next screen space of a series. It allows screen spaces to be chained together and to execute very quickly, bypassing the screen display and the normal OMEGAMON interval. It is particularly useful in exception analysis for implementing the Automatic Screen Facility (ASF) or the Timed Screen Facility (TSF).

The `.FGO` command has a conditional screen fetch feature that fetches a screen space only if a condition is true.

### Keywords for conditional setting of variables

The following keywords are available for conditional setting of variables. Their values are initialized by OMEGAMON.

- **CPSER**
  
  CPU serial number. In the case of a multi-processor, this will compare the supplied CPU serial number with the serial numbers of all CPUs in the complex. If the relational argument is equal (= or EQ), OMEGAMON will fetch the screen space the first time it finds a match. If the relational argument is NE, OMEGAMON will fetch the screen space only after it has checked all of the CPUs in the complex.

- **DIR**
  
  The ID assigned to the director in cross system mode.
Operational INFO-line and Immediate Commands

MODE  The 3-character code for OMEGAMON’s mode of operation. It is displayed on the INFO-line during a session. Refer to “INFO-line Format” on page 28 for a list.

OPSYS  The MVS operating system level

PREFIX  The OMEGAMON product code (for example, OI).

IMSID  The system ID from the SYS= startup parameter.

UNIT  The device number from the UNIT= startup parameter (the primary OMEGAMON console).

USER  The user profile suffix from the USER= startup parameter.

&var  Allows you to set any comparison you want. The keyword and argument can be any variable name set with the .VAR command or any OMEGAMON-defined variable. The .VAR command lists OMEGAMON-defined variables.

Relational operators

The relational operators require blanks on either side except for the equal sign (=).

= or EQ  Keyword equals argument.

GE  Keyword is greater than or equal to argument.

GT  Keyword is greater than argument.

LE  Keyword is less than or equal to argument.

LT  Keyword is less than argument.

NE  Keyword is not equal to argument.

argument  The argument is a 1- to 8-character value to which OMEGAMON compares the keyword. The keyword and argument can be any variable name set with the .VAR command or any OMEGAMON-defined variable. The .VAR command lists OMEGAMON-defined variables.

To protect against the possibility of a looping condition caused by the .FGO command, OMEGAMON limits the number of consecutive fetches allowed. The limit is controlled with the FGOLIMIT keyword of .SET, which is set to 64 by default (in the Candle-supplied profile). After the limit is reached, .FGO acts like .SGO (Screen Go) so that screen spaces continue to execute, but now they display on each cycle.

Therefore, if there was a loop caused by .FGO screen spaces, you will need to correct the condition and re-enable .FGO with the FGOLoop keyword of the .SET command.

Because FGOLoop=ON causes .FGO to display executing screen spaces, you may want to turn it on yourself to test your screen space fetch routines.
If multiple .FGO commands appear on one screen, the last .FGO command without a condition, or for which the condition is true, executes.

**Example:**

To fetch screen space SAMPLE only if the terminal address is 05E1, enter:

```
.FGO SAMPLE UNIT=05E1
```

or

```
.FGO SAMPLE UNIT EQ 05E1
```

See also the .SGO command.

<table>
<thead>
<tr>
<th><strong>/GIVE</strong></th>
<th>Gives the specified number of screen rows to the cross memory (XMF) or cross system (XSF) collector (alias is /G).</th>
</tr>
</thead>
</table>

This command determines the number of lines on the physical screen to be used by a cross memory or cross system collector.

**Type:** INFO-line

**Format:** /GIVE nn cccc

nn The number of screen rows. If you omit nn, all lines are given to collector cccc.

cccc The XMF or XSF collector to receive the lines.

If you issue this command from a collector without specifying cccc, /GIVE returns nn lines to the director’s screen segment.

For example, this next command assigns 15 lines to the screen segment for the collector running on CPU ID A083.

```
/GIVE 15 A083
```

<table>
<thead>
<tr>
<th><strong>/HELP</strong></th>
<th>Describes HELP facilities (alias is /H).</th>
</tr>
</thead>
</table>

**Type:** INFO-line

The help screen space tells you how to find out more about the functions, features, and operation of OMEGAMON.

For help with an individual major, minor, or immediate command, type a question mark (?) in column 1 in front of the command.

For help with an INFO-line command, refer to the .ILC. immediate command.
/HOLD  Controls the execution of OMEGAMON commands.

| Type:     | INFO-line                        |
| Format:   | /HOLD ccc                        |

**ON**  Suspends OMEGAMON command execution.

**OFF**  Returns to normal OMEGAMON command execution.

/HOLD ON allows you to stop automatic updating until you enter /HOLD OFF.

The /HOLD command is designed for users of VM/PASSTHRU. If you are not a user of VM/PASSTHRU, you can also stop automatic updating by placing the cursor in column 1, row 1.

[I]  Inserts nn blank lines.

| Type:     | Immediate                         |
| Format:   | .I nn                             |

This command inserts nn blank lines so that you can insert new commands on the screen. The number of logical rows on your terminal is the maximum.

For example, the next command inserts 5 lines before the current line.

.I 5

.ILC  Displays INFO-line commands or their help text.

| Type:     | Immediate                         |
| Format:   | .ILC /cccccc                      |

The variable /cccccc is an INFO-line command name (slash is optional). To display all of the INFO-line commands and their aliases, enter the .ILC command without a command name.

To display help text for a specific INFO-line command, enter .ILC followed by the command.

For example, the next command generates an explanation of the /STOP INFO-line command.

.ILC /STOP

Operational INFO-line and Immediate Commands

Operational Commands 51
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/LOG</td>
<td>Sends the current OMEGAMON REPORT log or the XLFLOG to the printer.</td>
</tr>
<tr>
<td>.LOG</td>
<td>Sends the current OMEGAMON REPORT log or the XLFLOG to the printer, or manipulates the status of the log.</td>
</tr>
</tbody>
</table>

**Type:** INFO-line

**.LOG**

LOG is the equivalent immediate command, which additionally offers the PUSH and POP arguments. /O is an alias for /LOG with the OUT argument.

**Type:** Immediate

**Format:** .LOGcccc

.LOG accepts the following arguments.

- **ON** Starts logging
- **OFF** Stops logging.
- **OUT** Prints the current log and leaves it open. The command comments itself out to prevent the log from automatically resetting again on the next cycle.
- **PUSH** Saves the status of the log (ON or OFF) so that it can be restored when you execute .LOGPOP. This capability allows you to manipulate the log’s status in screen spaces invoked by .FGO or .SGO, then return it to its original state after these screen routines are complete. One of the following messages will appear on the same line as the command.

  >> Log inactive. Status saved. <<
  or
  >> Log active. Status saved. <<

- **POP** Restores the log to the status in effect when you executed the last .LOGPUSH. One of the following messages will appear on the same line as the command.

  >> Log status restored to inactive. <<
  or
  >> Log status restored to active. <<

**Note:** The Automatic Screen Facility (ASF) and the Timed Screen Facility (TSF) PUSH and POP automatically.

The log is activated and deactivated with the LOG keyword of the OPTN command.

Report characteristics are set with the OUTP major command and its minors. For a full description of the logging facility, see the *OMEGAMON II for IMS User’s Guide*. 
/LOG is the equivalent INFO-line command, however, it does not offer the
PUSH and POP arguments.

<table>
<thead>
<tr>
<th>LSCR</th>
<th>Loads screen space members from the screen space library to main storage.</th>
</tr>
</thead>
</table>

You can make screen spaces more available and more easily fetched by
loading them into main storage with LSCR. For example, if a disk is not
available, you can continue to invoke the screen spaces that you loaded into
main storage with LSCR.

**Type:** Immediate

**Format:** `LSCR  cccccccc  cccccccc  ...  cccccccc`

The variables `ccccccc` are screen space names. Specify screen space member
names starting in column 8. You can load as many members as can fit on the
input line.

For example, the next command asks OMEGAMON to load screen spaces
`ZZ1`, `ZZ2`, and `ZZ3` from the screen space library `CANSOI` to main storage.

```
LSCR  ZZ1 ZZ2 ZZ3
```

**.MIN**

<table>
<thead>
<tr>
<th>.MIN</th>
<th>Lists all minor commands for the preceding major command.</th>
</tr>
</thead>
</table>

This command only applies to the major command that immediately precedes
it. **.MIN** displays the minors in alphabetical order.

**Type:** Immediate

**Format:** `[H].MIN [nn|c1 c2]`

(Blank) Without operands, **.MIN** displays all minor commands of the
major.

**H** The optional label **H** displays one-line help information for
each of the minor commands. The following arguments allow
you to limit the help display and avoid scrolling down to see
the desired help.

- **n** Skips the first `nn` minor commands.
- **c1 c2** Specifies a single character string or a range of minors
from `c1` to `c2` for the help display. A character string
can be 1 to 4 characters long.

The next example displays all minor commands of the DISK major command.

```
DISK
.MIN
```
.MIN comments itself out after execution.

<table>
<thead>
<tr>
<th>.MJ</th>
<th>Lists all major and immediate OMEGAMON commands.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** H.MJ cc

The optional label H displays a one-line help text for each command.  
The variable cc specifies one of the following optional 2-character group names.

<table>
<thead>
<tr>
<th>cc</th>
<th>Command Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>all major and immediate commands</td>
</tr>
<tr>
<td>DB</td>
<td>DMB commands</td>
</tr>
<tr>
<td>DD</td>
<td>dataset commands</td>
</tr>
<tr>
<td>DG</td>
<td>DEXAN for IMS commands (when DEXAN is installed)</td>
</tr>
<tr>
<td>DM</td>
<td>display IMS control block commands</td>
</tr>
<tr>
<td>DS</td>
<td>disk commands</td>
</tr>
<tr>
<td>DV</td>
<td>device commands</td>
</tr>
<tr>
<td>DY</td>
<td>dynamic allocation/deallocation commands</td>
</tr>
<tr>
<td>EX</td>
<td>exception analysis commands</td>
</tr>
<tr>
<td>LG</td>
<td>logging commands</td>
</tr>
<tr>
<td>ME</td>
<td>memory commands</td>
</tr>
<tr>
<td>MI</td>
<td>miscellaneous commands</td>
</tr>
<tr>
<td>PB</td>
<td>PSB commands</td>
</tr>
<tr>
<td>PD</td>
<td>dynamic control block commands</td>
</tr>
<tr>
<td>PL</td>
<td>pool commands</td>
</tr>
<tr>
<td>PS</td>
<td>plotting commands</td>
</tr>
<tr>
<td>RG</td>
<td>region commands</td>
</tr>
<tr>
<td>TC</td>
<td>TCO/TCF commands</td>
</tr>
<tr>
<td>XR</td>
<td>transaction rate commands</td>
</tr>
</tbody>
</table>

For example, the following command lists all device major and immediate commands.

.MJ DV

<table>
<thead>
<tr>
<th>.MJC</th>
<th>Lists all major commands.</th>
</tr>
</thead>
</table>

54  
OMEGAMON II for DBCTL Realtime Commands Reference Manual Version 510
Operational INFO-line and Immediate Commands

**Type:** Immediate

**Format:** H.MJCcc

The optional label H displays a one-line help text for each command. The variable cc is an optional group name. See the .MJ command for a list of these groups.

<table>
<thead>
<tr>
<th>.MJI</th>
<th>Lists all immediate commands.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** H.MJIcc

The optional label H displays a one-line help text for each command. The variable cc is an optional group name. See the .MJ command for a list of these groups.

<table>
<thead>
<tr>
<th>.MMA</th>
<th>Lists all major commands for a minor command.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** .MMA cccc

The variable cccc is a minor command.

<table>
<thead>
<tr>
<th>.MOD</th>
<th>Shows OMEGAMON module names and addresses.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** .MODc

The optional suffix A (.MODA) lists the module names in alphabetical order. This command provides debugging information, including module names and start addresses. If OMEGAMON detects a program check, these names and addresses are useful to Candle Customer Support.

<table>
<thead>
<tr>
<th>.NXE</th>
<th>Controls display of exceptions.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** .NXEcc

- **ON** Displays exceptions. This is the default.
- **OFF** Does not display exceptions.
Operational INFO-line and Immediate Commands

Enter the .NXE immediate command without an argument to show the current status of the exceptions display.

The frequency for executing an exception is controlled by the EXNCYC (EXecute Next CYCle) keyword of the XACB command. When you set the EXNCYC parameter to check the exception less often than every OMEGAMON cycle, the .NXE command controls whether the .NXE command continues to display tripped exceptions on the cycles when they are not due for execution.

<table>
<thead>
<tr>
<th>/O</th>
<th>Prints the existing OMEGAMON REPORT log without closing the log.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

The /O command is an alias for /LOG with the OUT argument. See the description of /LOG or .LOG for complete information about this command.

<table>
<thead>
<tr>
<th>.PCS</th>
<th>Displays OMEGAMON program check statistics.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
</tbody>
</table>

If OMEGAMON detects a program check, the information is useful to Candle Customer Support for debugging.

<table>
<thead>
<tr>
<th>.PFK</th>
<th>Displays or resets command mode PF key definitions for the current session.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
<tr>
<td>Format:</td>
<td>c.PFKnn=aaaaaaaa/*bbb...b</td>
</tr>
<tr>
<td>(blank)</td>
<td>Without operands, .PFK displays all current PF key settings for command mode. PF keys without assignments do not appear on the screen. The default PF key definitions in command mode differ from those in the menu system.</td>
</tr>
<tr>
<td>c</td>
<td>Label E for redefining several PF keys at one time.</td>
</tr>
<tr>
<td>nn</td>
<td>PF key number.</td>
</tr>
<tr>
<td>aaaaaaaaa</td>
<td>Screen space name (1 to 8 characters) or INFO-line command (/cccccc).</td>
</tr>
</tbody>
</table>
| /*bbb... | Comment of up to 32 characters (bbb...) following the slash and asterisk (/*).

Define PF keys

You can define up to 99 physical and logical PF keys. Enter the .PFK command and type the new definition after an equal sign. Type comment text following a slash and asterisk (/*).
For example, the following command sets PF15 to issue the /STOP INFO-line command for this session.

```
.PFK15=/STOP /* Stops OMEGAMON
```

### Assign screen space names to PF keys

Use the same format to assign screen space names to PF keys. For example, the following command sets PF26 to call the screen space DISKS for this session.

```
.PFK26=DISKS /* DASD information
```

To call a screen space assigned to a PF key, press the associated PF key or type its number on the INFO-line.

### Delete definitions

To delete a definition, enter a single underscore (_) for the definition. For example, the following command deletes the definition for PF18:

```
.PFK18=_
```

### Redefine multiple PF keys at once

Use the following steps to redefine several PF keys at one time without having to retype the .PFK command for each one:

1. Enter `E.PFK`.
   
   OMEGAMON gives you an extended display of all current PF key assignments and inserts .PFK before each key number as shown here:
   
   ```
   +.PFK11=/ZOOM /* ZOOMING FEATURE
   ```
   
2. For each new assignment, blank out the plus sign (+) in front of .PFK and type the new assignment following the equal sign.

3. Press Enter.

The assignments you have entered will be in effect for the duration of the session.

---

**Important**

At startup, OMEGAMON executes the Candle screen spaces containing default PF key assignments. The PF keys that you define with .PFK are in effect only for that OMEGAMON session. To make these assignments permanent, you must change the defaults in the screen spaces. For command mode, these screen spaces are KOI@PDEF (PF keys 1-12) and KOI@PDF2 (PF keys 13-24). For menu mode, they are KOI@PNEW (PF keys 1-12) and KOI@P2 (PF Keys 13-24).

---

<table>
<thead>
<tr>
<th><code>/PRINT</code></th>
<th>Prints the current logical screen (alias is /P).</th>
</tr>
</thead>
</table>

**Type:** INFO-line

When the screen prints, a `>LOGGED<` message appears on the INFO-line.
**Note:** The page limit set with the .SET command does not affect the /PRINT command.

<table>
<thead>
<tr>
<th>.PRM</th>
<th>Displays current values of the OMEGAMON startup command parameters.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong> Immediate</td>
<td></td>
</tr>
</tbody>
</table>

This command displays applicable OMEGAMON startup parameters in the following order.

- **IOMODE=cc**: Current OMEGAMON I/O mode. This is the 2-character code entered as the mode in the startup parameters.
- **SYS=cccc**: Current OMEGAMON system ID. This is the same system ID that appears on the INFO-line.
- **DIR=cccc**: Director system ID. This ID only appears when the current OMEGAMON is in collector mode.
- **USER=cc**: User profile suffix. This is the same suffix that appears on the INFO-line.
- **ROWS=nn**: Number of rows on the physical terminal.
- **LROWS=nnn**: Number of logical rows for the output area.
- **COLS=nnn**: Number of columns on the physical terminal.
- **UNIT=cuu**: Terminal address of a dedicated OMEGAMON session.

<table>
<thead>
<tr>
<th>.PRT</th>
<th>Prints the specified portion of the screen to the REPORT file.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong> Immediate</td>
<td></td>
</tr>
<tr>
<td><strong>Format:</strong> .PRTc</td>
<td></td>
</tr>
</tbody>
</table>

Without an argument, .PRT prints a screen image from the INFO-line to the line that contains the .PRT command. After the partial screen prints, .PRT changes to a comment. The optional argument H (.PRTH) prevents the .PRT command from commenting itself out so that it logs these lines continually.

<table>
<thead>
<tr>
<th>/PWD</th>
<th>Specifies an OMEGAMON password or reauthorizes a session.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong> INFO-line</td>
<td></td>
</tr>
</tbody>
</table>

Authorized commands require entry of a password for execution. You can use the /PWD command in the following ways.

1. To authorize your session for internal security, enter the /PWD command on the INFO-line.

   /PWD __________
The system prompts you for a password.

________________ Enter Password

The password does not appear on the screen as you type it. It remains in effect until you reset it.

2. To reset the security level to 0 after your authorized session, type /PWD on the INFO-line, but instead of entering a password, press Enter. Authorization will be cleared.

3. The /PWD command can be entered with your user ID to perform the following functions.
   - Log onto an existing OMEGAMON session and reauthorize external security to your level for the session.
   - In dedicated mode, gain access to external security.
   
   When you use the /PWD command with your user ID to log onto an existing session, you cannot change or update your password.

   For details about OMEGAMON’s security features, see the *Configuration and Customization Guide*.

<table>
<thead>
<tr>
<th>.R</th>
<th>Repeats the last major command with all following minors <em>nnn</em> times.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** `.Rnnn`

The `.R` command repeats all lines from the last major command *nnn* times, as if you had repeatedly entered the major followed by its minors.

The maximum value of *nnn* is the number of LROWS on your terminal.

The `.R` command is similar to `.RC`, which repeats the last major command with all following minor commands as many times as necessary to display all items selected by the major. However, `.Rnnn` specifies exactly how many times you want the major command repeated. The `.R` command is most useful as an editing command when you are creating screen spaces.

The `.R` command is similar to the other OMEGAMON line commands `.D`, `.I)` in that if you enter it on top of a line of data, that line is shifted down. Therefore, you do not have to insert a blank line in order to use this command.

<table>
<thead>
<tr>
<th>.RC</th>
<th>Repeats the last major command and all following minors until all available output is displayed.</th>
</tr>
</thead>
</table>

**Type:** Immediate
When you enter a major command that selects a large number of items, such as disks, only one line of output displays. The .RC command automatically repeats a major command and any minor command until all available lines of output are displayed.

Enter .RC after the major and any desired minors as shown below:

```
   DISK
dadr
dsta
  .RC
```

When you press Enter, all online disks are displayed (DISK) along with their unit addresses (DADR) and their mount status (DSTA). The display looks similar to the following:

```
<table>
<thead>
<tr>
<th>DISK</th>
<th>VMXA14</th>
<th>VMXA12</th>
<th>OMON28</th>
<th>OMON29</th>
<th>VMXA10</th>
<th>SYSB21</th>
<th>VMXA09</th>
<th>OMON2+</th>
</tr>
</thead>
<tbody>
<tr>
<td>dadr</td>
<td>140</td>
<td>141</td>
<td>142</td>
<td>143</td>
<td>144</td>
<td>145</td>
<td>146</td>
<td>147</td>
</tr>
<tr>
<td>dsta</td>
<td>Private</td>
<td>Private</td>
<td>Storage</td>
<td>Storage</td>
<td>Private</td>
<td>Private</td>
<td>Private</td>
<td>Private</td>
</tr>
<tr>
<td>DISK</td>
<td>PPSMPB</td>
<td>TS0026</td>
<td>OMON01</td>
<td>IMS1000</td>
<td>MTLIB3</td>
<td>DB2002</td>
<td>PPSMP3</td>
<td>TS0022+</td>
</tr>
<tr>
<td>dadr</td>
<td>148</td>
<td>149</td>
<td>14A</td>
<td>14B</td>
<td>14C</td>
<td>14D</td>
<td>14E</td>
<td>14F</td>
</tr>
<tr>
<td>dsta</td>
<td>Private</td>
<td>Storage</td>
<td>Private</td>
<td>Private</td>
<td>Private</td>
<td>Private</td>
<td>Private</td>
<td>Storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DISK</td>
<td>COM001</td>
<td>COM002</td>
<td>COM003</td>
<td>MP310A</td>
<td>PRO010</td>
<td>PRO016</td>
<td>PRO011</td>
<td></td>
</tr>
<tr>
<td>dadr</td>
<td>D8A</td>
<td>D8A</td>
<td>D8B</td>
<td>D8C</td>
<td>D8D</td>
<td>D8E</td>
<td>D8F</td>
<td></td>
</tr>
<tr>
<td>dsta</td>
<td>Private</td>
<td>Storage</td>
<td>Private</td>
<td>Storage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

RENM renames screen spaces in main storage or in the user-defined screen space library pointed to by ddname RKOIPCSV. It does not rename Candle-supplied screen spaces in the library pointed to by ddname CANSOI.

**Type:** Immediate

**Format:** `RENMc oldname newname`

- **c** Enter one of the following arguments in column 6 to specify the location of the screen space.
  - **B** Rename in both main storage and RKOIPCSV This is the default.
  - **D** Rename in RKOIPCSV only.
  - **I** Rename in main storage (in-storage) only.

**/REP** Replaces the existing saved screen space of the same name.
Use /REP in place of /SAVE if you want to replace an existing saved screen space in the user-defined screen space library pointed to by ddname RKOIPCSV.

**Type:** INFO-line  
**Format:** /REP cccccccc,a  

- **cccccccc** Specifies the screen space name (1 to 8 characters)  
- **a** One of three arguments that may follow the screen space name. The argument is separated from the screen space name with a comma (.).
  
  - **B** Replaces in both main storage and RKOIPCSV  
  - **D** Replaces in RKOIPCSV only.  
  - **I** Replaces in main storage (in-storage) only.

The following example replaces the current screen space SAMPLE with the newly-defined screen space in both main storage and RKOIPCSV.

/REP SAMPLE,B

**.REP** Displays and sets printer characteristics for the REPORT file.

**Type:** Immediate

When you enter .REP, a series of keywords appears to let you set printer characteristics. If you are changing more parameters than will fit on one line, use the OUTP command instead. The keywords for .REP (and .XLG) duplicate the minors of the OUTP major.

When you change any of the parameters (except FOLD) and press Enter, OMEGAMON automatically spins off the REPORT file and reallocates a new one.

- **SYSOUT=** SYSOUT class.  
- **HOLD=** Specifies whether output is to be placed in the hold queue.  
- **COPIES=** Specifies the number of copies to print.  
- **FORMS=** Specifies the form on which to print.  
- **DEST=** Destination, user ID, or both (separated by a period, colon, or slash) to receive report.  
- **FOLD=** Folds lowercase characters to uppercase.
The following keywords will also appear if their default values have been modified previously. If not, you can type them in and define a new value.

**ID1=** Requests OMEGAMON session-produced separator pages and page headers. The argument for ID1 can be:

- * OMEGAMON generates separator pages and page headers with the appropriate job name printed in block letters on the pages. This is the default.
- *NONE* OMEGAMON does not generate separator pages and page headers.
- `cccccccc` OMEGAMON generates separator pages and page headers with `cccccccc` printed in block letters on the pages. `cccccccc` is up to 8 user-defined characters.

**ID2=** Defines up to 16 characters on the left of the separator page.

**ID3=** Defines up to 16 characters in the center of the separator page.

**ID4=** Defines up to 16 characters on the right of the separator page.

**DDNAME=** Overrides standard OMEGAMON ddnames.

If you change any parameter other than HOLD=, OMEGAMON spins off the XLFLOG file and creates a new one.

If SYSOUT is active, then ddname is inactive and vice versa. The following table shows the parameters in effect and the default settings for SYSOUT or ddname.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default Values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SYSOUT</strong></td>
<td><strong>DDNAME</strong></td>
</tr>
<tr>
<td>SYSOUT=</td>
<td>A (inactive)</td>
</tr>
<tr>
<td>HOLD=</td>
<td>NO (inactive)</td>
</tr>
<tr>
<td>COPIES=</td>
<td>1 (inactive)</td>
</tr>
<tr>
<td>FORMS=</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DEST=</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DDNAME=</td>
<td>(inactive) OIREPORT</td>
</tr>
<tr>
<td>FOLD=</td>
<td>YES YES</td>
</tr>
<tr>
<td>ID1=</td>
<td>jobname jobname</td>
</tr>
<tr>
<td>ID2=</td>
<td>(blank) (blank)</td>
</tr>
<tr>
<td>ID3=</td>
<td>(blank) (blank)</td>
</tr>
<tr>
<td>ID4=</td>
<td>(blank) (blank)</td>
</tr>
</tbody>
</table>
### /RESHOW
Reshow the previously saved version of the current screen space.

<table>
<thead>
<tr>
<th>Type:</th>
<th>INFO-line</th>
</tr>
</thead>
</table>

The /RESHOW command, set to a PF key, gives you the convenience of refreshing your original screen space with a single keystroke after you have made temporary alterations.

### .RMF
Displays RMF information.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
</table>

This command displays the RMF version and version code, the current interval length, and the cycle time.

### .RMFS
Changes the RMF level set by OMEGAMON.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
</table>

**Format:** `RMFS nnn`

`nnn` specifies the RMF level for your system. For example, enter **RMFS 351** to monitor a system that is running RMF level 3.5.1.

OMEGAMON now sets the RMF level dynamically when you initialize it. However, OMEGAMON cannot correctly set the RMF level if RMF is not running, or if OMEGAMON is not APF authorized. OMEGAMON therefore selects a default RMF setting. If this setting is not correct, you can change it with RMFS.

RMFS cannot change the RMF level setting when OMEGAMON has dynamically determined the correct level. Use the SYS major command with the ENV minor to display the level of RMF that your system is running.

### .RTN
Terminates an ASF or TSF sequence and returns to the calling screen space.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
</table>

**Format:** `n.RTNcc aaaaaaaaa`

The .RTN command is required at the bottom of the last screen space in an Automatic Screen Facility (ASF) or Timed Screen Facility (TSF) sequence. It returns to the calling screen space and re-enables exception analysis for further automatic calls. See ASF and TSF described in "Automating and Logging Features" on page 331.
n  The optional label \( n \) specifies the number of cycles to delay the return to the calling screen space. The value of \( n \) can be the numbers 1 to 9 or the letters A to Z (representing 10 to 35). Each time the screen updates, \( n \) decrements by 1. When \( n=0 \), the current screen executes and OMEGAMON fetches the next screen space.

For example, the following command returns to the calling screen space after 7 cycles.

6.RTN

cc  The variable \( cc \) is the NR (no reset) argument. It prevents the .RTN command from automatically resetting the automatic update interval and the log status.

aaaaaaa  This optional argument specifies a screen space for ASF or TSF to return to other than the calling screen space.

The .RTN command automatically resets the automatic update interval and the log status to that in effect when the ASF or TSF sequence began.

/SAVE  Saves the specified new screen space (alias is /S).

Type: INFO-line

Format: /SAVE cccccccc,a

cccccccc  Specifies the screen space name (1 to 8 characters). The screen space name must be a unique alphanumeric name. It must begin with an alpha character and can contain national characters ($, *, or &). If the name you want to assign already exists, use /REP instead of /SAVE.

a  Specifies one of three arguments that may follow the screen space name. The argument is separated from the screen space with a comma (,).

B  Saves the screen space to both RKOIPCSV and main storage (in-storage screen facility).

D or b  Saves the screen space to RKOIPCSV only (default).

I  Saves the screen space to main storage only.

Note: Since screen spaces are saved to the library pointed to by \texttt{ddname RKOIPCSV}, the /SAVE command works only if RKOIPCSV was created and properly concatenated at installation. If you are unable to save your screen space, or if your screen space was successfully saved but OMEGAMON is not accessing it, check with your installer or check the Configuration and Customization Guide for details about RKOIPCSV.
This next example saves the current screen space SAMPLE in both main storage and RKOIPCSV.

**/SAVE SAMPLE,B**

For guidelines on creating screen spaces, see the *OMEGAMON II for IMS User’s Guide*.

<table>
<thead>
<tr>
<th>SCRN</th>
<th>Lists screen space member names.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** SCRNC aa bb

- **c** Enter one of the following arguments in column 6 to specify the source of the member list.
  - **B or b** Lists all screen spaces in both the screen space libraries and main storage (default).
  - **D** Lists all screen spaces in the screen space libraries only.
  - **I** Lists all screen spaces in main storage (in-storage) only.

- **aa bb** Lists all screen spaces beginning with characters aa to bb (start aa in column 8).
- **a** Lists all screen spaces beginning with character(s) a (start a in column 8).

The following example lists all screens that have names starting with C or D.

**SCRN C D**

The next example lists all screen spaces in main storage from PA to PA999999.

**SCRNI PA**

or

**SCRNI PA, PA**

**.SGO** Fetches the specified screen space on the next cycle.

**Type:** Immediate

Use the .SGO (Screen GO) command when creating screen spaces to build a series of screen spaces that will execute in sequence. .SGO causes screen spaces to branch to other screen spaces. It is particularly useful for implementing the Automatic Screen Facility (ASF) or the Timed Screen Facility (TSF) features of exception analysis.
.SGO has a conditional screen fetch feature that fetches a screen space only if a condition is true.

**Format:**

```
.n.SGO cccccccc

[CPSER  {=|EQ|GE|GT|LE|LT|NE} argument]
[DIR    {=|EQ|GE|GT|LE|LT|NE} argument]
[MODE   {=|EQ|GE|GT|LE|LT|NE} argument]
[OPSYS  {=|EQ|GE|GT|LE|LT|NE} argument]
[PREFIX {=|EQ|GE|GT|LE|LT|NE} argument]
[IMSID  {=|EQ|GE|GT|LE|LT|NE} argument]
[UNIT   {=|EQ|GE|GT|LE|LT|NE} argument]
[USER   {=|EQ|GE|GT|LE|LT|NE} argument]
```

**n** The variable `n` is an optional numeric label that allows you to delay the fetch of screen space `ccccccc` for a number of cycles up to 35. Use the numbers 1 to 9 or the letters A to Z (representing 10 to 35 cycles). Each time the screen updates, `n` decrements by 1. When `n=0`, screen `ccccccc` is fetched on the next cycle.

**ccccccc** Specifies the screen space name.

### Keywords for conditional setting of variables

The following keywords are available for conditional setting of variables. Their values are initialized by OMEGAMON.

- **CPSER** CPU serial number. In the case of an MP, this keyword will compare the supplied CPU serial number with the serial numbers of all CPUs in the complex. If the relational argument is equal (= or EQ), OMEGAMON will fetch the screen space the first time it finds a match. If the relational argument is NE, OMEGAMON will fetch the screen space only after it has checked all of the CPUs in the complex.

- **DIR** The ID assigned to the director in cross system mode.

- **MODE** The 3-character code for OMEGAMON’s mode of operation. It is displayed on the INFO-line during a session. Refer to “INFO-line Format” on page 28 for a list.

- **OPSYS** The MVS operating system level.

- **PREFIX** The OMEGAMON product code (for example, OI).

- **UNIT** The device number from the UNIT= startup parameter (the primary OMEGAMON console).

- **USER** The user profile suffix from the USER= startup parameter.

- **&var** Allows you to set any comparison you wish. The keyword and argument can be any variable name set with the .VAR command or any OMEGAMON-defined variable. The .VAR command lists OMEGAMON-defined variables.
Relational operators

The relational operators require blanks on either side except for the equal sign (=).

- **= or EQ** Keyword equals argument.
- **GE** Keyword is greater than or equal to argument.
- **GT** Keyword is greater than argument.
- **LE** Keyword is less than or equal to argument.
- **LT** Keyword is less than argument.
- **NE** Keyword is not equal to argument.

**argument**

The argument is a 1- to 8-character value to which OMEGAMON compares the keyword. The keyword and argument can be any variable name set with the .VAR command or any OMEGAMON-defined variable. The .VAR command lists OMEGAMON-defined variables.

For example,

- **.SGO DISK**
  
  fetches screen space DISK on the next cycle.

- **B.SGO DISK**
  
  delays the fetch of screen space DISK 11 cycles, and fetches it on the next cycle after \( n = 0 \).

To fetch screen space SAMPLE only if you are running in an XA environment, enter

- **.SGO SAMPLE OPSYS=210**
  
  or

- **.SGO SAMPLE OPSYS EQ 210**

If multiple .SGO commands appear on one screen, the last .SGO command without a condition, or for which the condition is true, executes.

You may also use the .FGO command to fetch screen spaces. It functions the same as .SGO except that .FGO bypasses the screen display and the OMEGAMON cycle wait.

<table>
<thead>
<tr>
<th>.SPT</th>
<th>Assigns a number to a pattern value to accommodate generic selection of certain major command output.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `.SPT/n cccccccc`
The following example sets pattern 2 to all names beginning with OP.

```
.SPT/2 OP*
```

The entry

```
DEVP/2
```

invokes pattern 2 for the major command DEVP.

<table>
<thead>
<tr>
<th>.STK</th>
<th>Saves and stacks the current screen output for later recall.</th>
</tr>
</thead>
</table>

OMEGAMON currently refreshes the screen every cycle. There are times, however, when you may want to save a screen’s output so you can return to it later. You may want to investigate a problem by going to another screen or by issuing commands, and then returning to the original screen. You can do this with the .STK command.

**Type:** INFO-line  
**Format:** .STK ccccc

where ccccc is one of the following arguments for recalling and deleting entries from the stack:

- **(blank)** Without an argument, .STK saves all data on the current screen (including all LROWS), not just the display window visible on the terminal. The maximum number of screens you can stack is 999.
- **n** Recalls stacked entry n.
- **U (up)** Recalls the entry prior to the current one. In the .DSE display, the Up argument moves the pointer to the entry above the current pointer.
- **D (down)** Recalls the entry immediately following the current entry. In the .DSE display, the Down argument moves the pointer to the entry below the current pointer.
- **EMPTY** Clears the entire contents of the stack.
DEL n  Deletes entry n.
R     Recalls the current entry.

The recall functions n, U, D, and R do not delete screens from the stack.

The .DSE immediate command displays the status of stacked screens and the
amount of storage being consumed to stack them. A current entry pointer
indicates the most recently referenced screen in the stack.

OMEGAMON displays a message on the INFO-line when it recalls a screen
from the stack to indicate that this is not a currently executed screen. If
OMEGAMON recalls a stacked screen while in auto-update mode, it places
the recalled screen in HOLD mode until you press ENTER.

The screen stacking feature works with extended color if extended color was
on when the screen space was saved and if extended color is on when
OMEGAMON recalls the screen. (See the .SCC command in the Profile Menu
to activate extended color.)

The following /STK command saves the current screen, M110, onto the stack.

/STK___________ M110  VTM DBCTL V500/C  I51C____ + 01/02/97 9:14:21   S

When there are stacked screens, the INFO-line of any non-stacked screen
displays an S on the far right of the screen. (When the bell is on and a B is
displayed, the S overlays the B.)

The next /STK command recalls entry number 2 (screen space DISKS) from
the stack onto the screen.

/STK 2_________ #03  VTM DBCTL V500/C  I51C____ + 01/02/97 9:14:21   S

The next figure shows a typical result of /STK 2.

<table>
<thead>
<tr>
<th>DISK</th>
<th>SYSB24</th>
<th>WORKB2</th>
<th>MVSA21</th>
<th>OMON11</th>
</tr>
</thead>
<tbody>
<tr>
<td>dadr</td>
<td>140</td>
<td>142</td>
<td>143</td>
<td>146</td>
</tr>
<tr>
<td>dalc</td>
<td>18</td>
<td>80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dio</td>
<td>311</td>
<td>138</td>
<td>39176</td>
<td>37902</td>
</tr>
<tr>
<td>data</td>
<td>PRIVATE</td>
<td>STORAGE</td>
<td>PRIVATE</td>
<td>STORAGE</td>
</tr>
<tr>
<td>dtyp</td>
<td>3380</td>
<td>3380</td>
<td>3380</td>
<td>3380</td>
</tr>
</tbody>
</table>
### Operational INFO-line and Immediate Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/STOP</code></td>
<td>Stops OMEGAMON.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>STOP</code></td>
<td>Stops OMEGAMON.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Immediate</td>
</tr>
</tbody>
</table>

STOP can be entered either on the INFO-line or in the main body of the screen to stop OMEGAMON.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/TAKE</code></td>
<td>Takes the specified number of screen rows from a cross memory (XMF) or cross system (XSF) collector (alias is <code>/T</code>).</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

| Format: | `/TAKE nn cccc` |

- **nn**: The number of lines that the cross system/cross memory segment takes from segment `cccc`.

If you issue this command from a collector and do not specify `cccc`, it takes `nn` lines from the director’s screen segment. If `nn` is omitted, OMEGAMON takes all lines but one from the specified segment.

For example, the following command takes 15 lines from the screen segment for collector A083:

```
/TAKE 15 A083
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/TOP</code></td>
<td>Scrolls to the top of the logical screen.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/UP</code></td>
<td>Scrolls up the specified number of lines (alias is <code>/U</code>).</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>INFO-line</td>
</tr>
</tbody>
</table>

| Format: | `/UP cccc` |

- **nnn**: Scrolls `nnn` lines (from 1 to 999).
- **CSR**: Scrolls according to the current location of the cursor. If the cursor is on the INFO-line, the scroll amount is a page.
- **MAX**: Scrolls to the top of the screen.
Operational Commands 71

Operational INFO-line and Immediate Commands

The following example scrolls up 20 lines.

/UP 20

If you have assigned the /UP command to a PF key (the default is PF19), you
can type any of the optional arguments on the INFO-line before you press the
PF key, and OMEGAMON interprets the entry as if you had typed the
command plus the arguments.

/UP works only if the number of logical rows (LROWS) is defined to a number
greater than the number of physical rows on the terminal. This definition can
be changed with the LROWS startup parameter.

<table>
<thead>
<tr>
<th>.VAR</th>
<th>Sets, displays, or deletes variables.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
<tr>
<td>Format:</td>
<td></td>
</tr>
</tbody>
</table>

[C].VAR

{SET &variable value (comment)}
{LIST &variable}
{DEL &variable}
{CPSER {= EQ GE GT LE LT NE} argument}
{DIR {= EQ GE GT LE LT NE} argument}
{MODE {= EQ GE GT LE LT NE} argument}
{OPSYS {= EQ GE GT LE LT NE} argument}
{PREFIX {= EQ GE GT LE LT NE} argument}
{UNIT {= EQ GE GT LE LT NE} argument}
{USER {= EQ GE GT LE LT NE} argument}
{&var {= EQ GE GT LE LT NE} argument}

C Requests conditional processing. Allows you to specify a value for
given variable when the criteria is matched.

SET or S Sets or changes the value of a variable.

&variable The variable must be 1 to 8 characters. The
ampersand (&) preceding the variable name is
optional.

value 1- to 64-character alphanumeric string assigned
to &variable. Single quotes are required only if
special characters or blanks are used in the string.

comment 1- to 35-character alphanumeric annotation that
follows the variable string.
Keywords for conditional setting of variables

The following keywords are available for conditional setting of variables. Their values are initialized by OMEGAMON.

**CPSER**
CPU serial number. In the case of a multi-processor environment, the supplied CPU serial number is compared with the serial numbers of all CPUs in the complex. If the relational argument is equal (= or EQ), OMEGAMON will set the variable the first time it finds a match. If the relational argument is not equal (NE), OMEGAMON will set the variable only after it has checked all of the CPUs in the complex.

**DIR**
The ID assigned to the director in cross system mode.

**MODE**
The 3-character code for OMEGAMON’s mode of operation. It is displayed on the INFO-line during a session. Refer to “INFO-line Format” on page 28 for a list.

**PREFIX**
The OMEGAMON product code (for example, OI).

**UNIT**
The device number from the UNIT= startup parameter (the primary OMEGAMON console).

**USER**
The user profile suffix from the USER= startup parameter.

**&var**
Allows you to set any comparison you want. The keyword and argument can be any variable name set with the .VAR command or any OMEGAMON-defined variable.

**OMEGAMON-defined variables**

OMEGAMON defines the following variables internally for your use:

- &ZFRSTSS - First screen space name.
- /ZOOM INFO-line command variables:
  - &ZOOM data found at the cursor location
  - &ZOOMC command or exception name field
  - &ZOOMS originating screen space
- Exception variables:
  - &ZXccccT threshold value
  - &ZXccccV last trip value
  - &ZXccccW worst trip value

The variable cccc is the exception name.
Note: The letter Z is reserved for Candle Corporation use as the first alpha character of a variable.

Relational operators
The relational operators require blanks on either side except for the equal sign (=).

= Keyword equals argument. The equal sign can be entered without blanks on either side of it.

EQ Keyword equals argument.

GE Keyword is greater than or equal to argument.

GT Keyword is greater than argument.

LE Keyword is less than or equal to argument.

LT Keyword is less than argument.

NE Keyword is not equal to argument.

argument The argument is a 1- to 8-character value to which OMEGAMON compares the keyword. OMEGAMON variables are permitted. The keyword and argument can be any variable name set with the .VAR command.

Examples of setting variables
You can use OMEGAMON variables to build generic screen spaces, pass values to other screen spaces, and alter the flow of screen spaces (.SGO).

For example,

```
.VAR SET &NXTSCRN MONITOR
.VAR SET &DEVICE 123
```

set values for NXTSCRN and DEVICE. You can now use those variables in a screen space.

```
DEV &DEVICE
DIO
.SGO &NXTSCRN
```

OMEGAMON interprets the screen space entries as if you had entered the following commands:

```
DEV 123
DIO
.SGO MONITOR
```
Operational INFO-line and Immediate Commands

Examples of conditionally setting variables

In this example, the variable SYSTEM will be set to A if the variable SWITCH has previously been set to YES; SYSTEM will be set to B if SWITCH has previously been set to NO; and SYSTEM will be set to C if SWITCH has previously been set to MAYBE.

C.VAR SET &SYSTEM A &SWITCH=YES
C.VAR SET &SYSTEM B &SWITCH=NO
C.VAR SET &SYSTEM C &SWITCH=MAYBE

In the following example, the variable TOKEN will be set to MVS only if the product prefix is OM.

C.VAR SET &TOKEN MVS PREFIX=OM

<table>
<thead>
<tr>
<th>.VTM</th>
<th>Displays terminal ID and session information for all users logged onto OMEGAMON in VTAM mode.</th>
</tr>
</thead>
</table>

Type Immediate

This command allows a user running in a multi-session environment moderated by KOBVTAM to display information about other KOBVTAM users. You can use this command to monitor and manage access to the VTAM environment. Dedicated mode users do not show up.

Here is an example of the display.

<table>
<thead>
<tr>
<th>.VTM</th>
<th>Userid</th>
<th>Terminal</th>
<th>Mode</th>
<th>Session Start</th>
<th>Last Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>TSOX07</td>
<td>L65F</td>
<td>VTM</td>
<td>02/11/95 11:45:32</td>
<td>02/12/95 13:14:55</td>
</tr>
<tr>
<td>+</td>
<td>TSOX06K</td>
<td>BBLVM06</td>
<td>VTM</td>
<td>02/12/95 13:57:32</td>
<td>IN INITIALIZATION</td>
</tr>
<tr>
<td>+</td>
<td>AFOPER</td>
<td>BBLVM04</td>
<td>VTT</td>
<td>02/12/95 13:57:32</td>
<td>02/12/95 13:57:12</td>
</tr>
<tr>
<td>+</td>
<td>TSOX21</td>
<td>L616A09</td>
<td>VTS</td>
<td>02/12/95 10:23:31</td>
<td>02/12/95 11:24:31</td>
</tr>
<tr>
<td>+</td>
<td>TSOX36A</td>
<td>L674</td>
<td>VTM</td>
<td>02/12/95 13:40:33</td>
<td>02/12/95 13:57:51</td>
</tr>
<tr>
<td>+</td>
<td>TSOX04</td>
<td>L655</td>
<td>VTD</td>
<td>02/12/95 11:35:32</td>
<td>02/12/95 12:14:55</td>
</tr>
</tbody>
</table>

The Mode field indicates the type of session. The possibilities are:

- **VTD**: A director segment running under VTAM in a cross memory or cross system mode session.
- **VTM**: A VTAM mode session.
- **VTS**: An ISPF mode session running under VTAM.
- **VTT**: A TSO mode session running under VTAM.
The Userid field in this display will be blank if external security is not being used to control logon access.

<table>
<thead>
<tr>
<th>/WAIT</th>
<th>Controls synchronization of a cross system or cross memory collector with the director.</th>
</tr>
</thead>
</table>

**Type:** INFO-line  
**Format:** /WAIT ccc

ON  
Specifies that the collector synchronize with the director.  

OFF  
Specifies that the collector not synchronize with the director.

For information on cross system and cross memory modes, see the *Configuration and Customization Guide*.

<table>
<thead>
<tr>
<th>.WAT</th>
<th>Waits nn seconds before executing all commands below.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** .WATnn

The .WAT command provides a delay mechanism for the execution of commands that require information from a currently executing command, such as one that involves calculation of a rate.

<table>
<thead>
<tr>
<th>/XLF OUT</th>
<th>Sends exception logging facility (XLF) data to the printer and resets the log.</th>
</tr>
</thead>
</table>

**Type:** INFO-line

The exception logging facility (XLF) writes exceptions to the XLFLLOG. To view or change the defaults for this file, use the OUTP major command and its minors.

For information about using the XLF feature, see “Automating and Logging Features” on page 331.

<table>
<thead>
<tr>
<th>.XLG</th>
<th>Displays and sets printer characteristics for the Exception Logging Facility (XLFLLOG).</th>
</tr>
</thead>
</table>

**Type:** Immediate

When you enter .XLG, a series of keywords appears for setting printer characteristics. If you are changing more parameters than will fit on one line,
use the OUTP command instead. The keywords for .XLG (and .REP) duplicate the minors of the OUTP major.

**SYSOUT=** SYSOUT class.

**HOLD=** Specifies whether output is to be placed in the hold queue.

**COPIES=** Specifies the number of copies to print.

**FORMS=** Specifies the form on which to print.

**DEST=** Destination, user ID, or both (separated by a period, colon, or slash) to receive report.

**FOLD=** Folds lowercase characters to uppercase.

The following keywords will also appear if their default values have been previously modified. If not, you can type in the keyword and define a new value.

**ID1=** Requests separator pages and page headers. The argument for ID1 can be:

* OMEGAMON generates separator pages and page headers with the appropriate job name printed in block letters on the pages. This is the default.

*NONE* OMEGAMON does not generate page headers or separator pages.

cccccccc OMEGAMON generates separator pages and page headers with cccccccc printed in block letters on the pages. cccccccc is up to 8 user-defined characters.

**ID2=** Defines up to 16 characters on the left of the separator page.

**ID3=** Defines up to 16 characters in the center of the separator page.

**ID4=** Defines up to 16 characters on the right of the separator page.

**DDNAME=** Overrides standard OMEGAMON ddnames.

If you change any parameter other than FOLD=, OMEGAMON automatically spins off the XLFLOG file and creates a new one.

If SYSOUT is active, then DDNAME is inactive and vice versa. The following table shows the parameters in effect and the default settings for SYSOUT or DDNAME.
The zooming feature is designed to simplify the investigation of system conditions by supplying a detailed level of information at the touch of the Zoom key. /ZOOM substitutes the value above the current cursor position for a variable contained in a predefined screen space. The variable substitution allows one zooming screen space to analyze multiple items, such as devices or volume serials. It also enables quick investigation of exception conditions.

Note the following points regarding this feature.

- **Candle ships OMEGAMON with the /ZOOM INFO-line command assigned to PF11 so you can access the zooming screen spaces with a single key. We refer to PF11 as the Zoom key.**

- **The menu system uses the zooming feature extensively. For example, when you are looking at an exception analysis display, you can place your cursor on an exception name and press PF11. OMEGAMON zooms to a recommendation screen that gives you suggestions on actions you may want to take.**

### Parameters and Default Values

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Default Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSOUT=</td>
<td>A (inactive)</td>
</tr>
<tr>
<td>HOLD=</td>
<td>NO (inactive)</td>
</tr>
<tr>
<td>COPIES=</td>
<td>1 (inactive)</td>
</tr>
<tr>
<td>FORMS=</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DEST=</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DDNAME=</td>
<td>OIXLFLOG (inactive)</td>
</tr>
<tr>
<td>FOLD=</td>
<td>YES YES</td>
</tr>
<tr>
<td>ID1=</td>
<td>jobname jobname</td>
</tr>
<tr>
<td>ID2=</td>
<td>(blank) (blank)</td>
</tr>
<tr>
<td>ID3=</td>
<td>(blank) (blank)</td>
</tr>
<tr>
<td>ID4=</td>
<td>(blank) (blank)</td>
</tr>
</tbody>
</table>

**/ZOOM**  Invokes navigational zoom feature using the cursor as a pointer.

**Type:** INFO-line

The zooming feature is designed to simplify the investigation of system conditions by supplying a detailed level of information at the touch of the Zoom key. /ZOOM substitutes the value above the current cursor position for a variable contained in a predefined screen space. The variable substitution allows one zooming screen space to analyze multiple items, such as devices or volume serials. It also enables quick investigation of exception conditions.

Note the following points regarding this feature.

- **Candle ships OMEGAMON with the /ZOOM INFO-line command assigned to PF11 so you can access the zooming screen spaces with a single key. We refer to PF11 as the Zoom key.**

- **The menu system uses the zooming feature extensively. For example, when you are looking at an exception analysis display, you can place your cursor on an exception name and press PF11. OMEGAMON zooms to a recommendation screen that gives you suggestions on actions you may want to take.**
In the menu system, when there are fields on a display that respond to the Zoom key, such as exception names or device names, **Zoom PF11** is shown under the INFO-line as a navigation option.

**Note:** If you want to be able to zoom to exception recommendations from command mode, you can set a command mode PF key to /ZOOM @ZSM.

- You can use zooming in command mode by setting up customized investigative screen spaces and zooming on command or exception names. These screen spaces can contain one or more of the following variables:

  - **&ZOOM** Data found at the cursor location.
  - **&ZOOMC** Command or exception name field (columns 2 to 5).
  - **&ZOOMS** Originating screen space.

For information on how to create screen spaces, see the *OMEGAMON II for IMS User’s Guide*.

OMEGAMON provides some sample zooming screen spaces to help you become familiar with both the setup of a zooming screen space and the types of commands for which /ZOOM is most appropriate. To list the names of those screen spaces, use the SCRN command and look for screen spaces named in the format @ZOMcccc where cccc is a command name. Then try zooming on those command names.

Here is an example of using the zooming feature. You can follow this example either in command mode by entering the DSKB command or in menu mode by choosing the I/O DASD option.

- Execute the major command DSKB. DSKB lists the unit addresses of disks that are currently performing I/O.
To examine one of the disks more closely, place your cursor under the volser you wish to query (in this case MVSA21) and press PF11. OMEGAMON looks for a screen space starting with @ZOM and ending with the major command name that precedes the cursor. OMEGAMON finds the screen space @ZOMDSKB.

/ZOOM replaces the variable &ZOOM with the volser over the cursor. The result might look like this:

You could now place your cursor on a different volser and press PF11 again to receive the same detail on another device. A zooming screen space can display more detailed analyses of any device, volume serial number, or address space.

Valid delimiter characters for zoom values (characters that OMEGAMON recognizes as the beginning or end of the value) are the:

- blank
- + plus sign
- ( ) left and right parenthesis
- | vertical bar
- , single quote
- <-> greater than and less than sign
- = equal sign
- * asterisk
Chapter overview

This chapter provides information about the user profile facility.

Chapter contents

Summary of Profile Definition Commands .......................... 82
Multi-line Input .................................................. 85
Profile Maintenance Commands ................................. 86
Controlling Session and Display Options ......................... 87
Setting Print Output Options .................................. 89
Setting Color Options ......................................... 95
Setting Operational Parameters ................................ 101
Setting Installation Performance Options ....................... 103
**Introduction**

OMEGAMON profiles control the characteristics of an active OMEGAMON session. There are three types of profiles: the Candle-supplied profile, the installation profile, and individual user profiles. Installation profiles are intended to be created by the installer (or other authorized personnel) and specified as the default for the site or for a system. There can be as many individual profiles as you want. Users can create them, save them by assigning them a unique 2-character identifier, and specify their desired profile ID at logon.

Both installation profiles and individual user profiles are created by customizing the settings of the profile definition commands described in this chapter and issuing a profile save command. The save command for an installation profile is IPRF; for a user profile, it is PPRF. You can change the settings of user profile options at any time during a session, and the new settings take effect immediately.

The following table summarizes the profile definition commands. For more information about setting profile options, see *OMEGAMON II for DBCTL User’s Guide*.

<table>
<thead>
<tr>
<th>Command Function</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controlling session and display options</td>
<td>OPTN</td>
<td>Activates and deactivates the:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Automatic Screen Facility (ASF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Timed Screen Facility (TSF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Exception Logging Facility (XLF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- terminal bell</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- log</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Controls these display characteristics:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the date (USA or European)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- minor commands (upper or lower case)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- all command output (upper or mixed case)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- scroll amount (page or cursor)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the first screen when OMEGAMON is started</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the interval for the terminal bell</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> <em>The setting for the ZEROS keyword is not saved in a profile.</em></td>
</tr>
</tbody>
</table>
### Table 3. Profile Definition Commands

<table>
<thead>
<tr>
<th>Command Function</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting print output options</td>
<td>OUTP</td>
<td>The minor command settings saved in a profile are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- COPY: Specifies the number of copies to print.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DDNM: Specifies the ddname to override the standard ddname.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DEST: Specifies the report destination.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FOLD: Specifies whether lowercase is folded to uppercase.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- HOLD: Specifies that the output be placed in the hold queue and retrieved from TSO.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LNCT: Sets the number of lines per page on report.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SOUT: Defines the SYSOUT class of the report.</td>
</tr>
<tr>
<td>Note: Other minors of OUTP are not saved in profiles.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Setting color options</td>
<td>.SCC</td>
<td>Allows you to set color, highlighting, and extended attribute options for each field on the OMEGAMON display.</td>
</tr>
<tr>
<td>Setting operational parameters</td>
<td>.SET</td>
<td>Sets parameters for:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the screen space fetch feature (.FGO)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the interval for OMEGAMON cycles</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the number of entries in the device name table</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- guarding against loops caused by the PEEK command</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the automatic updating delay cycle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the size of the REPORT file for logging screens</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- the size of the work area for the PEEK command</td>
</tr>
<tr>
<td>Setting installation profile options</td>
<td>IOPT</td>
<td>Determines whether OMEGAMON storage is page-fixed and whether OMEGAMON issues a DASD RESERVE when users save a screen space member.</td>
</tr>
</tbody>
</table>
### Table 3. Profile Definition Commands

<table>
<thead>
<tr>
<th>Command Function</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defining Timed Screen Facility entries</td>
<td>.TSF</td>
<td>See “Automating and Logging Features” on page 331.</td>
</tr>
<tr>
<td>Setting exception analysis options</td>
<td></td>
<td>See table in “Commands for Customizing and Creating Exceptions” on page 105.</td>
</tr>
<tr>
<td>Customizing IMS messages</td>
<td>MSGD</td>
<td>Assigns names to IMS messages for use in exception analysis. See the full description of this command in “Commands for Customizing and Creating Exceptions” on page 105.</td>
</tr>
</tbody>
</table>
Introduction

Some profile-definition commands in this chapter use a multi-line input facility (for example, OPTN, .SCC, and .SET). When you enter these commands, OMEGAMON displays the current settings for their options in a multi-line table.

The following considerations apply to multi-line tables:

- You can modify any display line that begins with a colon in column 1.
- Blanking out fields has no effect; OMEGAMON redisplays the line on the next cycle.
- To change a setting, type over the displayed value and press ENTER.
Profile Maintenance Commands

Introduction

The profile maintenance commands are IPRF (for the installation profile) and PPRF (for user profiles).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPRF</td>
<td>Saves or deletes the installation-defined default profile.</td>
</tr>
<tr>
<td>PPRF</td>
<td>Saves, deletes, comments, or lists the user’s session profile.</td>
</tr>
</tbody>
</table>

**IPRF**

Type: Immediate

Format: **IPRF SAVE|DELETE**

- **SAVE** Saves the installation profile in the dataset defined by the RKOIPFSV DD statement.
- **DELETE** Deletes the installation profile from the dataset defined by the RKOIPFSV DD statement.

OMEGAMON automatically assigns the 2-character code /I to the installation profile. /I is used on the USER= startup parameter to load this profile, and it appears on the INFO-line during the session.

The installation-defined profile for OMEGAMON for IMS is stored as member name OIINSTAL.

**PPRF**

Type: Immediate

Format: **PPRF SAVE|DELETE|COMMENT|LIST  cc**

- **SAVE** Saves the user profile in the dataset defined by the RKOIPFSV DD statement.
- **DELETE** Deletes the user profile from the dataset defined by the RKOIPFSV DD statement.
- **COMMENT** Displays a description of the current profile and allows you to change it dynamically.
- **LIST** Lists all members of the user profile dataset.
- **cc** Specifies the 2-character user profile identifier. To start subsequent sessions with this profile, specify this value in the USER= startup parameter.

If you do not specify cc, the PPRF command uses the current value of the USER= startup parameter.
Controlling Session and Display Options

Introduction

Parameters of the OPTN command set session and display options.

<table>
<thead>
<tr>
<th>OPTN</th>
<th>Sets session control and display options.</th>
</tr>
</thead>
</table>

**Type:** Immediate

OPTN displays its current settings in the following format:

```
OPTN
:ASF = OFF     BELL = OFF
:BELLINT = 60.00 DATEFORMAT = USA
:FIRSTSCREEN = KOINITZZ LOG = OFF
:MINORCASE = LOWER SCREENCASE = MIX
:SCROLL = PAGE TSF = OFF
:XLF = OFF ZEROS = OFF
```

**ASF**

Turns the Automatic Screen Facility (ASF) ON or OFF. See “Automating and Logging Features” on page 331 for information about this facility.

**BELL**

Turns the audible alarm ON or OFF.

**BELLINT**

Sets the minimum interval (in seconds) between rings of the bell. A valid value is an integer between 5.00 and 99.00.

**DATEFORMAT**

Sets display format for the date (mm/dd/yy or dd/mm/yy). Valid values are USA or EUROPEAN.

**FIRSTSCREEN**

Identifies the name of the first screen space to execute, or it might FGO to another screen space.

**LOG**

Turns the log ON or OFF.

**MINORCASE**

Sets the display case for minor commands. Valid values are UPPER and LOWER.

**SCREENCASE**

Sets the display case for screen output. Valid values are UPPER and MIX.

**SCROLL**

Sets the default scroll amount. Valid values are PAGE, which scrolls an entire screen at a time, and CSR, which scrolls from the cursor position.

**TSF**

Turns the Timed Screen Facility (TSF) ON or OFF. See “Automating and Logging Features” on page 331 for information about this facility.
Controlling Session and Display Options

**XLF**
Turns the Exception Logging Facility (XLF) ON or OFF. See “Automating and Logging Features” on page 331 for information about this facility.

**ZEROS**
Sets the way in which zeros display on your terminal. When this parameter is ON, OMEGAMON displays the numeral 0 in fields that have a value of zero. When it is OFF, OMEGAMON displays a blank in these fields. This value is not saved in a profile.
Setting Print Output Options

Introduction

The OUTP major command and its minor commands control the printing of XLFLOG and REPORT files.

<table>
<thead>
<tr>
<th>OUTP</th>
<th>Controls the characteristics of log files.</th>
</tr>
</thead>
</table>

**Type:** Major  
**Format:** OUTP ccccc

The variable ccccc is either XLFLOG or REPORT.

The OUTP major command displays column headings for pending and current values associated with all of its minor commands follows.

| OUTP REPORT + | ----- Pending ------ | ----- Current ------ |

When you type an OUTP minor command followed by the new value you want to assign and press ENTER, the new value displays in the pending value column. It will become the active value when you reallocate the log with /LOGOUT, .LOGOUT, /XLFLOG, or .XLFLOG.

The log will be routed to one of the following dataset types:

- SYSSOUT, the initial dataset type for OUTP. This file is designated FREE=CLOSE. This means that every time you enter the /LOGOUT, LOGOUT, /XLFOUT, or XLFOUT command, the REPORT log or the XLFLOG automatically spins off and is available for printing.

A sequential dataset, with a ddname you specify. Its DCB attributes are LRECL=nn, where nn is the screen column width plus 1; RECFM=FBA; and BLKSIZE is a multiple of LRECL. The DISP parameter can be set to MOD, SHR, or OLD.

**Caution**

Logging to a dataset is optional and is not recommended. If you use DISP=SHR or DISP=OLD, any action to close and reopen the log file reinitializes the dataset and deletes any existing log information.

Only one session at a time per address space should use a specific ddname. Otherwise, you get interleaved output.
Note: Some prior releases of OMEGAMON used ddname OIREPORT if it was present. This release also looks for ddname OIREPORT if present, but will access it only if it is not currently in use.

Each OUTP minor command controls one XLFLOG or REPORT file characteristic. That characteristic is under the control of ddname or SYSOUT. When SYSOUT is active, ddname is inactive, and vice versa. Parentheses around a value in the OUTP display indicate that it is currently inactive. The following table shows the minors and the initial settings for the SYSOUT or ddname datasets.

Table 4. OUTP Minor Commands with Initial Settings

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Initial Values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SYSOUT</td>
</tr>
<tr>
<td>SOUT</td>
<td>A (inactive)</td>
</tr>
<tr>
<td>HOLD</td>
<td>NO (inactive)</td>
</tr>
<tr>
<td>COPY</td>
<td>1 (inactive)</td>
</tr>
<tr>
<td>FORM</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DEST</td>
<td><em>NONE</em> (inactive)</td>
</tr>
<tr>
<td>DSTU</td>
<td>userID (inactive)</td>
</tr>
<tr>
<td>DDNM</td>
<td>(inactive) *DYNAMIC</td>
</tr>
<tr>
<td>FOLD</td>
<td>YES</td>
</tr>
<tr>
<td>LNCT</td>
<td>60</td>
</tr>
<tr>
<td>ID1</td>
<td>jobname</td>
</tr>
<tr>
<td>ID2</td>
<td>(blank)</td>
</tr>
<tr>
<td>ID3</td>
<td>(blank)</td>
</tr>
<tr>
<td>ID4</td>
<td>(blank)</td>
</tr>
</tbody>
</table>

COPY

| Type: Minor of OUTP |
| Format: COPY nn |

The COPY minor is under SYSOUT control.

If you set the HOLD minor command to YES, OMEGAMON ignores the COPY command. If the value you type is not valid, OMEGAMON redisplay it where you typed it and does not transfer it to the pending column.
### Setting Print Output Options

<table>
<thead>
<tr>
<th><strong>DDNM</strong></th>
<th>Directs output to a particular ddname.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of OUTP</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><strong>DDNM cccccccc</strong></td>
</tr>
</tbody>
</table>

The DDNM minor removes control from SYSOUT.

If the value you type is not valid, OMEGAMON redisplays it where you typed it and does not transfer it to the pending column.

<table>
<thead>
<tr>
<th><strong>DEST</strong></th>
<th>Specifies the destination to receive the output.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of OUTP</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><strong>DEST cccccccc</strong></td>
</tr>
</tbody>
</table>

The default is *NONE*, which sends output to the local printer. The destination can be a terminal, a node, a remote work station, a local device or group of devices, or a user ID.

Your site determines valid destinations. OMEGAMON checks your site’s table for valid destinations only when you reset the log (with .LOGOUT or .XLFOUT). During initialization, OMEGAMON checks only syntax validity.

If the destination is a specific user ID (at the device destination), use the DSTU minor command for the user ID.

<table>
<thead>
<tr>
<th><strong>DSTU</strong></th>
<th>Specifies the destination user ID to receive a report.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of OUTP</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><strong>DSTU cccccccc</strong></td>
</tr>
</tbody>
</table>

The default is *NONE*, which sends output to the local printer. Enter the destination user ID in the format established for your site.

Your site determines valid destinations. OMEGAMON checks your site’s JES parameters for valid destinations only when you reset the log (with .LOGOUT or .XLFOUT). During initialization, OMEGAMON checks only syntax validity.
Setting Print Output Options

<table>
<thead>
<tr>
<th><strong>FOLD</strong></th>
<th>Changes lowercase characters to uppercase.</th>
</tr>
</thead>
</table>

FOLD is active under ddname or SYSOUT.

**Type:** Minor of OUTP  
**Format:** `FOLD ccc`

- **YES**  
  Lowercase characters are changed to uppercase before printing (default).
- **NO**  
  Lowercase characters are not changed to uppercase before printing.

If the value you type is not valid, OMEGAMON redisplayes it where you typed it and does not transfer it to the pending column.

<table>
<thead>
<tr>
<th><strong>FORM</strong></th>
<th>Specifies the name of the form on which to print.</th>
</tr>
</thead>
</table>

The FORM minor is under SYSOUT control.

**Type:** Minor of OUTP  
**Format:** `FORM cccc`

The default value is *NONE*. This means that OMEGAMON uses the form defined as the default for the printer at that destination.

If the value you type is not valid, OMEGAMON redisplayes it where you typed it and does not transfer it to the pending column.

If you change *NONE* to a value (such as HOLE), and you then want to return to *NONE*, type an asterisk (*).

<table>
<thead>
<tr>
<th><strong>HOLD</strong></th>
<th>Specifies that output be placed in the hold queue.</th>
</tr>
</thead>
</table>

The HOLD minor is under SYSOUT control.

**Type:** Minor of OUTP  
**Format:** `HOLD cccc`

- **YES**  
  Places the print file on the hold queue, and allows it to be retrieved from TSO.
- **NO**  
  Does not place the print file on the hold queue (default).

If the value you type is not valid, OMEGAMON redisplayes it where you typed it and does not transfer it to the pending column.
ID1 | Requests separator pages and page headers that identify output from different OMEGAMON sessions.

ID1 is active under ddname or SYSOUT.

Type: Minor of OUTP

Format: ID1 cccccccc

* OMEGAMON generates separator pages and page headers with the appropriate job name printed in block letters on the pages. This is the default.

*NONE* OMEGAMON does not generate separator pages and page headers.

cccccccc OMEGAMON generates separator pages and page headers with cccccccc printed in block letters on the pages. cccccccc is up to 8 user-defined characters.

ID2 | Defines up to 16 characters at the left of the separator page.

ID2 is active under ddname or SYSOUT.

Type: Minor of OUTP

Format: ID2 ccc...ccc

When separator pages and page headers are requested with ID1, ID2 can define up to 16 characters to appear justified below the block letters on the left of the separator page.

ID3 | Defines up to 16 characters in the center of the separator

ID3 is active under ddname or SYSOUT.

Type: Minor of OUTP

Format: ID3 ccc...ccc

When separator pages and page headers are requested with ID1, ID3 can define up to 16 characters to appear centered below the block letters on the separator page.
ID4 | Defines up to 16 characters at the right of the separator page.

ID4 is active under ddname or SYSOUT.

Type: Minor of OUTP
Format: ID4 ccc...ccc

When separator pages and page headers are requested with ID1, ID4 can define up to 16 characters to appear justified below the block letters on the right of the separator page.

LNCT | Sets the number of lines per page for the REPORT or XLFLOG file output.

ID4 is active under ddname or SYSOUT.

Type: Minor of OUTP
Format: LNCT nn

SOUT | Removes control from a sequential dataset, and directs it to the specified SYSOUT class.

The SOUT minor removes control from the ddname currently used for your XLFLOG or REPORT log and directs it to SYSOUT.

Type: Minor of OUTP
Format: SOUT c

The variable c represents the SYSOUT class.
Setting Color Options

Introduction

The .SCC command controls color or highlighting.

<table>
<thead>
<tr>
<th>.SCC</th>
<th>Sets display color or highlighting for text and commands by type.</th>
</tr>
</thead>
</table>

**Type:** Immediate

The .SCC command displays and sets options that determine how highlighting and/or color is used when an OMEGAMON screen is sent to a user’s terminal. Because display characteristics depend on the type of terminal you are supporting, .SCC contains keywords that, in combination, will accommodate any of the various 3270-type devices. Here is the screen display format:

```
.SCC
: Display=cccccc ProfileDefinitionMode=OFF
: ExtendedHighlighting=ccc
+ : Major=cccccccc Minor=cccccccc Immed=cccccccc Default=cccccccc
+ XACB Display Options:
  : Clr1=cccccccc Clr2=cccccccc Clr3=cccccccc Clr4=cccccccc
  : Clr5=cccccccc Clr6=cccccccc Clr7=cccccccc
```
Device support

Color/highlighting capabilities are separated into the following device categories:

**non-EDS**

Devices that do not support an extended data stream (EDS), but support highlighting. There are two types of non-EDS devices:

- Monochrome non-EDS devices, which use high or low intensity to distinguish fields in a display.
- Color non-EDS devices, which use different colors rather than high and low intensity to distinguish fields in a display. The colors displayed depend on whether the device is a 2-base color or 4-base color device, and whether a field is protected or unprotected. For example, most 4-base color devices display high intensity fields in red (unprotected) and white (protected), and low intensity fields in green (unprotected) and blue (protected).

**Monochrome EDS**

Monochrome devices that support the extended data stream. A field might be displayed in either high or low intensity in combination with the extended highlighting attributes (blinking, reverse video, and underscoring).

**Color EDS**

Color devices that support the extended data stream. A field might be displayed in any of the 7 extended colors in combination with the extended highlighting attributes.

Profile definition mode

Before customizing your color/highlighting definitions, determine which mode of .SCC is appropriate. The mode is controlled by the ProfileDefinitionMode keyword, as shown below.

```
.SCC
:  Display=ccccc
:  ExtendedHighlighting=ccc
ProfileDefinitionMode=ON/OFF
```
Setting Color Options

OFF
Changes to .SCC affect only the current OMEGAMON session.

ON
Intended for use when creating or changing a user profile without altering the current session. To change your settings without altering your current session, turn Profile Definition mode on, and alter the settings. After defining the settings, issue a profile save command. Then, change the settings back, and turn Profile Definition mode off.

The new settings might also take effect during the current session if you subsequently change the value of the DISPLAY= keyword. Setting this keyword to ON enables you to configure options for different types of terminals on the same screen.

Keywords and values
The following sections describe the rest of the .SCC keywords and valid values.

For any keyword or value, you need only type as many letters as it takes to make an entry unique.

You can type any keyword and value on the command line after .SCC and bypass the multi-line display. The command comments itself out and flags itself DONE at the end of the line. This facility allows for changes to .SCC values from within screen spaces.

Display intensity or color
The DISPLAY keyword has 4 possible settings.

\[ .SCC : \text{Display}=\text{BASIC}|\text{HIGH}|\text{LOW}|\text{COLOR} \]

BASIC
Allows you to set HIGH or LOW intensity for fields on monochrome terminals or non-EDS color terminals.

Note: When the BASIC display option is used on color devices that can support EDS, OMEGAMON will treat that device as a 4-base color terminal (non-EDS).

HIGH
Specifies that all fields be displayed in high intensity.

LOW
Specifies that all fields be displayed in low intensity.

COLOR
For color EDS terminals only, allows you to specify the color of each field.
Setting Color Options

Note: In cases where OMEGAMON is given a color value instead of an intensity value or vice versa, it makes the following internal conversion:

- On a non-EDS terminal, values of Green and Blue translate to low intensity; all other color values translate to high intensity.
- On an EDS terminal, a value of HI translates to the color Red; a value of LO translates to the color Green.

Extended highlighting

The EXTENDEDHIGHLIGHTING keyword is used only with devices that support the extended data stream.

: ExtendedHighlighting=ON|OFF

OFF

Extended highlighting features are not available. Use with non-EDS devices.

ON

Extended highlighting features are available. When DISPLAY=COLOR, this value is automatically set to ON. Extended attributes are not supported in ISPF mode.

Caution

Do not set EXTENDEDHIGHLIGHTING=ON unless you have a terminal that supports an extended data stream (or unless you have PROFILEDEFINITIONMODE=ON). If you do this accidentally, you may get a PROGnnnn or a screen erasure error. You may press the ATTN or PA1 key to resume the session, but be aware that this action also clears the current security authorization and the current screen space.

Display fields

Following is the display format of .SCC with PROFILEDEFINITIONMODE=ON. It shows the display field keywords and valid values.

For the DISPLAY=COLOR option, color names can be Red, Green, White, Blue, Pink, Yellow, or Turquoise.

For the DISPLAY=BASIC option, highlighting values can be HIGH or LOW.

The value of the DEFAULT keyword can be used as a variable definition for the Major, Minor, Immed, and XACB Display Options. In the figure below, the default value is abbreviated as DEF.
Setting Color Options

The field names that you can control with .SCC are:

**Major**  Controls color or highlighting for major commands and their output.

**Minor**  Controls color or highlighting for minor commands and their output.

**Immed**  Controls color or highlighting for immediate commands and their output.

**Default**  Controls color or highlighting for other unprotected fields (for example, error message text, help text).

**XACB Options**  Controls exception analysis message text. The keywords Clr1 through Clr7 can be used as substitutes for the color names (Red, Blue, and so on) or highlighting (HI or LO) when customizing exception messages with the XACB command.
The following table shows the possible color and highlighting setting variations according to the type of terminal you have.

### Table 5. Color/Highlighting Settings in .SCC

<table>
<thead>
<tr>
<th>If you have...</th>
<th>and you want...</th>
<th>keywords and possible settings are...</th>
</tr>
</thead>
</table>
| A non-EDS terminal | some fields in high intensity and some fields in low intensity | DISPLAY=BASIC  
EXTENDEDHIGHLIGHTING=OFF  
Major, Minor, Immed, XACB  
options=HI|LO|DEF  
DEFAULT=HI|LO |
| | all fields in high intensity | DISPLAY=HI  
EXTENDEDHIGHLIGHTING=OFF  
(All other settings default to HI.) |
| | all fields in low intensity | DISPLAY=LO  
EXTENDEDHIGHLIGHTING=OFF  
(All other settings default to LO.) |
| A monochrome EDS terminal | some fields in high intensity and some fields in low intensity, plus blinking, underscoring, or reverse video | DISPLAY=BASIC  
EXTENDEDHIGHLIGHTING=ON  
Major, Minor, Immed, XACB  
options=HI|LO|DEF  
DEFAULT=HI|LO |
| | all fields in high intensity, plus blinking, underscoring, or reverse video | DISPLAY=HI  
EXTENDEDHIGHLIGHTING=ON  
(All other settings default to HI.) |
| | all fields in low intensity plus blinking, underscoring, or reverse video | DISPLAY=LO  
EXTENDEDHIGHLIGHTING=ON  
(All other settings default to LO.) |
| A color EDS terminal | to specify the color of each field, plus blinking, underscoring, or reverse video | DISPLAY=COLOR  
EXTENDEDHIGHLIGHTING=ON  
Major, Minor, Immed, XACB  
options=color|DEF  
DEFAULT=COLOR |
Setting Operational Parameters

Introduction

The .SET immediate command displays data about the OMEGAMON environment in table form. You can change data on any of the output lines.

| .SET | Sets operation control parameters. |

Type: Immediate

| Format: |

```plaintext
.SET FGOLIMIT = 64 FGOLOOP = OFF
.SET GDEVUCBS = 200 INTERVAL = 10.00
.SET LOOPTIME = 5.00 PAGELIMIT = 15000
.SET PEEKSIZE = 4096 STATUSMODE = OFF
```

To change a setting, move the cursor to the value you want to change, type the new value over the current value, and press Enter.

- **FGOLIMIT**: Specifies the maximum number of consecutive .FGO screens that can execute before OMEGAMON detects a loop and sets .FGOLOOP=ON. The maximum number is 1000.

- **FGOLOOP**: Turns .FGO screen loop detection ON or OFF. If OMEGAMON detects an .FGO loop, it sets this keyword to ON and then treats subsequent .FGO commands as .SGO commands. (.FGO executes a screen space without displaying it; .SGO displays each screen space it executes.) You can also set .FGOLOOP=ON yourself if you want to test screen spaces you have linked together with .FGO commands.

- **GDEVUCBS**: Specifies the number of entries in the device name table for the GDEV command. The maximum number is 4000.

- **INTERVAL**: Specifies the interval in seconds between automatic updates. This interval is called an OMEGAMON cycle. The following restrictions apply:
  - This value is effective only in dedicated mode or VTAM mode. You can, however, be operating in any mode when you define a new value for the purpose of saving it in a user profile.
  - The maximum interval is 99.00 seconds.
  - VTAM mode does not allow an interval shorter than 5 seconds.
  - Dedicated mode does not allow an interval shorter than .5 seconds.
**IODELAY**
In automatic updating, the number of cycles to delay the next screen refresh after you have moved the cursor. The maximum number is 100.

**LOOPCOUNT**
The maximum number of control blocks that the PEEK command can test before OMEGAMON detects a loop. The valid range is 1–60000.

**LOOPTIME**
The threshold (in seconds) for OMEGAMON built-in loop detection. The default is 5 seconds. The maximum value is 99.

**PAGELIMIT**
The size (in pages) of the REPORT file used to log OMEGAMON screens. The maximum is 99999. This number dynamically decreases as the log is printing to reflect the number of pages left before the limit is reached. Consequently, be sure to check this parameter (and reset it, if necessary) before saving a profile.

**PEEKSIZE**
The size (in bytes) of the PEEK buffer. The maximum is 204800.

**STATUSMODE**
This parameter does not apply to OMEGAMON for IMS.
Setting Installation Performance Options

Introduction

The IOPT command sets performance options.

<table>
<thead>
<tr>
<th>IOPT</th>
<th>Assigns global OMEGAMON performance options.</th>
</tr>
</thead>
</table>

Type: Immediate

Because this command determines how OMEGAMON runs for the entire site, sites might want to use this command at installation time and then restrict access to it.

```
IOPT
: NONSWAP = OFF
: PAGEFIX = OFF
: RESERVE = OFF
: TSOPFIX = OFF
```

**NONSWAP**

This parameter does not apply to OMEGAMON for IMS.

**PAGEFIX**

Indicates whether OMEGAMON storage is page-fixed. Possible values for this are ON and OFF; the default is OFF. If you change the value to ON, your change does not take effect until the next session.

This option requires that OMEGAMON be APF authorized.

**RESERVE**

Indicates whether OMEGAMON issues a DASD RESERVE when users save a member in the PROCSV dataset. Possible values for this are ON and OFF; the default is OFF.

**TSOPFIX**

This parameter does not apply to OMEGAMON for IMS.

You can enter the IOPT command followed by the keyword and setting you want on the same line.
Chapter overview

This chapter provides information about the commands you can use to customize and create exceptions.

Chapter contents

Customizing and Creating Exceptions for User Profiles . . . . . . . . . . . . . .  106
Customizing Profiles for Users Who Log On through the CUA Interface .  107
Commands for Controlling Exception Analysis . . . . . . . . . . . . . . . . . . . .  108
Resource Contention Exception Commands. . . . . . . . . . . . . . . . . . . . . .  123
Customizing and Creating Exceptions for User Profiles

Introduction

To better monitor your system, you can customize OMEGAMON II for DBCTL’s exceptions, and also create your own exceptions for your user profiles.

There are two types of user profiles:

**Type1** Profile that one or more users is going to specify when logging on to the CUA interface

**Type2** Profile that users logging on directly to the menu/command interface, in dedicated, VTAM, ISPF, or TSO mode, use exclusively

This chapter explains the commands you use to create your own exceptions, which appear in the menu/command interface. This chapter also explains the commands you use to customize exceptions for user profile type 2. If you are working with the type 2 user profile, you can use any of the commands in this chapter.

For detailed procedural information about customizing exceptions for either type of profile, and using exception analysis to improve your system’s performance, see your OMEGAMON II for DBCTL User’s Guide.

<table>
<thead>
<tr>
<th>Warning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do not use the XACB command on the menu interface Profile Menu to customize exceptions for a type 1 profile.</td>
</tr>
</tbody>
</table>

If you are customizing the type 1 user profile, you must customize that profile’s exceptions in the CUA interface using the customizing profiles procedure in the OMEGAMON II for DBCTL User’s Guide.

The first time any user logs on to OMEGAMON II for DBCTL through the CUA interface with a given profile, OMEGAMON II for DBCTL builds an exceptions table using the specified profile’s existing exceptions values. These values can be either the original defaults, or values any user specified previously, either in the CUA interface or using commands.

If you use the XACB command or the Profile Menu (options A through E) of the OMEGAMON II for DBCTL menu interface to customize a profile’s exceptions, OMEGAMON II for DBCTL writes over those values with the values in the exceptions table the next time any user logs on with that profile through the CUA interface.

For this reason, Candle recommends that you not use any of the commands documented in this chapter to customize the OMEGAMON II for DBCTL exceptions associated with a type 1 profile. You should use these commands only to create your own exceptions, which appear in the menu/command interface.
Introduction

If you are customizing a user profile that even one user is going to specify when logging on to OMEGAMON II’s CUA interface, you should use the commands documented in this chapter only to create your own exceptions. There are 4 commands that you need to create and work with your own exceptions:

- **MSGD**: Use to define IMS messages for exception analysis.
- **PMSG**: Use to display IMS messages that you define using the MSGD command.
- **XACB**: Use to customize Mnnn exceptions.
  
  **Note**: This is the only recommended use of XACB for customizing this type of user profile.
- **XIMS**: Use to invoke exception analysis for all groups.
  
  **Note**: Use only in conjunction with the PMSG command when you are working with this type of user profile.

See “Commands for Controlling Exception Analysis” on page 108 for detailed explanations of these commands.
## Commands for Controlling Exception Analysis

### Introduction

With the exception analysis immediate commands, you can control exception analysis and display current status. The following table summarizes these commands:

<table>
<thead>
<tr>
<th>Command Function</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting exception analysis</td>
<td>XIMS</td>
<td>Invokes exception analysis for all exceptions.</td>
</tr>
<tr>
<td></td>
<td>XGRP</td>
<td>Invokes exception analysis for the specified exception group.</td>
</tr>
<tr>
<td>Summarizing exception activity</td>
<td>XSUM</td>
<td>Displays a summary of the last and worst trip values for all exceptions.</td>
</tr>
<tr>
<td></td>
<td>XTRP</td>
<td>Displays a summary of the last and worst trip values for tripped exceptions.</td>
</tr>
<tr>
<td>Defining exception characteristics</td>
<td>GDFN</td>
<td>Displays existing groups (both Candle-defined and user-defined).</td>
</tr>
<tr>
<td></td>
<td>LEXC</td>
<td>Allows you to define new groups.</td>
</tr>
<tr>
<td></td>
<td>MSGD</td>
<td>Allows you to specify the order for OMEGAMON to sample exceptions and display messages.</td>
</tr>
<tr>
<td></td>
<td>XACB</td>
<td>Defines IMS messages for exception analysis.</td>
</tr>
<tr>
<td></td>
<td>XTXT</td>
<td>For each individual exception, XACB allows you to:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- set a threshold</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- control the state (for example, ON or OFF)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- define display characteristics for warnings</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- request audible alarm when the exception trips</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- control the frequency for OMEGAMON sampling</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- define parameters for XLF and ASF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Allows you to define the message that displays when no exceptions have tripped.</td>
</tr>
</tbody>
</table>
Commands for Controlling Exception Analysis

Starting exception analysis

The XGRP and XIMS commands start exception analysis for selected groups or for all groups.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XGRP</td>
<td>Invokes exception analysis for exception group cc.</td>
</tr>
</tbody>
</table>

The XGRP command invokes only those exceptions belonging to the specified group.

Type: Immediate

Format: XGRPcc

The variable cc is the ID for a group that has been defined with the GDFN command or for a Candle-supplied default group. Use GDFN to display currently defined groups.

- **AL**: pool utilization exceptions
- **DB**: database exceptions
- **DL**: DASD logging exceptions
- **FA**: Fast Path exceptions
- **FR**: largest free block (fragmentation exceptions)
- **IM**: IMS internal exceptions
- **IV**: IMS virtual storage exceptions
- **OS**: MVS resource exceptions
- **ST**: static exceptions
- **VS**: VSAM exceptions
- **XR**: XRF exceptions

---

Note: Definitions for the GDFN, LEXC, MSGD, XACB, and XTXT commands can be saved in a profile. See “User Profile Facility” on page 81.

Table 6. Exception Analysis Control Commands

<table>
<thead>
<tr>
<th>Command Function</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controlling Groups</td>
<td>XGSW</td>
<td>Allows you to control the state (for example, ON or OFF) of exceptions by group.</td>
</tr>
</tbody>
</table>

---

Commands for Customizing and Creating Exceptions 109
For example, to invoke only the exceptions that belong to the IMS internal group (IM), enter XGRPIM as shown below:

```
XGRPIM   OMEGAMON for IMS Group Exception Analysis
+      Message Dequeue rate = .06/second (Low)
+      Output Queue length for logical terminal ‘MTO’ = 5
+      Output Queue length for logical terminal ‘MTOPRINT’ = 5
```

<table>
<thead>
<tr>
<th>XIMS</th>
<th>Invokes exception analysis for all groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
<tr>
<td>Format:</td>
<td>cXIMS</td>
</tr>
</tbody>
</table>

Use the label L to display the 4-character exception names invoked by XIMS.

The XIMS command can invoke over 150 different exception conditions. Exception analysis groups the exceptions into exception groups. See the XGRP command for a list of exception groups.

You can turn each exception on or off individually. Some exceptions require that you set a threshold. If a 3279 color display terminal is available, you can assign each exception message a specific color, which causes important messages to stand out. If the OMEGAMON terminal supports the audible alarm (or bell), you can request that the alarm ring when certain exceptions occur. You can set the options for all of these exceptions with the XACB command and save your settings in a profile with the PPRF command. See “User Profile Facility” on page 81 for an explanation of PPRF.

The static exceptions (type ST) are state-driven; they indicate whether monitored conditions are ON or OFF (such as traces being active or inactive).

By default, XIMS does not display a message for some static exceptions that persist for more than 5 cycles. However, you can use the XGRP command to display static exceptions at any time. If you want to change the setting for an exception in the static group so that it will display after five cycles, use the STOP= parameter of XACB.

**Summarizing exception activity**

The XSUM and XTRP commands show you exception activity.

```
XSUM        Displays a summary of exceptions and their current status.
```

| Type:       | Immediate                              |
| Format:     | XSUM GROUP={A|I} LIST={A|I} RESET       |
**GROUP**

The group ID (cc) can be any of the same exception groups used with the XGRP command. Enter a 2-character group ID to summarize the exceptions of one group only.

**LIST**

The value can be A or I.
- **A** Lists exceptions in alphabetical order. This is the default.
- **I** Lists exceptions in the order in which they are executed, as specified by the LEXC command.

**RESET**

Resets the last and worst values back to zero. Does not reset the cumulative value.

The following figure shows an example of a partial XSUM display.

```
XSUM
+----------+----------+-----------+---------------+------------+---------------+
+ DNRS      Threshold  Trip Value  Time Occurred  Total Trips  Trips Since Reset
+ State=On
+ Last 06/18  17:07:46 14 14
+ Worst 06/18  17:07:46
+ Group=OS  Limit=3  Persist=2  Auto=OFF  Log=NO
+----------+----------+-----------+---------------+------------+---------------+
+ DRDY      Threshold  Trip Value  Time Occurred  Total Trips  Trips Since Reset
+ State=On
+ Last NOT TRIPPED 0 0
+ Worst
+ Group=OS  Limit=3  Persist=2  Auto=OFF  Log=NO
+----------+----------+-----------+---------------+------------+---------------+
+ WSHI      Threshold  Trip Value  Time Occurred  Total Trips  Trips Since Reset
+ State=Test  2500
+ Last 3640K 06/18  16:09:30 6 8
+ Worst 3650K 06/18  16:07:46
+ Group=OS  Limit=3  Persist=2  Auto=OFF  Log=NO
+----------+----------+-----------+---------------+------------+---------------+
+ WSLO      Threshold  Trip Value  Time Occurred  Total Trips  Trips Since Reset
+ State=Test  300
+ Last 270K 06/18  16:09:30 5 9
+ Worst 265K 06/18  16:07:46
+ Group=OS  Limit=3  Persist=2  Auto=OFF  Log=NO
```

The XSUM command displays the current settings for the exception state (*State=ON/OFF/TEST*), the group to which it is assigned (*Group=*), and XLF or ASF settings (*Limit=*, *Persist=*, *Auto=*, and *Log=*). In addition, it displays the last and worst values for the following fields:

- **Threshold**
  - The current threshold value set for this exception.
- **Trip Value**
  - The value that caused this exception to trip.
- **Time Occurred**
  - The date and time the exception last exceeded its threshold.
The XTRP display is the same as the XSUM display, but shows only tripped exceptions rather than all exceptions.

### Defining exception characteristics

This section lists the commands that allow you to customize your exceptions and save the new definitions in a user profile.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XTRP</td>
<td>Displays a summary of tripped exceptions for a group.</td>
</tr>
</tbody>
</table>

**Type:** Immediate  
**Format:**  

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(blank)</td>
<td>XTRP with no keywords displays all of the exception groups in alphabetical order. This is the default.</td>
</tr>
<tr>
<td>GROUP</td>
<td>The group ID (cc) can be any of the same exception groups used with the XGRP command. Enter a 2-character group ID to display the exceptions of one group only.</td>
</tr>
<tr>
<td>LIST</td>
<td>The value can be A or I.</td>
</tr>
<tr>
<td>A</td>
<td>Lists exceptions in alphabetical order. This is the default.</td>
</tr>
<tr>
<td>I</td>
<td>Lists exceptions in the order in which they are executed, as specified by the LEXC command.</td>
</tr>
<tr>
<td>RESET</td>
<td>Resets the last and worst values back to zero. Does not reset the cumulative value.</td>
</tr>
</tbody>
</table>

With this command, exceptions can be organized by groups such as hardware, software, system services, critical applications, tape and disk drives, and online applications. Then, when you invoke exception analysis by group with the XGRP command, critical and related exceptions appear together on the display.
GDFN

[GROUP=cc]
[LIST=cccc,aaaa NAME=‘ccc...ccc’]
[POSITION=nn]
[DELETE=EXCEPTION LIST=cccc,aaaa]
[DELETE=GROUP]

(Blank) Lists user-defined and Candle-defined exception groups along with the exceptions included in each group.

GROUP=cc Specifies the 2-character exception group ID. A group ID can be any two unique alphanumeric characters. To list the entries for an existing group, enter this keyword and the group ID.

LIST=cccc,aaaa,... Lists and adds exceptions to the exception group specified with the GROUP= keyword. An exception may be associated with only one group at a time.

NAME=’ccc...ccc’ Specifies a 25-character user-defined description of the exception group. Enclose in single quotes if there are blanks, special characters, or single quotes in the name.

POSITION=nn Specifies the order in which GDFN displays defined groups. The variable nn is a position number for the specified group relative to the other groups.

DELETE=EXCEPTION Deletes exceptions specified with LIST= from the group specified by the GROUP= keyword. Note that the syntax does not allow you to use the DELETE keyword followed by a list of exceptions. You must enter the LIST keyword.

DELETE=GROUP Deletes the entire exception group specified by the GROUP= keyword.

For example, to define the group SP and its related exceptions, enter:

GDFN GROUP=SP POSITION=1 NAME=’SYSTEMS PROGRAMMER’
GDFN GROUP=SP
LIST=DISP,ROLO,LDMB,PBTR,DNRS,TNRS,DRDY,TRDY
GDFN GROUP=SP LIST=WSHI,IMHI,IMLO,PIMC,ACWA,ACEA

To delete specific exceptions from group TX, enter:

GDFN GROUP=TX DELETE=EXCEPTION LIST=WSHI,WSLO

To delete the group TX and all its related exceptions, enter:
Commands for Controlling Exception Analysis

**GDFN GROUP=TX DELETE=GROUP**

<table>
<thead>
<tr>
<th>Important</th>
</tr>
</thead>
<tbody>
<tr>
<td>In the examples of the GDFN DELETE keyword above, note that you must specify what type of delete function you want GDFN to perform: delete only individual exceptions that you list (DELETE=EXCEPTION and LIST=exception,...), or the entire group (DELETE=GROUP).</td>
</tr>
</tbody>
</table>

**LEXC** Sets order of exceptions for exception analysis sampling.

**Type:** Immediate

The LEXC command displays the order in which OMEGAMON executes exceptions. To change the sequence of exception messages displayed by the XIMS command, you can dynamically reorder the execution sequence.

The following figure shows a partial LEXC display. To change the order of the exceptions, type over an exception name or its number.

```
LEXC
:  INAC = 1  DISP = 2  DNRS = 3  TNRS = 4  DRDY = 5  TRDY = 6
:  IORC = 7  SPAH = 8  QKH = 9  SMGH = 10  LGMH = 11  RDHS = 12
:  DHLH = 13  MFSH = 14  TMFH = 15  ACBH = 16  OSBL = 17  SAPW = 18
:  ITWH = 19  SDFP = 20  CROL = 21  ARSP = 22  CSVC = 23  PSVC = 24
:   .   .   .   .   .   .
:   .   .   .   .   .   .
:   .   .   .   .   .   .
```

You can also type in exceptions with new order numbers on the command line following the LEXC command. For example:

**LEXC DNRS=1 WSHI=2 DRDY=3**
MSGD | Defines messages for message exception analysis.

**Type:** Immediate

**Format:**

```
MSGD [ADD|DELETE|LIST cccccccc]
[OPTION=BUFSHOW|BUFFER|NOSHOW]
```

- **(blank)** Lists all defined messages.
- **ADD** Adds a specified IMS message.
- **DELETE** Deletes a specified message.
- **LIST** Lists a specified message or series of messages.
- **cccccccc** Specifies the IMS message ID. These characters match the first characters of the IMS message text. There are no assumed blanks appended to any message ID of less than eight characters. You may put the message ID in quotes to add blanks to the message text.
  
  You may use an asterisk (*) as a wildcard character. There is no need to use the asterisk (*) to delineate the end of a message ID string, because this is assumed with an unquoted string.

  ```
  MSGD ADD DFS206
  MSGD ADD DFS206*
  ```

  Both of the above translate to the same compare string.

  ```
  MSGD ADD DFS206
  MSGD ADD ‘DFS206 ‘
  ```

  In the above example, the first entry matches the IMS message ID DFS206 and DFS2061I. The second entry matches only the IMS message ID DFS206.
The OPTION= parameter only applies to the ADD keyword. Valid options are:

**BUFSHOW**
Displays the message through exception analysis and retains it for display by the PMSG immediate command. BUFSHOW is the default.

**BUFFER**
Retains the message for display by the PMSG immediate command.

**NOSHOW**
Does not display the message in exception analysis or with the PMSG command. However, if the exception occurs, you can set it to start the exception logging facility (XLF). See “Automating and Logging Features” on page 331 for an explanation of how to log exceptions.

You must issue the ICNS command to activate message exception analysis before you can define messages with MSGD. See “IMS messages and IMS-related MVS messages (ICNS)” on page 301 for more information about the ICNS command.

MSGD assigns a number (Mnnn) that becomes an exception name. You can define up to 200 messages. The following figure shows output from the MSGD command when it is entered without any keywords:

```
MSGD
+ M002 DFS428    OPTION=BUFSHOW
+ M003 DFS730I   OPTION=BUFSHOW
+ M004 DFS236    OPTION=BUFSHOW
+ M005 DFS286I   OPTION=BUFFER
+ M006 DFS551I   OPTION=BUFSHOW
```

You can also add MVS messages that are generated by an IMS region, as illustrated in the following figure:

```
MSGD ADD ICE074 OPTION=BUFSHOW
+ M007 ICE074    OPTION=BUFSHOW
MESSAGE HAS BEEN ADDED
```

**Note:** Messages defined with MSGD exist as exceptions during the current OMEGAMON session only, unless you save them in a profile with the PPRF command.
The PMSG command displays IMS messages that were defined with the MSGD command. The message must be defined with MSGD ADD and the OPTION=BUFSHOW or OPTION=BUFFER parameter.

Type the XIMS immediate command above PMSG to invoke the IMS message exception function and to keep current IMS messages displaying under PMSG.

<table>
<thead>
<tr>
<th>PMSG</th>
<th>Displays IMS messages that have been defined to exception analysis and have tripped.</th>
</tr>
</thead>
</table>

Type: Immediate

The PMSGnn command displays IMS messages that were defined with the MSGD command. The message must be defined with MSGD ADD and the OPTION=BUFSHOW or OPTION=BUFFER parameter.

Type the XIMS immediate command above PMSGnn to invoke the IMS message exception function and to keep current IMS messages displaying under PMSGnn.

<table>
<thead>
<tr>
<th>PMSGnn</th>
<th>Displays IMS messages that have been defined to exception analysis and have tripped. If more than the specified number of priority messages exist, the command displays the most recent (nn) ones.</th>
</tr>
</thead>
</table>

Type: Immediate

The PMSGnn command displays IMS messages that were defined with the MSGD command. The message must be defined with MSGD ADD and the OPTION=BUFSHOW or OPTION=BUFFER parameter.

Type the XIMS immediate command above PMSGnn to invoke the IMS message exception function and to keep current IMS messages displaying under PMSGnn.

<table>
<thead>
<tr>
<th>XACB</th>
<th>Sets exception thresholds and attributes.</th>
</tr>
</thead>
</table>

Type: Immediate

Format:

```
{ALL|GROUP=cc|LIST=cccc} FORCE
{(VERBOSE|TERSE) FORCE
```

(Blank) Lists all exceptions with their current settings in columnar display. This is the default.

ALL Lists all exceptions in invocation sequence.

GROUP Specifies the 2-character identifier (cc) to be associated with the exception.
**LIST**

Specifies the exceptions (cccc) to display. The exceptions requested are listed with their current settings, which you can modify.

**VERBOSE**

Multi-line display for each defined exception. Displays all exception parameters.

**TERSE**

Single line display for each defined exception. Displays the following exception parameters:

- exception name
- threshold value
- display value
- exception state
- bell state

**FORCE**

This keyword causes the exception analysis routine to become active. If the exception trips based upon the current threshold settings, an exception message will appear as if the exception analysis command were actually executing. To display a sample message text of a specific exception, the STATE=TEST must be specified along with the FORCE keyword.

XACB displays parameters in the following format:

```
XACB LIST=cccc
 : cccc
  + DISPLAY Parameters: THRESHOLD Parameters: XLF Parameters:
  : State= Threshold= Auto=
  : Group= Display= Log=
  : Bell= Attribute= Limit=nn (n)
  + BOX Parameters: CYCLE Parameters: Repeat=
    : Boxchar=' ' ExNcyc=n Persist=nn
    : Boxclr= Stop=n (m)
    : Boxattr= Cumulative=n Sc=
```

To change an option value, type over the displayed value and press ENTER. The next time OMEGAMON invokes this exception, it uses these new characteristics.

**DISPLAY parameters**

**STATE**

One of the following:

- **NDSP**
  
  Allows you to suppress the display of exceptions that you do not need to act on at this time. OMEGAMON treats the exception as ON, but the exception does not appear. Instead, it can be logged to the XLFLOG, or can trigger an automatic screen space routine when it occurs.

- **ON**
  
  Invokes this exception during the current OMEGAMON session.
OFF Does not invoke this exception during the current OMEGamon session.

TEST This parameter is used primarily for the purpose of training or demonstration. In conjunction with the FORCE keyword, it causes a sample exception message to be displayed. When the TEST state forces a message to display, a T appears in column 2 of the message lines under XIMS.

Note: The zoom function is not available for exceptions in test mode.

GROUP Specifies the 2-character group identifier. Candle Corporation ships the product with groups predefined. Use the GDFN command to display existing groups or to define new groups. The settings for groups override the settings for individual exceptions.

BELL Specifies whether the audible alarm on the terminal sounds when this exception occurs. The BELL must be activated with the OPTN BELL=ON command.

THRESHOLD parameters

THRESHOLD Exception threshold. For exceptions that are just either ON or OFF and do not have a numeric threshold, this entry is blank.

DISPLAY Sets the exception display color or intensity. Can be set to the variables Clr1 through Clr7. The variable values are defined with the .SCC command. They are associated with the colors listed below, and follow the same order. Optionally, this value can be HI or LO on four- or non-color terminals and one of the following on terminals that support the extended data stream.

- **RE** Sets the exception text red.
- **BL** Sets the exception text blue.
- **YE** Sets the exception text yellow.
- **PI** Sets the exception text pink.
- **GR** Sets the exception text green.
- **TU** Sets the exception text turquoise.
- **WH** Sets the exception text white.
- **NONE** Specifies the hardware default colors.

The presentation of the intensity or color level on your terminal is determined by the type of terminal and the settings of the SCC keywords.

ATTRIBUTE Sets an additional highlight attribute for the box.

- **BLINK** Turns on blinking for an exception.
- **RVRS** Displays message in reverse video.
- **UNDR** Underscores a message.
- **NONE** Uses the default extended highlight attributes.

These attributes take effect only in modes other than ISPF and cross memory/cross system on terminals that support the extended data stream.
XLF parameters

See “Automating and Logging Features” on page 331 for an explanation of the exception logging facility (XLF) and automatic screen facility (ASF) parameters.

BOX parameters

BOXCHAR  Specifies box character, enclosed in single quotes. The default is a plus sign (+). Do not use a single quote as a box character, since it is the delimiter. Enter NOBOX without quotes to turn off boxing for an exception. If Boxchar=NOBOX, then the BOXCLR and BOXATTR parameters have no effect.

BOXCLR  Sets the color or intensity of the exception box.

The Boxclr= keyword settings follow the same format as the Threshold parameter, Display=.

BOXATTR  For seven-color terminals modes other than ISPF or cross-memory, sets an additional highlight attribute for the box.

    BLINK  Turns on blinking for an exception.
    RVRS   Displays message in reverse video.
    UNDR   Underscores a message.
    NONE   Specifies the hardware default attributes.

If you set Boxchar=NOBOX, then the Boxclr= and Boxattr= parameters have no effect.

If you do not set color and highlighting attributes for the box, OMEGAMON uses those that you set for the exception.
CYCLE parameters

EXNCYC Sets the frequency for checking the exception at every n OMEGAMON cycles. If this parameter is set to 0 or 1, it will be tested every OMEGAMON cycle. If it is set to a higher number, it will only be tested each time that number of cycles elapses. The default setting for EXNCYC is 0.

This parameter is provided so that you can tailor high overhead exceptions for your own environment. You can avoid using CPU time to test them every cycle. For example, if you have many devices in the class being examined by an exception (such as DASD or tape), you may want to set this parameter for corresponding exceptions in the hardware group.

When an exception that is not tested every cycle trips, the exception message appears on the screen as usual. In the following cycles during which it is not scheduled for testing, the exception message redisplays on the screen below the primary exception analysis display.

See also the .NXE immediate command that controls the display of frequency-limited exceptions.

STOP=n (m) Sets a limit on the number of times an exception is allowed to trip. After the exception trips n times, the exception will not be tested or displayed during the current OMEGAMON session, unless the user resets this parameter. The (m) value, which is informational only, indicates the number of times the exception has already tripped since the user last reset the Stop parameter. The default value for Stop is 0, which means that there is no limit to how many times the exception can be tested and displayed.

CUMULATIVE Indicates how many times the exception has tripped during the current OMEGAMON session. Users may not alter this value.

The following is a typical XACB display:

```
XACB
: DSTR
+  DISPLAY Parameters: THRESHOLD Parameters: XLF Parameters:
   State=ON  Threshold=N/A  Auto=OFF
   Group=IM  Display=CLR1  Log=OFF
   Bell=OFF  Attribute=REVERSE  Limit=0
+  BOX Parameters: CYCLE Parameters: Repeat=NO
   Boxchar='#'  ExNcyc=0  Persist=0
   Boxclr=TURQUOISE  Cumulative=0
   Boxattr=REVERSE
: DISP
+  DISPLAY Parameters: THRESHOLD Parameters: XLF Parameters:
   State=ON  Threshold=N/A  Auto=OFF
   Group=ST  Display=CLR1  Log=OFF
   Bell=OFF  Attribute=REVERSE  Limit=0
+  BOX Parameters: CYCLE Parameters: Repeat=NO
   Boxchar='#'  ExNcyc=0  Persist=0
   Boxclr=TURQUOISE  Cumulative=2
   Boxattr=REVERSE  SS=
```
The following is a partial XACB terse mode display.

```
XACB TERSE
: DNRS Threshold=N/A Display=Red State=ON Bell=ON
: WSHI Threshold=2500 Display=Pink State=ON Bell=OFF
: WSLO Threshold=300 Display=Blue State=NDSP Bell=OFF
```

### Controlling groups

The XGSW command gives you control of exceptions by group.

<table>
<thead>
<tr>
<th>XGSW</th>
<th>Sets exception group switch settings.</th>
</tr>
</thead>
</table>

The group switch command allows you to set the exception state for an entire exception group. This switch overrides the individual exception setting.

If you type in XGSW with no keywords, it displays all existing groups with their current settings. Overtype the current setting for the STATE keyword to change the setting.

**Type:** Immediate

**Format:** `XGSW GROUP=cc STATE=cccc`

- **GROUP** Any two unique alphanumeric characters (cc) to specify the group. Use this keyword to display only entries for a particular group.
- **STATE** Controls whether the exception is in any of these five states:
  - **ON** Invokes the exception group during the current session.
  - **OFF** Does NOT invoke the exception group during the current session.
  - **TEST** Forces a sample warning message, even if the exception condition is not presently occurring, for purposes of training or demonstration. (When a message has been displayed because of TEST mode, a T appears in column 2 of the message lines.)
  - **Note:** The zoom function is not available for exceptions in test mode.
  - **NDSP** Exceptions in the group are ON, but the exceptions are not displayed. Instead, they can be logged to the XLFLOG or can trigger automatic screen spaces.
  - **NULL** Specifies that the individual exception, rather than the group switch, is to maintain control. This is the default.
Resource Contention Exception Commands

Introduction

The resource contention immediate commands display exception information when resource conflicts occur.

<table>
<thead>
<tr>
<th>CONF</th>
<th>Displays IRLM or PI lock conflicts.</th>
</tr>
</thead>
</table>

Type: Immediate

The following figure displays the CONF command output.

<table>
<thead>
<tr>
<th>CONF</th>
<th>Subsys</th>
<th>Workunit</th>
<th>PSBname</th>
<th>Tx/RgID</th>
<th>Lterm ID</th>
<th>Status</th>
<th>DB/AREA</th>
<th>RBA/RBN</th>
<th>DCB</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>IMSA</td>
<td>CICS321</td>
<td>ACCNT010</td>
<td>1</td>
<td>--none--</td>
<td>UP/OWN</td>
<td>ACCNTDBA</td>
<td>0000940C</td>
<td>1</td>
</tr>
<tr>
<td>+</td>
<td>IMSA</td>
<td>CICS321</td>
<td>UPDCUST1</td>
<td>2</td>
<td>--none--</td>
<td>UP/WAT</td>
<td>ACCNTDBA</td>
<td>0000940C</td>
<td>1</td>
</tr>
<tr>
<td>+</td>
<td>IMSA</td>
<td>CICS321</td>
<td>UPDCUST1</td>
<td>3</td>
<td>--none--</td>
<td>UP/WAT</td>
<td>ACCNTDBA</td>
<td>0000940C</td>
<td>1</td>
</tr>
</tbody>
</table>

The following list describes the output fields for CONF.

- **Subsys**: Subsystem name holding the lock.
- **Workunit**: Name of job holding the lock.
- **PSBname**: Program specification block (PSB) associated with the lock.
- **Tx/RgID**: The transaction name or region number (if DBCTL is being monitored) associated with the lock.
- **Lterm ID**: Logical terminal name associated with the lock. A value of **--NONE--** in this field indicates there is no logical terminal associated with this lock.
- **Status**: Intent and status of the workunit (job) holding the lock or waiting for the resource. Valid intents are:
  - **UP**: Update intent.
  - **RD**: Read-only intent.
  - **EX**: Exclusive intent.
  - **SH**: Share intent.
  Valid statuses are:
  - **OWN**: Workunit owns this resource.
  - **WAT**: Workunit is waiting for this resource.
- **DB/AREA**: Database name or DEDB area name.
- **RBA/RBN**: Displays the relative byte address for PI locking or the relative block number for IRLM locking. Displays **n/a** if data is not available because there is more than one IRLM and the secondary IRLM is running on a different CPU.
- **DCB**: Data control block (DCB) number within the named DMB.
If data is not available because the lock was freed before analysis completed, **n/a** appears.

**Note:** You might see that a job is waiting for a resource when no owner for that resource is displayed. This is probably because the resource is locked by an IRLM running on another system.

Only those locks held by the first region displayed by the major command are shown. For example, in Figure 14 on page 202, the lock for region MPP00131 is the only one displayed. To display the locks for the other dependent regions, use RGNL.

<table>
<thead>
<tr>
<th>XFPQ</th>
<th>Displays active Fast Path DEDB resource (control interval) request conflicts for all DEDB database areas.</th>
</tr>
</thead>
</table>

**Type:** Immediate

The example below shows the results of the XFPQ command.

<table>
<thead>
<tr>
<th>XFPQ</th>
<th>Jobname</th>
<th>PSBName</th>
<th>Tx/RgID</th>
<th>DBname</th>
<th>AREAname</th>
<th>R.B.A.</th>
<th>Status</th>
<th>P.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>CICS321</td>
<td>DBFSAMP3</td>
<td>1</td>
<td>DBFSAMD3</td>
<td>CUSDB</td>
<td>0000C000</td>
<td>EX/OWNER</td>
<td>YES</td>
</tr>
<tr>
<td>+</td>
<td>CICS321</td>
<td>DBFSAMP3</td>
<td>2</td>
<td>DBFSAMD3</td>
<td>CUSDB</td>
<td>0000C000</td>
<td>EX/OWNER</td>
<td>YES</td>
</tr>
</tbody>
</table>

The following are the XFPQ fields and their meanings:

**Jobname**
Name of region holding or waiting for resource. If name is OTHR, an output thread is holding the resource (control interval).

**PSBName**
Name of the program specification block.

**Tx/RgID**
The transaction name or region number (if DBCTL is being monitored).

**DBname**
Name of the Fast Path DEDB.

**AREAname**
Name of this partition of the DEDB.

**R.B.A.**
Relative byte address; the address of the resource (control interval).

**Status**
Status of the resource request. Possible values are:
- **EX/OWNER** exclusive/owner
- **EX/WAITING** exclusive/waiting (highlighted)
- **NE/OWNER** non-exclusive/owner
- **NE/WAITING** non-exclusive/waiting (highlighted)

**P.I.**
Indicates whether the IMS lock manager knows about the resource control request.
XLTQ | Scans the IMS latch tables and searches for conflicts.

**Type:** Immediate

When one dependent region holds an IMS latch that another region wants, the second region waits until the latch is available. A sample lockout of this type appears below:

<table>
<thead>
<tr>
<th>XLTQ</th>
<th>Jobname</th>
<th>PSBname</th>
<th>Tx/RgID</th>
<th>Lterm ID</th>
<th>Status</th>
<th>Latch Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>CICS321</td>
<td>PY4BUP00</td>
<td>1</td>
<td>--none--</td>
<td>Owner</td>
<td>LOGL Logical Logger</td>
</tr>
<tr>
<td>+</td>
<td>J0</td>
<td>PY4MUP00</td>
<td>2</td>
<td>--none--</td>
<td>Waiting</td>
<td>LOGL Logical Logger</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>J3</td>
<td>DBFSAMP3</td>
<td>FPSAMP1</td>
<td>--none--</td>
<td>Owner</td>
<td>OPEN FP DEDB Open</td>
</tr>
<tr>
<td>+</td>
<td>J2</td>
<td>DBFSAMP3</td>
<td>FPSAMP1</td>
<td>--none--</td>
<td>Waiting</td>
<td>OPEN FP DEDB Open</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>J2</td>
<td>DBFSAMP3</td>
<td>FPSAMP1</td>
<td>--none--</td>
<td>Owner</td>
<td>DMAC FP DEDB CUSDB</td>
</tr>
<tr>
<td>+</td>
<td>J1</td>
<td>DBFSAMP3</td>
<td>FPSAMP1</td>
<td>--none--</td>
<td>Waiting</td>
<td>DMAC FP DEDB CUSDB</td>
</tr>
</tbody>
</table>

In this example, the region BMP01 holds the logical logger latch (LOGL), and the region J0 wants it; region J0 waits until BMP01 frees the latch.
Chapter overview

This chapter provides information about OMEGAMON commands that provide general DBCTL system information.

Chapter contents

Real Memory Information .................................................. 128
Common Storage Area Virtual Memory Analysis ....................... 129
CSA Storage Isolation Information ..................................... 131
IMS Control Block Display Commands .................................. 132
Dynamic Control Block Table Analysis ................................ 138
Dynamic Control Block Table Minor Commands ..................... 139
Display Checkpoint ID Information .................................... 141
Locate IMS Modules in Virtual Memory ................................. 142
IMS Startup Parameters and Overrides ................................. 143
IRLM Startup Parameters ................................................. 145
IMS Control Region Virtual Storage Analysis ......................... 148
General IMS Information ................................................. 150
IMS Online Change Analysis ............................................ 153
IMS Dataset Information .................................................. 155
IMS Trace Table Entries ................................................... 160
Plotting System Information .............................................. 161
Real Memory Information

Introduction

CSAR is the command to display real memory information.

<table>
<thead>
<tr>
<th>CSAR</th>
<th>Displays the users of CSA real memory by their storage protect keys.</th>
</tr>
</thead>
</table>

Type: Immediate

IMS CSA is in storage key 7.

If you run more than one DBCTL and/or IMS system at a time, remember that the real memory for key 7 reflects the CSA usage of all your IMS systems, not just the one OMEGAMON is monitoring.
Common Storage Area Virtual Memory Analysis

Introduction

There are three commands to analyze common storage area virtual memory:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSA</td>
<td>Analyzes the common storage area by subpool number and protection key.</td>
</tr>
<tr>
<td>Type</td>
<td>Immediate</td>
</tr>
</tbody>
</table>

| ECSA     | Analyzes the extended CSA by subpool number and protection key. |
| Type    | Immediate   |

| ESQA     | Analyzes the extended system queue area. |
| Type    | Immediate   |

The ESQA command displays the amount of virtual memory allocated and used for each subpool. The Ext field displays the number of extents assigned to subpools. (A GETMAIN request that MVS cannot satisfy with the storage currently allocated to a subpool causes MVS to assign another extent to the subpool; these extents are always multiples of 4K long.) The allocated blocks field displays the amount of virtual storage assigned to a subpool. The Storage Used field shows the amount of space within the storage assigned to a subpool that is in use (that is, turned over to the user in response to a GETMAIN request). The largest free block is the size of the largest chunk within a subpool that is not currently in use, except that the total line shows the largest block not allocated to a subpool.

The storage values display in megabytes + kilobytes + bytes. A megabyte (M) is 1,048,576 bytes, while a kilobyte (K) is 1024 bytes.
OMEGAMON cannot break down the usage of ESQA by address space, because the operating system does not generally save this information.

<table>
<thead>
<tr>
<th>SQA Subpl-Key</th>
<th>Ext</th>
<th>Alloc. Blks</th>
<th>Storage Used</th>
<th>Largest Free Blk</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ 226-0</td>
<td>4</td>
<td>56K</td>
<td>52K+128</td>
<td>2K</td>
</tr>
<tr>
<td>+ 239-0</td>
<td>3</td>
<td>88K</td>
<td>80K+816</td>
<td>3K+1,016</td>
</tr>
<tr>
<td>+ 245-0</td>
<td>6</td>
<td>368K</td>
<td>185K+48</td>
<td>168K+72</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ Total:</td>
<td>13</td>
<td>512K</td>
<td>317K+992</td>
<td></td>
</tr>
<tr>
<td>+ Percent of SQA:</td>
<td>100.0%</td>
<td>62.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ SQA Size:</td>
<td>512K</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ESQA Subpl-Key</th>
<th>Ext</th>
<th>Alloc. Blks</th>
<th>Storage Used</th>
<th>Largest Free Blk</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ 239-0</td>
<td>3</td>
<td>52K</td>
<td>48K+184</td>
<td>3K+736</td>
</tr>
<tr>
<td>+ 245-0</td>
<td>4</td>
<td>8M+204K</td>
<td>479K+208</td>
<td>7M+736K+592</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ Total:</td>
<td>7</td>
<td>8M+256K</td>
<td>527K+392</td>
<td></td>
</tr>
<tr>
<td>+ Percent of ESQA:</td>
<td>100.0%</td>
<td>6.2%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SQA Analyses the system queue area.

Type: Immediate

The SQA immediate command displays virtual memory usage by subpool and protect key.

The SQA command displays the amounts of virtual memory allocated and used for each subpool. Ext displays the number of extents assigned to subpools. A GETMAIN request that MVS cannot satisfy with the storage currently allocated to a subpool causes MVS to assign another extent to the subpool; these extents are always multiples of 4K long. The Allocated Blocks field displays the amount of virtual storage assigned to a subpool. The Storage Used field shows the amount of space within the storage assigned to a subpool that is in use (that is, turned over to the user in response to a GETMAIN request). The largest free block is the size of the largest chunk within a subpool that is not currently in use, except that the total line shows the largest block not allocated to a subpool.

The storage values display in megabytes + kilobytes + bytes. A megabyte (M) is 1,048,576 bytes, while a kilobyte (K) is 1024 bytes.

OMEGAMON cannot break down the usage of SQA by address space, because the operating system does not generally save this information.
CSA Storage Isolation Information

Introduction

The command that displays CSA storage isolation is CSTI.

<table>
<thead>
<tr>
<th>CSTI</th>
<th>Displays the parameters and data relevant to CSA storage isolation.</th>
</tr>
</thead>
</table>

**Type:** Immediate

For CSA storage isolation, paging is fenced in units of page-ins per second (rather than page-ins per CPU-second for address space fencing). In the example below, the fence limits are set at two and five page-ins per second; the current page-ins rate is one CSA page/second.

<table>
<thead>
<tr>
<th>CSTI</th>
<th>Common:</th>
<th>Working</th>
<th>Page-ins</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Set Size</td>
<td>/second</td>
</tr>
<tr>
<td>+</td>
<td>Maximum</td>
<td>1520K</td>
<td>5</td>
</tr>
<tr>
<td>+</td>
<td>Target</td>
<td>1500K</td>
<td>----</td>
</tr>
<tr>
<td>+</td>
<td>Actual</td>
<td>1500K</td>
<td>1</td>
</tr>
<tr>
<td>+</td>
<td>Minimum</td>
<td>200K</td>
<td>2</td>
</tr>
</tbody>
</table>

IPS parameters are specified in 4K blocks, rather than in units of 4K, as the example above shows. The common area consists of CSA + PLPA.

When you specify storage isolation, bear in mind that all cross-memory address spaces use page-ins per second, as opposed to page-ins per execution second.

If you use the DLISAS address space, the control region is not a cross-memory address space. Use execution seconds when you calculate storage isolation for the control region.
IMS Control Block Display Commands

Introduction

The control block display commands are listed and described on the following pages.

<table>
<thead>
<tr>
<th>DUMP</th>
<th>Groups together a set of minor commands which dump various IMS control blocks.</th>
</tr>
</thead>
</table>

Type: Major

By default, each minor produces a display in hexadecimal format, with each line being 16 hex bytes followed by its character equivalents.

Note: Each minor also accepts either X or C in the label field, if you want the display to be all hexadecimal or all character.

<table>
<thead>
<tr>
<th>BCPT</th>
<th>Dumps the checkpoint ID table.</th>
</tr>
</thead>
</table>

Type: Minor of DUMP
Format: BCPT

<table>
<thead>
<tr>
<th>BFSP</th>
<th>Dumps the VSAM buffer pool prefix.</th>
</tr>
</thead>
</table>

Type: Minor of DUMP
Format: BFSP

<table>
<thead>
<tr>
<th>BFUS</th>
<th>Displays the VSAM buffer statistics block.</th>
</tr>
</thead>
</table>

Type: Minor of DUMP
Format: BFUSnn
### IMS Control Block Display Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BSPH</strong></td>
<td>Displays the VSAM buffer subpool header.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DUMP  
**Format:** `BSPHnn`  
`nn` Specifies the control block to display. This number corresponds to the VSAM subpool number the DBVS immediate command displays.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DDIR</strong></td>
<td>Dumps the DMB directory entry.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DUMP  
**Format:** `DDIRcccccccc`  
`cccccccc` Specifies a number, `n`, to dump the `n`th control block entry or the name of a DMB.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DMAC</strong></td>
<td>Dumps the DEDB area control.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DUMP  
**Format:** `DMACcccccccc`  
`cccccccc` Specifies a number, `n`, to dump the `n`th control block entry or the name of a DMB.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DMCB</strong></td>
<td>Dumps the DEDB master control.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DUMP  
**Format:** `DMCBcccccccc`  
`cccccccc` Specifies a number, `n`, to dump the `n`th control block entry or the name of a DEDB.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DMHR</strong></td>
<td>Dumps Fast Path buffer headers.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DUMP  
**Format:** `DMHRnn`  
`nn` Specifies the control block to dump.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EPST</td>
<td>Dumps extended PST.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of DUMP</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><code>EPST cccccccc</code></td>
</tr>
<tr>
<td></td>
<td><code>cccccccc</code> Specifies a number, n, to dump the nth control block entry or the job name of a region.</td>
</tr>
<tr>
<td>ESCD</td>
<td>Dumps extended SCD.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of DUMP</td>
</tr>
<tr>
<td>IBPL</td>
<td>Dumps the ISAM/OSAM buffer pool prefix.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of DUMP</td>
</tr>
<tr>
<td>IBPR</td>
<td>Dumps the ISAM/OSAM buffer subpool buffer prefix.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of DUMP</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td><code>IBPRnn</code></td>
</tr>
<tr>
<td></td>
<td><code>nn</code> Specifies the subpool to dump.</td>
</tr>
<tr>
<td>IPB</td>
<td>Dumps the initialization parameter block.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of DUMP</td>
</tr>
</tbody>
</table>
### ISBP

Dumps the ISAM/OSAM buffer subpool header.

**Type:** Minor of DUMP  
**Format:** `ISBPnn`  
`nn` Specifies the control block to dump. This number corresponds to the ISAM/OSAM subpool number the DBOS immediate command displays.

### IZIB

Dumps the zone initial block.

**Type:** Minor of DUMP  
**Format:** `IZIBnn`  
`nn` Specifies the control block to dump.

### MLCD

Dumps the DC monitor log control directory.

**Type:** Minor of DUMP  
**Format:**

### PAB

Dumps the parameter anchor block.

**Type:** Minor of DUMP  
**Format:**

### PDIR

Dumps the PSB directory entry.

**Type:** Minor of DUMP  
**Format:** `PDIRcccccccccccc`  
`cccccccc` Specifies a number, n, to dump the nth control block entry or the name of a PSB.
**PST**

Dumps the partition specification table.

**Type:** Minor of DUMP  
**Format:** `PSTcccccccc`  
`cccccccc` Specifies a number, n, to dump the nth control block entry or the job name of the region.

**RSIN**

Dumps the checkpoint/restart log record buffer.

**Type:** Minor of DUMP  
**Format:** `RSINnn`  
`nn` Specifies the control block to dump.

**SAP**

Displays the save area set prefix.

**Type:** Minor of DUMP  
**Format:** `SAPnnn`  
`nnn` Specifies the control block to dump.

**SCD**

Dumps the system contents directory.

**Type:** Minor of DUMP  
**SLCD**

Dumps the system log control directory.

**Type:** Minor of DUMP
The following figure is a sample of the DUMP major command and some of its minors.
Dynamic Control Block Table Analysis

Introduction

OMEGAMON has three major commands (CBTA, CBTL, CBTP) which, along with their minors, provide data about dynamic storage area control block tables (IPAGES).

For a list of the current dynamic storage area control block tables, refer to the IMS.GENLIBB member DFSCBTS.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBTA</td>
<td>Selects all dynamic storage area control block tables.</td>
</tr>
<tr>
<td>CBTL</td>
<td>Selects listed dynamic storage area control block tables.</td>
</tr>
<tr>
<td>CBTP/n</td>
<td>Selects all dynamic storage area control block tables matching a pattern.</td>
</tr>
</tbody>
</table>

**Type:** Major

IPGA is an alias for the CBTA command.

**Type:** Major

IPGL is an alias for the CBTL command.

**Type:** Major

**Format:** CBTP/n

/n Specifies the pattern.

The .SPT command sets the pattern. See the .SPT command for information about setting patterns. If CBTP does not find a /n value, it uses the first pattern supplied with the last .SPT command.

IPGP is an alias for the CPTP command.
Dynamic Control Block Table Minor Commands

Introduction

The minor commands of CBTA, CBTL, and CBTP provide additional information about IPAGES.

The following figure shows an example of using the dynamic control block table minor commands.

<table>
<thead>
<tr>
<th>CBTA</th>
<th>IOSB</th>
<th>GIOB</th>
<th>OSWA</th>
<th>GOWA</th>
<th>PST</th>
<th>DPST</th>
<th>SAP</th>
<th>GQMW</th>
</tr>
</thead>
<tbody>
<tr>
<td>blkl</td>
<td>576</td>
<td>1</td>
<td>1024</td>
<td>1024</td>
<td>2312</td>
<td>2312</td>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>blkn</td>
<td>21</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>12</td>
<td>2</td>
<td>94</td>
<td>30</td>
</tr>
<tr>
<td>blku</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>1</td>
<td>76</td>
<td>16</td>
</tr>
<tr>
<td>curr</td>
<td>12288</td>
<td>0</td>
<td>8192</td>
<td>0</td>
<td>49152</td>
<td>8192</td>
<td>12288</td>
<td>8192</td>
</tr>
<tr>
<td>fres</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>gets</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ipfr</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ipln</td>
<td>12288</td>
<td>0</td>
<td>8192</td>
<td>0</td>
<td>49152</td>
<td>8192</td>
<td>12288</td>
<td>8192</td>
</tr>
<tr>
<td>locp</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
<td>CSA/CTL</td>
</tr>
<tr>
<td>maxs</td>
<td>12288</td>
<td>0</td>
<td>8192</td>
<td>0</td>
<td>49152</td>
<td>8192</td>
<td>12288</td>
<td>8192</td>
</tr>
<tr>
<td>npgs</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>12</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>sbpl</td>
<td>228</td>
<td>228</td>
<td>228</td>
<td>228</td>
<td>231</td>
<td>231</td>
<td>231</td>
<td>231</td>
</tr>
</tbody>
</table>

**BLKL**

Displays length of the control block for this IPAGE type.

**Type:** Minor of dynamic control block table majors

**BLKN**

Displays total number of control blocks for this IPAGE type.

**Type:** Minor of dynamic control block table majors

**BLKU**

Displays total number of control blocks used for this IPAGE type.

**Type:** Minor of dynamic control block table majors

**CURR**

Displays CURRENT STORAGE usage for this IPAGE type.

**Type:** Minor of dynamic control block table majors

**FRES**

Displays total number of FREEMAINs for this IPAGE type.

**Type:** Minor of dynamic control block table majors
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETS</td>
<td>Displays total number of GETMAINs for this IPAGE type.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>IPFR</td>
<td>Displays total number of free IPAGES.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>IPLN</td>
<td>Displays the IPAGE length for this IPAGE type.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>LOCP</td>
<td>Displays location of the control block.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>MAXS</td>
<td>Displays MAXIMUM STORAGE usage for this IPAGE type.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>NPGS</td>
<td>Displays TOTAL NUMBER of IPAGEs for this IPAGE type.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
<tr>
<td>SBPL</td>
<td>Displays MVS SUBPOOL for this IPAGE type.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of dynamic control block table majors</td>
</tr>
</tbody>
</table>

The location is CSA or PVT (private) and CTL (control region) or DLS (DLISAS region).
Display Checkpoint ID Information

Introduction

The command to display checkpoint ID information is CKPT.

<table>
<thead>
<tr>
<th>CKPT</th>
<th>Displays general IMS checkpoint information.</th>
</tr>
</thead>
</table>

**Type:** The CKPT major command displays the current checkpoint ID and the IMS log volume serial numbers.

The following figure shows an example of CKPT output.

```
CKPT Current Checkpoint id = 97161/145022
  Block Number = 68  Offset = 0
+ Latest DUMPQ/SNAPQ Checkpoint id = 97161/095321
  Block Number = 8  Offset = 0
```
Locate IMS Modules in Virtual Memory

Introduction

The command to locate IMS modules in virtual memory is FCDE.

| FCDE | Locates IMS modules in virtual memory. |

**Type:** Immediate

The FCDE immediate command accepts an 8-character IMS module name as input. It then searches the IMS jobpack queue and the PLPA directory to locate the contents directory entry (CDE) for the module.

OMEGAMON also searches the jobpack queue for all IMS system control address spaces (including DLISAS, DBRC, and IRLM).

If OMEGAMON finds the module, it displays information about it, as shown in the following figure. This can also be useful in locating various IMS control blocks, because IMS artificially creates CDEs for many of its important work areas.

```
FCDE  DFSMVRC0
+      Module located in the CTL Region Jobpack Queue
+      Entry Point = 00115E78  Length = 000188  Use Count = 1
+      Module located in the DLS Region Jobpack Queue
+      Entry Point = 00115E78  Length = 000188  Use Count = 1
+      Module located in the DBRC Region Jobpack Queue
      Entry Point = 00115E78  Length = 000188  Use Count = 1
```
IMS Startup Parameters and Overrides

Introduction

Use the FCPB command to display the startup parameters for an IMS control region.

| FCPB  | Displays IMS control region startup parameters and indicates if the displayed parameter value is a user-initiated override or has been set by IMS initialization processing. |

Type: Immediate

You can override IMS control region options with the PARM= keyword of the EXEC card as well as with the DFSPRRGx member, found using a normal STEPLIB search. The options they specify merge according to rules in the IMS/VS System Programmer’s Reference Manual.

Various symbols might appear next to the displayed value. Their meanings are as follows:

(Blank) IMS used the value specified in the IMS SYSGEN.

* IMS used the value specified in the startup JCL or the DFSPRRGx parm member.

** IMS determined that the value specified in the JCL, in DFSPRRGx, or in the SYSGEN was inadequate and therefore supplied a new value.

|| The specified parameter is obsolete.
The following figure shows a sample display.

When you are monitoring IMS, the FCPB command adds a new section to its display entitled “Expandable Storage Pool Upper Limits in bytes.” Within this section, the upper limit for each fixed pool is displayed. The G suffix indicates gigabytes.
IRLM Startup Parameters

Introduction
The command RLMO displays the startup options. RLMX displays IRLM contention and coupling facility information.

<table>
<thead>
<tr>
<th>RLMO</th>
<th>Displays IRLM startup options.</th>
</tr>
</thead>
</table>

**Type:** Immediate

<table>
<thead>
<tr>
<th>RLMO</th>
<th>IRLM options</th>
<th>IRLMN=</th>
<th>KRLM</th>
<th>IRLMID=</th>
<th>1</th>
<th>COMCYCL=</th>
<th>20</th>
<th>MAXCSA=</th>
<th>1 (1M)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEADLOK= ‘15,4’</td>
<td>SCOPE=</td>
<td>LOCAL</td>
<td>RULES=</td>
<td>COMPAT</td>
<td>PC=</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>APPLS= KRLM1, PRLM1</td>
<td>APPL2=</td>
<td>KRLM1, TRLM1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>APPL3= KRLM1, TRLM1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>----- Appl options</td>
<td>SCOPE=</td>
<td>LOCAL</td>
<td>RULES=</td>
<td>COMPAT</td>
<td>PC=</td>
<td>YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RH= ON</td>
<td>PTB=</td>
<td>OFF</td>
<td>INTERNAL=</td>
<td>OFF</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>----- IRLM Trace status</td>
<td>CURRENT</td>
<td>11392</td>
<td>MAXCSA=</td>
<td>1024K (1.11% of MAXCSA used)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The fields displayed by RLMO are IRLM release-dependent. Unless indicated below, the RLMO display fields represent the parameter values specified in the IRLM startup procedure.

**Note:** The parameters displayed on your panel may be different from this example, depending on which version of IRLM your site is using.

- **IRLMNM=** Specifies the IRLM subsystem name.
- **IRLMID=** Displays the ID= parameter.
- **COMCYCL=** Displays the COMCYCL= parameter.
- **MAXCSA=** Displays the MAXCSA= parameter and the actual storage size.
- **CURRENT=** Displays the current CSA usage.
- **DEADLOK=** Displays the DEADLOK= parameter.
- **SCOPE=** Displays the SCOPE= parameter (GLOBAL or LOCAL).
- **RULES=** Displays the RULES= parameter (COMPAT or AVAIL).
- **PC=** Displays the PC= parameter (YES or NO).
- **APPLS=** Displays the APPLS= parameter.
- **APPL2=** Displays the APPL2= parameter.
- **APPL3=** Displays the APPL3= parameter.
- **RH=** Displays if request handler traces active or inactive (ON or OFF).
This command displays statistical information concerning IRLM real and false contention.

Real contention occurs when two PSBs attempt to access the same database block at the same time. False contention occurs when there is a lock synonym, but no real contention.

The Real Contention Rate is the actual lock rate, per second, during this interval. False Contention Rate is the lock synonym rate, per second, during this interval.

In an MVS 5.1, 5.2 or OS/390 environment, you may define your IRLM to participate in a “Data Sharing group”. The group name you defined is also displayed plus all of the participants in the Data Sharing group.

<table>
<thead>
<tr>
<th>CF Name</th>
<th>Coupling Facility name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job Name</td>
<td>The MVS started task or job name of the participating IRLM subsystem.</td>
</tr>
<tr>
<td>System ID</td>
<td>The MVS system ID on which the IRLM subsystem is running.</td>
</tr>
<tr>
<td>Status</td>
<td>The current Coupling Facility status.</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>The IRLM subsystem is actively connected to the Coupling Facility and is eligible for N-Way Datasharing.</td>
</tr>
<tr>
<td>CREATED</td>
<td>The IRLM subsystem is defined to the Coupling Facility but is not yet active. This should be a transient state.</td>
</tr>
<tr>
<td>FAILED</td>
<td>The IRLM subsystem has failed to connect to the Coupling Facility. See your MVS System Console error messages.</td>
</tr>
<tr>
<td>QUIESCED</td>
<td>The IRLM subsystem is removing itself from the Coupling Facility. This is a transient state.</td>
</tr>
</tbody>
</table>
IRLM Startup Parameters

IRLMX

Lock Structure
Name . . . . . . . . . . : IRLMLT1
Size . . . . . . . . . . : 8388608
Record List Entries (RLEs)
Used . . . . . . . . . . : 871
In Lock Structure . . . : 44719
Percentage Used . . . . : 1.95%
Totals:
Real Contention Granted . : 2118
Real Contention Rate . . : 5.6/s
Global False Contention . : 721
False Contention Rate . . : .67/s

N-way Datasharing Subsystems for Group IRLMD1

<table>
<thead>
<tr>
<th>CF Name</th>
<th>Jobname</th>
<th>System ID</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLMD1$$</td>
<td>IMS510AI</td>
<td>SP11</td>
<td>Active</td>
</tr>
</tbody>
</table>
**IMS Control Region Virtual Storage Analysis**

**Introduction**

The VMEM command tracks virtual storage use.

<table>
<thead>
<tr>
<th>VMEM</th>
<th>Tracks virtual storage use and warns when shortages develop within the IMS control region.</th>
</tr>
</thead>
</table>

**Type:** Immediate

Exception analysis monitors a variety of virtual storage areas for exceptional conditions, based on thresholds you set. With the VMEM command, you can identify storage shortages as well as excesses. To trim these excesses back, you can readjust the position of IEALIMIT within the control region. The following figure shows the position of IEALIMIT within the control region. The following figure shows a sample VMEM display.

<table>
<thead>
<tr>
<th>VMEM</th>
<th>IMS Virtual Storage Constraint Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td></td>
</tr>
<tr>
<td>+</td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>Control Region : 5836k 5796k 3700k 2288k 2136k 2136k</td>
</tr>
<tr>
<td>+</td>
<td>DBRC Region : 7224k 7220k 4892k 2356k 2332k 2332k</td>
</tr>
<tr>
<td>+</td>
<td>DLS Region : 7428k 7424k 3868k 3592k 3560k 3560k</td>
</tr>
<tr>
<td>+</td>
<td>IRLM Region : 7896k 7884k 5924k 1972k 1972k 1972k</td>
</tr>
</tbody>
</table>

The following figure shows a diagram of IMSCTL region virtual storage areas.

**FIGURE 3. IMSCTL region virtual storage areas**

1. LSQA can acquire free space below the IEALIMIT line.
2. The control region private area can acquire space up to the IEALIMIT line.
OMEGAMON monitors the following virtual storage quantities by request or dynamically through exception analysis:

- total free storage available for LSQA
- largest free block available for LSQA
- amount of free storage above the IEALIMIT for LSQA (assured free LSQA)
- total free storage available for IMSCTL
- largest free block available for IMSCTL
- size of free block at top of IMSCTL region

OMEGAMON only analyzes unallocated free areas, not free areas within allocated storage. It excludes these small areas, which result from fragmentation.
General IMS Information

Introduction

ISYS is the command to display information about the status of the IMS system.

<table>
<thead>
<tr>
<th>ISYS</th>
<th>Displays general information about the state of the IMS system.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** There are two forms of the ISYS command as follows:

- **ISYS**
  - Displays the following general information about the state of the IMS system.
  - number of active MPPs
  - number of active BMPs
  - number of applications scheduled since the last warm start
  - number of transactions in the input queue
  - message enqueue and dequeue rates
  - OSAM I/Os issued by the control region (for example, OLDS I/O)
  - number of database OSAM I/Os performed in the DLS address space

- **SISYS**
  - Displays a short form of the ISYS display.

Examples

How ISYS displays information depends on the version of IMS you have installed and whether or not you have Remote Site Recovery installed.

**IMS without Remote Site Recovery**

If you have IMS and do not have Remote Site Recovery installed, the system displays the following:

**FIGURE 4. Information display for IMS without Remote Site Recovery**

```
ISYS  IMS Version 7.1.0                  Subsystem ID = ‘I51C’
+    MVS/ESA -- SP5.1.0                   IRLM Release  6
+    IMS Restart date = 97.101            IMS Restart time = 04:58:08
+    Checkpoints taken = 1                Current Checkpoint id = 97101//045808
+    MPPs active = 3                      BMPs active = 0
+    Applications scheduled = 0           Transactions queued = 0
+    Msg Enqueue rate = .00/sec           Msg Dequeue rate = .00/sec
+    System Den OSAM I/O’s = 2605          DLS OSAM I/O count = 0

>>> Remote Site Recovery not installed <<<
```
IMS with Remote Site Recovery

If you have IMS with Remote Site Recovery installed, the system displays the following:

FIGURE 5. Information display for IMS with Remote Site Recovery

<table>
<thead>
<tr>
<th>ISYS</th>
<th>IMS Version 7.1.0</th>
<th>Subsystem ID = 'I51R'</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>MVS/ESA -- SP6.0.5</td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>IMS Restart date = 98.327</td>
<td>IMS Restart time = 06:08:47</td>
</tr>
<tr>
<td>+</td>
<td>Checkpoints taken = 1</td>
<td>Current Checkpoint id = 98327/060847</td>
</tr>
<tr>
<td>+</td>
<td>MPPs active = 2</td>
<td>BMPs active = 0</td>
</tr>
<tr>
<td>+</td>
<td>Applications scheduled = 0</td>
<td>Transactions queued = 0</td>
</tr>
<tr>
<td>+</td>
<td>Msg Enqueue rate = .00/sec</td>
<td>Msg Dequeue rate = .00/sec</td>
</tr>
<tr>
<td>+</td>
<td>System Dsn OSAM I/O's = 1467</td>
<td>DLS OSAM I/O count = 0</td>
</tr>
<tr>
<td>+</td>
<td>RSR Type = ACTIVE</td>
<td>Global Service Group = IMS51RGS</td>
</tr>
<tr>
<td>+</td>
<td>Service Group = I51R</td>
<td>Transport Mgr SSID = ELXR</td>
</tr>
<tr>
<td>+</td>
<td>Readiness Level = Recovery</td>
<td>VTAM Connection = Active</td>
</tr>
<tr>
<td>+</td>
<td>Transport Manager = Active</td>
<td></td>
</tr>
</tbody>
</table>

MSYS Displays general information about the state of the MVS system.

Type: Immediate
Format: cMSYS Enter the S argument in the label field to see a short form of the MSYS display.

The MSYS command displays the amount of memory used by all system control address spaces and the number of active CPUs. The following figure is an example of the MSYS command display.

FIGURE 6. Typical MSYS system information display

<table>
<thead>
<tr>
<th>MSYS</th>
<th>System CPU usage = 65.12%</th>
<th>System SIO rate = 13.45/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>IMS CPU usage = 34.45%</td>
<td>IMS SIO rate = 7.35/sec</td>
</tr>
<tr>
<td>+</td>
<td>Average IMS CPU = 33.49%</td>
<td>IMS SIO average = 5.55/sec</td>
</tr>
<tr>
<td>+</td>
<td>Number of active CPUs = 1</td>
<td>System ID = SYSG</td>
</tr>
<tr>
<td>+  --Virtual Storage--</td>
<td>---------Working Set-------------</td>
<td></td>
</tr>
<tr>
<td>+    Below 16m Above 16m Real Expanded Total WKST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+    Control Region : 1408K 11316K 1492K N/A 1492K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+    DBRC Region : 564K 8832K 80K N/A 80K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+    DLS Region : 708K 8672K 120K N/A 120K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+    IRLM Region : 272K 9016K 296K N/A 296K</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: In this example, N/A indicates that expanded storage is not available on this CPU.
The fields in the MSYS display are:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>System CPU usage</strong></td>
<td>Displays total CPU utilization for the entire MVS system. OMEGAMON calculates system CPU usage over the last SRM interval.</td>
</tr>
<tr>
<td><strong>IMS CPU usage</strong></td>
<td>Displays the amount of CPU time that the IMS control region and its dependents have expended, which includes both TCB and SRB times. OMEGAMON calculates IMS CPU usage over the last OMEGAMON cycle, which is a shorter interval than the SRM interval. The CPU figures are percentages of the CEC, and therefore vary between 0 and 100, rather than between 0 and (100 * (number of online CPUs)).</td>
</tr>
<tr>
<td><strong>Average IMS CPU</strong></td>
<td>Displays the average amount of CPU that the IMS control region and its dependents have expended since this OMEGAMON session was started.</td>
</tr>
<tr>
<td><strong>Number of active CPUs</strong></td>
<td>Displays the number of active CPUs on the processor.</td>
</tr>
<tr>
<td><strong>System SIO rate</strong></td>
<td>Displays the SIO rate across all of MVS.</td>
</tr>
<tr>
<td><strong>IMS SIO rate</strong></td>
<td>Displays the SIO rate within all regions.</td>
</tr>
<tr>
<td><strong>IMS SIO average</strong></td>
<td>Displays the average SIO rate within all regions since this OMEGAMON session started.</td>
</tr>
<tr>
<td><strong>System ID</strong></td>
<td>Displays the SMF ID of the MVS system on which you are executing. This information is in SYS1.PARMLIB(SMFPRMnn).</td>
</tr>
<tr>
<td><strong>Virtual Storage</strong></td>
<td>Displays the amount of virtual pages IMS has currently GETMAINed. (This number is suitable as an estimate for the size of the REGION= parameter on the MVS EXEC JCL statement.)</td>
</tr>
<tr>
<td><strong>Working Set</strong></td>
<td>Displays real and expanded memory usage, and the current working set size.</td>
</tr>
</tbody>
</table>
IMS Online Change Analysis

Introduction

The command to analyze online change operations is OCHG.

<table>
<thead>
<tr>
<th>OCHG</th>
<th>Analyzes online change operations.</th>
</tr>
</thead>
</table>

Type: Immediate

The following is an example of the OCHG command display.

```
OCHG Online change is not in progress
+ Current online change id: 3 Current nucleus suffix: 0
+ Modify work area address: 00000000 Modstat work area address: 00118298
+ MODBLKSA is active ACBLIBA is active
+ DATE TIME
+ LAST MODIFY PREPARE 97157 09:59:068
+ LAST MODIFY COMMIT 97157 09:59:128
+ MODSTAT VOL=IMS500 DSN=IMS.V500.MODSTAT
```

The first line of the display indicates whether or not an online change operation is currently in progress. The IMS /MODIFY PREPARE command invokes an online change operation.

The second line of the display shows both the current online change ID and the suffix of the IMS nucleus module (DFSVNUCx) currently in use.

The third line of the display shows the address of the modify work area, which IMS allocates during the PREPARE phase of online change. This work area contains the change, add, and delete lists for every class of resource eligible for online change, as well as various status and option flags. IMS uses the information that this work area contains during the COMMIT phase of online change. IMS deletes the work area at the end of COMMIT or ABORT phase processing.

The fourth line of the display shows which of the MODBLKSA/MODBLKSB, IMSACBA/IMSACBB, and dataset areas are in use. IMS uses the online change function to swap among one or more of these pairs of datasets.

The fifth line of the display shows the date and time that the command was issued. If this information is not available, OCHG displays N/A.

The sixth line of the display shows the date and time that the :i1./MODIFY command command was issued.

The seventh line of the display shows the MODSTAT dataset name and the volser.

The eighth line of the display shows whether XRF is available. If it is, this line shows the VOLSER and DSN for the standby system. If XRF is not available, MODSTAT2 displays NOT XRF.
In addition, the OCHG command also displays if security failed during an online change. The possible security statuses are:

- **Pwd security**: Password security failed.
- **AGT security**: Application group security failed.
IMS Dataset Information

Introduction

The commands to display information and statistics about IMS datasets are listed and described on the following pages.

<table>
<thead>
<tr>
<th>IDDN</th>
<th>Displays information and statistics about IMS datasets.</th>
</tr>
</thead>
</table>

**Type:** Major

Used with its minor commands, IDDN displays information about the specified type of IMS dataset. Each minor command of IDDN displays the following information:

- the dataset name
- the volume serial number and unit address of the disk on which the dataset resides
- the logical record length and block size of the dataset
- the total number of I/Os that IMS has issued against the dataset
- the I/O rate over the last OMEGAMON cycle

**Note:** The first time you issue IDDN, OMEGAMON displays "Initializing" for the rates.

<table>
<thead>
<tr>
<th>ACB</th>
<th>Displays ACBLIB dataset information.</th>
</tr>
</thead>
</table>

**Type:** Minor of IDDN

**Format:** ACBe

OMEGAMON displays information about both the A and the B datasets and highlights the active dataset of each pair. To specify a particular dataset, type one of the following arguments after the command name:

- **A** Displays only the A dataset.
- **B** Displays only the B dataset.
- ***** Displays only the active dataset.

For example, ACBA displays information about the A dataset only.
The following example shows the ACB command display.

<table>
<thead>
<tr>
<th>IDDN</th>
<th>IMS Dataset Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>acb</td>
<td>DDNAME = ACBLIBA Status = Inactive -- Closed</td>
</tr>
<tr>
<td>+</td>
<td>Unit address = 14B</td>
</tr>
<tr>
<td>+</td>
<td>Logical record length = 0 Blocksize = 0</td>
</tr>
<tr>
<td>+</td>
<td>CTL I/O Count = 0 CTL I/O Rate = .00 per second</td>
</tr>
<tr>
<td>+</td>
<td>DLS I/O Count = 176 DLS I/O Rate = .00 per second</td>
</tr>
<tr>
<td>+</td>
<td>DDNAME = ACBLIBB Status = Active -- Open</td>
</tr>
<tr>
<td>+</td>
<td>Unit address = 14B</td>
</tr>
<tr>
<td>+</td>
<td>Logical record length = 0 Blocksize = 23476</td>
</tr>
<tr>
<td>+</td>
<td>CTL I/O Count = 47 CTL I/O Rate = .00 per second</td>
</tr>
<tr>
<td>+</td>
<td>DLS I/O Count = 176 DLS I/O Rate = .00 per second</td>
</tr>
</tbody>
</table>

If you use LSO=S, the ACBLIB I/O counts appear for both the DLISAS region and for the IMS control region.

**DDNM** Displays dataset and device I/O statistics for a specific DDName.

**Type:** Minor of IDDN

The following example shows the DDNM command display.

<table>
<thead>
<tr>
<th>IDDN</th>
<th>IMS Dataset Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>ddnm</td>
<td>DDNAME = DI21PARO</td>
</tr>
<tr>
<td></td>
<td>Jobname = IMS510AD</td>
</tr>
<tr>
<td></td>
<td>Status = Open</td>
</tr>
<tr>
<td></td>
<td>DSNAME = IMS.V5R1.DI21PARO</td>
</tr>
<tr>
<td></td>
<td>Index CI Size = 0 Data CI Size = 4096</td>
</tr>
<tr>
<td></td>
<td>CI Splits = 0 CA Splits = 0</td>
</tr>
<tr>
<td></td>
<td>EXCP Count = 3 EXCP Rate = .00 per second</td>
</tr>
<tr>
<td></td>
<td>UCB VOLSER EXCP Count EXCP Rate</td>
</tr>
<tr>
<td>----</td>
<td>------ ---------- ---------</td>
</tr>
<tr>
<td>04F8</td>
<td>PPSMP3 3 .00</td>
</tr>
</tbody>
</table>

This command displays the I/O rates for all the datasets allocated to the specified DDName. All the devices for each dataset are listed with the I/O rates calculated for the EXCPs for a particular device for the particular datasets.

**MDBL** Displays information about MODBLKS datasets.

**Type:** Minor of IDDN

**Format:** MDBLc
OMEGAMON displays information about both the A and the B datasets and highlights the active dataset of each pair. To specify a particular dataset, type one of the following arguments after the command name:

**A** Displays only the A dataset.

**B** Displays only the B dataset.

**_* Displays only the active dataset.

For example, MDBLA displays information about the A dataset only.

The following figure shows an example of the MDBL command display.

![Example of the MDBL command display](image)

**MDST** Displays MODSTAT dataset information.

**Type:** Minor of IDDN

The figure that follows shows the MDST command display.

![Example of the MDST command display](image)

**RCNS** Displays database recovery control (DBRC) RECON dataset information.

**Type:** Minor of IDDN
The following example shows a typical RCNS command.

```
+ DDNAME = RECON1                  Status = Active -- Open
+      DSNAME = IMSV.V500.RECON01
+      Unit Address = 161               Volume = IMS210
+      Index CI Size = 4096            Data CI Size = 4096
+      CI Splits = 0                CA Splits = 0
+      I/O Count = 39              I/O Rate = .00 per second
```

RDS Displays restart dataset information.

**Type:** Minor of IDDN

The following example shows a typical RDS command.

```
+ DDNAME = RDS                     Status = Open
+      DSNAME = IMS.V500.RDS
+      Unit address = 161               Volume = IMS200
+      Logical record length = 23476   Blocksize = 23476
+      I/O Count = 5                I/O Rate = .00 per second
```

SUMM Displays dataset I/O statistics for IMS system datasets.

**Type:** Minor of IDDN
This command displays an overview of the I/O rates for the various IMS system datasets.

<table>
<thead>
<tr>
<th>Jobname</th>
<th>DDName</th>
<th>Status</th>
<th>EXCP Count</th>
<th>EXCP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMS510AC</td>
<td>IMSACBA</td>
<td>ACT/OPEN</td>
<td>53</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AD</td>
<td>IMSACBA</td>
<td>ACT/OPEN</td>
<td>213</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>IMSACBB</td>
<td>INACT/CLOSED</td>
<td>0</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AD</td>
<td>IMSACBB</td>
<td>INACT/CLOSED</td>
<td>0</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>IMSTFMTA</td>
<td>ACT/OPEN</td>
<td>0</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>IMSTFMTB</td>
<td>INACT/CLOSED</td>
<td>0</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>LGMSG</td>
<td>OPEN</td>
<td>129</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>SHMSG</td>
<td>OPEN</td>
<td>58</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>QBLKS</td>
<td>OPEN</td>
<td>66</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>FORMATA</td>
<td>ACT/OPEN</td>
<td>48</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>FORMATB</td>
<td>INACT/CLOSED</td>
<td>0</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AB</td>
<td>RECON1</td>
<td>OPEN</td>
<td>237</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AB</td>
<td>RECON2</td>
<td>OPEN</td>
<td>223</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>IMSRDS</td>
<td>OPEN</td>
<td>13</td>
<td>.00</td>
</tr>
<tr>
<td>IMS510AC</td>
<td>MODBLKSA</td>
<td>ACT/CLOSED</td>
<td>10</td>
<td>.00</td>
</tr>
</tbody>
</table>
IMS Trace Table Entries

Introduction

Use the TRAC command to display IMS trace table entries.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format:</td>
<td>TRACcc PST=n FUNC=xx</td>
</tr>
</tbody>
</table>

cc Specifies the type of trace:

DG Disk trace log.
DL DL/I trace table entries.
DS Dispatcher trace table entries.
FO Force trace.
FP Fast Path trace.
LA Latch trace.
SC Scheduler trace table entries.
SS Subsystem trace.

PST=n Specifies the number of the PST.

FUNC=xx Specifies the number of hexadecimal trace entries.

The following example shows a small portion of the dispatcher trace table.
Plotting System Information

Introduction

Use the PLOT command to produce graphs of various system resources.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLOT</td>
<td>Graphically displays historical information about IMS resources.</td>
</tr>
</tbody>
</table>

**Type:** Immediate  
**Format:** cPLOT aaaa

OMEGAMON collects data about IMS resources and stores this data in an internal table. The PLOT command displays information from this internal table.

The PLOT command accepts one, two, or three 4-character resource names (aaaa) as parameters. If you supply more than three names, PLOT displays the message **EXTRANEOUS NAMES IGNORED. PLOT CAN ONLY SHOW THREE PLOTS PER COMMAND**.

The PLOT command displays information about the following resources:

- **CPUC**: Amount of CPU utilized.
- **IORT**: IMS I/O rate per second.
- **PGAV**: Private page-in rate per second for all non-dependent regions.
- **PGCR**: Private page-in rate per second for the control region.
- **PGDB**: Private page-in rate per second for the DBRC region.
- **PGDL**: Private page-in rate per second for the DLI region.
- **PGIR**: Private page-in rate per second for the IRLM region.
- **ROAV**: Average region occupancy for all dependent regions.

If you supply a name that PLOT does not recognize, it displays the message **NOT DEFINED**.

The PLOT command uses the graph character you supply in the label field (c) to plot each point on the graph. If you do not supply a graph character, PLOT uses the default (an asterisk).

PLOT displays information in a vertical bar graph. The following figure shows the results of a typical PLOT command. The vertical bars furthest to the right of the graph represent the most current data. As you move to the left, the
vertical bars represent an averaged value, which OMEGAMON updates every 2, 4, 8, 16, 32, 64, and 128 cycles respectively.

Each graph contains an upper and lower threshold line. When a resource equals or exceeds the upper threshold, that resource is considered in danger. If a resource is below the lower threshold, that resource may or may not be in danger, depending on the particular resource and your operating environment. Use the PSET major command and its minors to adjust the upper and lower thresholds.

If the value in the column representing data averaged over the last two cycles (the column to the left of the current cycle) exceeds the upper threshold, PLOT displays a horizontal line of five highlighted asterisks below the resource graph.

In extended color mode, PLOT displays the body of the graph in reverse video. The upper threshold displays in red. The lower threshold displays in blue. If the value in the column representing data averaged over the last two cycles (the column to the left of the current cycle) exceeds the upper threshold, PLOT displays the body of the graph in red. If the value in the column representing data averaged over the last two cycles is between the two thresholds, PLOT displays the body of the graph in yellow. If the value in the column representing data averaged over the last two cycles is less than the lower threshold, PLOT displays the body of the graph in blue.

If you enter the command PLOTX instead of PLOT in extended color mode, each vertical column displays in its own color: red for values above the red line, yellow for values between the two thresholds, and blue for values below the blue line.
The following example shows a PSET command with the PGCR minor command:

```
PSET
PGCR M=30 U=22 L=7
```

The above commands set the maximum Y-axis scale value of the private page-in rate plot to 30, the upper threshold to 22, and the lower threshold to 7.

The following figure shows a sample of PLOT command results for PGCR.

**FIGURE 7. Sample PLOT command results**

The upper threshold value must be greater than the lower threshold value. The maximum Y-axis scale value must be greater than the upper and lower threshold values.
If the lower threshold is greater than the upper threshold, or if the maximum Y-axis scale value is less than the upper threshold, PSET displays an error message and does not set these values.

If you issue one of the PSET minor commands without an M=, U=, or L= parameter, the minor command displays the current M=, U=, and L= parameters for that resource.
Chapter overview

This chapter provides information about OMEGAMON commands that display information about disk and tape devices.

Chapter contents

Device Listing Commands ........................................ 166
Disk Information .................................................... 168
Long-Term Device Utilization .............................. 177
Device Listing Commands

Introduction

These immediate commands display lists of devices.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLST</td>
<td>Lists all online and offline disks.</td>
</tr>
</tbody>
</table>

Type: Immediate

A hyphen (-) in the command display indicates the selected offline disks.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLST</td>
<td>Lists esoteric and generic device names by device class.</td>
</tr>
</tbody>
</table>

Type: Immediate

Format: GLST <cccc|cccc ...>

The GLST display can be limited to specific device classes with the following operands:

- Displays all defined device names. Blank is the default.
- ALL Displays all device names. ALL produces the same display as blank.
- CHAR Displays character reader devices.
- COMM Displays communication devices.
- CTC Displays channel-to-channel devices.
- DASD Displays direct access storage devices.
- DISP Displays display devices.
- TAPE Displays tape devices.
- UREC Displays unit record devices.
These operands can be used in combination. For example, to display all tape devices and communications devices, type:

```
GLST TAPE COMM
```

Use of the optional operands is limited to MVS/ESA SP4.1 and above. See also the GDEV command.
Disk Information

Introduction

This section describes the major commands that select disks and the minor commands that display detailed information about the disks.

Disk select major commands

<table>
<thead>
<tr>
<th>DEV</th>
<th>Selects a disk with volser ccccccc or address xxx.</th>
</tr>
</thead>
</table>

Type: Major
Format: DEV ccccccc|xxx

If you supply the volser or the address of a disk, DEV displays the volser, the address, and the online or offline status of the disk.

If you enter:

```
DEV 520
```

the result is:

```
DEV 520 volser=TSO099  Online  Alloc
```

Alternatively, you could supply the volser (TSO099).

<table>
<thead>
<tr>
<th>DEVL</th>
<th>Selects list of online disks by volser ccccccc or unit address xxx.</th>
</tr>
</thead>
</table>

Type: Major
Format: DEVL ccccccc ... ccccccc|xxx ...xxx

DEVL selects a list of disks for examination, whereas DEV selects only one device at a time.

The following example shows disks at addresses 123 and 141 and volser TSO021 and TSO022:

```
DEVL 123 TSO021 TSO022 141
```

If you list an invalid address or volser, or if you specify the same disk twice, OMEGAMON eliminates the invalid or duplicate value from the list.
The variable \( n \) is a number from 0–9 indicating the pattern set with the .SPT immediate command.

For example, you could set the pattern and then display all online disks beginning with TSO as shown in this figure:

<table>
<thead>
<tr>
<th>.SPT/9</th>
<th>TSO*</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEVP/9</td>
<td>TSO024</td>
</tr>
<tr>
<td></td>
<td>TSO025</td>
</tr>
<tr>
<td></td>
<td>TSO021</td>
</tr>
<tr>
<td></td>
<td>TSO022</td>
</tr>
<tr>
<td></td>
<td>TSO023</td>
</tr>
<tr>
<td></td>
<td>TSO069</td>
</tr>
</tbody>
</table>

An argument of AL (DISKAL) displays all offline disks as well as those online.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Major</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format:</td>
<td><strong>DEVP/n</strong></td>
</tr>
</tbody>
</table>

| DEVP | Selects a list of online disks using patterns set with .SPT. |

| DISK | Selects online disks. |
| Type: | Major |

| DSKB | Selects busy disks. |
| Type: | Major |

| DSKC | Selects disks with suspended channel programs. |
| Type: | Major |

| DSKE | Selects permanently resident disks. |
| Type: | Major |

| DSKG | Selects mass storage (MSS) virtual disks. |
| Type: | Major |

| DSKM | Selects disks waiting for mounts. |
| Type: | Major |

| DSKN | Selects disks with volsers that start with cc. |
| Type: | Major |
| Format: | **DSKNcc** |
The following example selects disks with volsers that begin with the characters TS, such as all TSO disks:

**DSKNTS**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKP</td>
<td>Selects DASD volumes with a mount status of PUBLIC.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>DSKQ</td>
<td>Selects disks with I/O queue length of ( nn ) or more.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td>DSKQ( nn )</td>
</tr>
</tbody>
</table>

The DPLT\( nn \) minor command plots a microscopic analysis of device utilization. You can use it to investigate disks with I/O queues.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKR</td>
<td>Selects disks with a RESERVE currently issued from this CPU.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>DSKS</td>
<td>Selects DASD volumes with a mount status of STORAGE.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>DSKU</td>
<td>Selects disks with UCBnames starting with xx.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td>DSKU( xx )</td>
</tr>
</tbody>
</table>

The variable \( xx \) specifies the UCBnames with which the disks begin. This may or may not correspond to control unit \( xx \). DSKU\( xx \) does not consider alternate paths.

The following example selects disks with UCBnames that start with 58, which includes disks 580 through 58F:

**DSKU58**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKV</td>
<td>Selects DASD volumes with a mount status of PRIVATE.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

**GDEV** lists devices with the name \( cccccc \).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td>nGDEV cccccc</td>
</tr>
</tbody>
</table>
This major command lists disk and tape devices with the generic name `cccccc`. The value of `n` can be one of the following:

- the less-than symbol (<) to display all devices
- the numbers 1–9 to display individual rows of the list of devices
- the letters A–Z (representing 10–35) to display individual rows of the list of devices

For example, SYSDA is a generic device name in the system. To list all devices with the generic name SYSDA, enter the following:

```
<GDEV SYSDA
```

You can specify the number of entries in the device name table with the `GDEVUCBS` keyword of the `.SET` command. The maximum is 4000.

**Note:** Use the `GLST` major command to list the generic names in the system.

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2305</td>
<td>Displays 2305 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

For non-base exposures, the unit address and exposure number appear instead of the volser.

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3330</td>
<td>Displays 3330 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td><code>3330nn</code></td>
</tr>
</tbody>
</table>

An argument of 11 displays 3330-11 disks.

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3340</td>
<td>Displays 3340 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3350</td>
<td>Displays 3350 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td><code>3350c</code></td>
</tr>
</tbody>
</table>

An argument of P displays the disks attached to 3880-11 or 3880-21 buffered-paging facility control units. In addition, for non-base exposures, the unit address and exposure number appear instead of the volser.

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3375</td>
<td>Displays 3375 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Device</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3380</td>
<td>Displays 3380 disks.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>
### Disk Information

You can use the following minor commands immediately after a disk major command to display information about the disk drives selected by the disk major commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCAT</td>
<td>Displays whether a device is static, installation-static, or dynamic.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DADR</td>
<td>Displays the unit address of the device.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DALC</td>
<td>Displays number of allocations to the device.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DIO</td>
<td>Displays EXCPs issued to a device.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DIOQ</td>
<td>Displays I/O queue length on the disk.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DOPN</td>
<td>Displays number of open DCBs and ACBs on the device.</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>DPIN</td>
<td>Displays whether a device is pinned.</td>
<td>Minor of disk majors</td>
</tr>
</tbody>
</table>

**Selected-disk minor commands**

DCAT will display **STATIC**, **I-STATIC**, or **DYNAMIC**, to indicate the device category.

The DCAT command applies to MVS/ESA SP4.2 and above.

**Note:** If RMF is not currently monitoring the device when you issue DIO, the command displays the message **UNMNTRD**.

DIO requires RMF.
The DPIN command applies to MVS/ESA SP4.2 and above.

<table>
<thead>
<tr>
<th>DPLT</th>
<th>Displays device activity every ( nn ) milliseconds.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of disk majors</td>
</tr>
<tr>
<td>Format:</td>
<td>DPLT( nn )</td>
</tr>
</tbody>
</table>

DPLT allows inspection of processes that occur between OMEGAMON cycles. For any given major device command, DPLT takes 50 samples every \( nn \) milliseconds. DPLT only plots the first device that the major specifies. Each sample indicates changes in device status and user.

As the plot progresses from left to right, a number of fields show either the status of the address space or its activity since the last sample. One column in the display represents each sample.

If you use DPLT as a minor command of DSKQ, OMEGAMON only performs the sampling when there is a need, such as when a disk is selected by the DSKQ command because it has an I/O queue length of \( nn \) or more.

To monitor a specific device, use DPLT as a minor command of DEV.xxx, where xxx specifies the device address.

The following figure shows an example of the DPLT command used with the DSKQ major.

**FIGURE 8. The DPLT major command with the DSKQ minor command**

To report on the I/O# subfield properly, DPLT requires RMF to be monitoring the device.

The Interval value to the right is the true interval between samples, calculated after DPLT completes processing. On a system running perfectly, this number is the same as the \( nn \) sampling interval. This number may vary due to your system’s workload.

- **DBsy**: D indicates device busy.
- **IOQ**: S (XA and ESA) indicates suspended channel program.
- **IOQ Length**: Length of IOQ.
Figure 8 on page 173 shows that the DSKQ01 command selected a disk with a volser of MVS005 because it had an I/O queue length of one or more. The minor command DPLT03 plots the activity on the selected device at 3 millisecond intervals (OMEGAMON always takes 50 samples). The resulting plot shows two jobs (PAYROLL and SORT) competing for the disk arm at cylinder addresses 280 and 403. This explains the I/O queue on the selected device. Even though we specified a 3 millisecond interval (DPLT03), the actual interval that appears is 4 milliseconds (Interval: 4). This is because other address spaces operated at a higher priority than OMEGAMON at the time.

<table>
<thead>
<tr>
<th>I/O#</th>
<th>Wraparound I/O#. (The &gt; indicates the point at which OMEGAMON calculates a new wraparound I/O#. The new number appears to the right of the &gt;.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Can be one of these symbols:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>. CPU dispatchable</td>
</tr>
<tr>
<td></td>
<td>. Waiting</td>
</tr>
<tr>
<td>User</td>
<td>Current user. (The &gt; indicates the point at which a new user had the device.)</td>
</tr>
<tr>
<td>Note:</td>
<td>OMEGAMON may not be able to identify the user of a device if that user is a system routine or utility issuing its own seek commands.</td>
</tr>
<tr>
<td>Cyl</td>
<td>Cylinder address. (The &gt; indicates the point at which a new cylinder was accessed.) Rls in this field indicates that a stand alone release is in progress, and therefore no cylinder is involved.</td>
</tr>
<tr>
<td>Nrdy</td>
<td>Not ready.</td>
</tr>
<tr>
<td>Resv</td>
<td>Device reserved this CPU.</td>
</tr>
</tbody>
</table>

Figure 8 on page 173 shows that the DSKQ01 command selected a disk with a volser of MVS005 because it had an I/O queue length of one or more. The minor command DPLT03 plots the activity on the selected device at 3 millisecond intervals (OMEGAMON always takes 50 samples). The resulting plot shows two jobs (PAYROLL and SORT) competing for the disk arm at cylinder addresses 280 and 403. This explains the I/O queue on the selected device. Even though we specified a 3 millisecond interval (DPLT03), the actual interval that appears is 4 milliseconds (Interval: 4). This is because other address spaces operated at a higher priority than OMEGAMON at the time.

| DRES       | Displays device reserve count from this CPU.                                                   |
| Type:      | Minor of disk majors                                                                           |
| DSTA       | Displays mount status.                                                                         |
| Type:      | Minor of disk majors                                                                           |

Status can be:
- PRIVATE
- PUBLIC
- STORAGE

| DTYP       | Displays disk type.                                                                            |
| Type:      | Minor of disk majors                                                                           |
Disk type can be 3380, 3390, and so on.

**DUSR**

<table>
<thead>
<tr>
<th>Displays current user of device.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Minor of disk majors</td>
</tr>
</tbody>
</table>

**DVMP**

<table>
<thead>
<tr>
<th>Displays unit control block (UCB) hex dump.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Minor of disk majors</td>
</tr>
</tbody>
</table>

DVMP dumps the UCB, the UCB prefix, and all appropriate extensions for the disk. It also shows the device status.

The following screen shows a typical DVMP display.

```
DEV     SYS640
dvmp  Mount Status: Perm_Res Private
+      Status:
+      User: *MASTER*            Waiting I/O's:
+      Status: Ch_Active - Suspended Channel Program <Paging Device>
+      UCB Prefix: 001188
+      00000000 00FF39B4
+      UCB Common + DASD Device Dependent Segment: 001190
+      008BFF8C 0240A201 B0000100 00F1F4F0 3010200E 0008D38 19D70100 E2E8E2F0
+      F2F45000 00000400
+      Common Extension: 008D38
+      00000000 18820040 000A0000 F755001A 01000000 0032898 0000EAF0 0010005
```

**DVOL**

<table>
<thead>
<tr>
<th>Displays the volser of a selected device.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Minor of DEV and DEVL</td>
</tr>
</tbody>
</table>

**ICHPn**

<table>
<thead>
<tr>
<th>Displays installed channel paths for the disk (XA).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Minor of disk majors</td>
</tr>
</tbody>
</table>

The value of $n$ is a number from 1–8.

**OCHP**

<table>
<thead>
<tr>
<th>Displays online channel paths for the disk (XA or ESA).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Minor of disk majors</td>
</tr>
<tr>
<td>Format: OCHPn</td>
</tr>
</tbody>
</table>

The value of $n$ is a number from 1–8.

**Examples of disk commands**

The following are examples of how to use the disk commands.
This example shows how to use the DSKU major command to monitor the I/O rates of devices at 160 to 16F.

<table>
<thead>
<tr>
<th>Disk Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKU16 IMS001 IMS002 ....</td>
</tr>
<tr>
<td>dadr 160 161 ....</td>
</tr>
<tr>
<td>dio .R 1.4 0 ....</td>
</tr>
<tr>
<td>DSKU16 SCRAT1 SCRAT2 ....</td>
</tr>
<tr>
<td>dadr 168 169 ....</td>
</tr>
<tr>
<td>dio .R 2.5 8.7 ....</td>
</tr>
</tbody>
</table>

The next example shows I/O queue length. For any device with an I/O queue length greater than or equal to 2, the minor commands display the current user, the I/O queue length, and the address.

<table>
<thead>
<tr>
<th>Disk Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKQ02 MVSRES SPOOL</td>
</tr>
<tr>
<td>dusr <em>MASTER</em> JES2</td>
</tr>
<tr>
<td>dioq 3 2</td>
</tr>
<tr>
<td>dadr 163 167</td>
</tr>
</tbody>
</table>

The next example selects all private disks.

<table>
<thead>
<tr>
<th>Disk Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKV DB0021 DB0022 DB0023 DB0024 KN0056 KB0087</td>
</tr>
<tr>
<td>dusr MESSAGE1 MESSAGE1</td>
</tr>
<tr>
<td>dio .R 2.3 5.3</td>
</tr>
</tbody>
</table>

The next example selects all disks whose volsers begin with WO.

<table>
<thead>
<tr>
<th>Disk Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSKNWO WORK01 WORK02</td>
</tr>
<tr>
<td>dadr 140 147</td>
</tr>
</tbody>
</table>
Long-Term Device Utilization

Introduction

The commands to display long-term device utilization are described on the following pages.

<table>
<thead>
<tr>
<th>IDEV</th>
<th>Displays device activity data from all IMS system control address spaces.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Major</td>
<td></td>
</tr>
<tr>
<td>Format: cIDEV</td>
<td></td>
</tr>
</tbody>
</table>

Long-term utilization displays are available if RMF is active for the desired devices. If RMF is not active for the devices, the average queueing and response time information does not appear.

IDEV can also display device activity data from IMS dependent regions. Use region commands (RGNc) to select a region for IDEV. The IDEV minor commands (PDSK, SDSK, and XDSK) display device activity about that region. The IDEV command can use the following label field arguments to select target IMS address spaces or OLDS and WADS devices:

- (blank) Displays devices allocated to the IMS control region.
- D or S Displays devices allocated to the DLISAS address space.
- F or I Displays devices allocated to the IMS control region (default).
- L or M Displays devices allocated to the IRLM address space.
- O Displays devices currently being used for DASD logging OLDS.
- P Displays devices allocated to the IMS dependent region that the preceding RGNc major command selected.
- R or C Displays devices allocated to the DBRC address space.
- W Displays devices currently being used by IMS logging WADs.

PDSK aaaaa selects any device (DASD) whose volume serial number matches the pattern aaaaa specifies. Use alphanumeric characters to specify the pattern.

In the pattern, you can also use asterisks (*) as wildcard characters. For example, the pattern IMS**1 causes OMEGAMON to select volser IMS001, IMS011, IMS201, and so on. If an asterisk is the last character in the pattern, any number of characters after that point match. For example, the pattern VS* selects the volumes VSRESA, VSAM01, VS, and so on.

SDSKxx selects any device of the indicated type that belongs to the xx string of devices (that is, any device whose device address begins with xx).

XDSK selects any DASD device that exceeds certain thresholds.
These threshold commands do not produce output; they simply control which devices XDSK will display.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVQ</td>
<td>Sets threshold for average IOS queue depth.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IDEV</td>
</tr>
<tr>
<td>Format:</td>
<td>AVQnnn</td>
</tr>
</tbody>
</table>

After AVQ, you can specify the following:

- **blank**: Displays current threshold.
- **nn**: Specifies the threshold for the XDSK command in milliseconds.
- **OF**: Turns off the previously set threshold.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CON</td>
<td>Sets the threshold for the average device connect time.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IDEV</td>
</tr>
<tr>
<td>Format:</td>
<td>CONnnn</td>
</tr>
</tbody>
</table>

After CON, you can specify the following:

- **blank**: Displays current threshold.
- **nn**: Specifies the threshold for the XDSK command in milliseconds.
- **OF**: Turns off the previously set threshold.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSC</td>
<td>Sets the threshold for the average device disconnect time.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IDEV</td>
</tr>
<tr>
<td>Format:</td>
<td>DSCnnn</td>
</tr>
</tbody>
</table>

After DSC, you can specify the following:

- **blank**: Displays current threshold.
- **nn**: Specifies the threshold for the XDSK command in milliseconds.
- **OF**: Turns off the previously set threshold.
### DUT
Sets the threshold for the average device utilization.

**Type:** Minor of IDEV  
**Format:** \texttt{DUTnnn|OF}

After DUT, you can specify the following:
- **blank** Displays current threshold.
- **nn** Specifies the threshold for the XDSK command in milliseconds.
- **OF** Turns off the previously set threshold.

### IOS
Sets the threshold for the IOS queue time.

**Type:** Minor of IDEV  
**Format:** \texttt{IOSnnn|OF}

After IOS, you can specify the following:
- **blank** Displays current threshold.
- **nn** Specifies the threshold for the XDSK command in milliseconds.
- **OF** Turns off the previously set threshold.

### PND
Sets the threshold for the average device pending time.

**Type:** Minor of IDEV  
**Format:** \texttt{PNDnnn|OF}

After PND, you can specify the following:
- **blank** Displays current threshold.
- **nn** Specifies the threshold for the XDSK command in milliseconds.
- **OF** Turns off the previously set threshold.

### RSP
Sets the threshold for the total device pending time.

**Type:** Minor of IDEV  
**Format:** \texttt{RSPnnn|OF}
Long-Term Device Utilization

After RSP, you can specify the following:

**blank** Displays current threshold.

**nn** Specifies the threshold for the XDSK command in milliseconds.

**OF** Turns off the previously set threshold.

The example below shows long-term device utilization for volser's whose names match the pattern DI*.

FIGURE 9. PDSK under IDEV display

This data pertains to the current RMF interval. The IDEV command displays when the last RMF interval started, and how much of the current interval has elapsed.

FIGURE 10. PDSK display of devices allocated to the DLS address space

---

**Legend**

**Volser** Device volume serial number.

**Rate** I/O rate (I/Os per second) for both the total system and IMS.

**Util%** Device utilization percentage.

**Avg.Q** Average IOS queue length.

**Resp** Average total device response time in milliseconds.

**IOSQ** Average IOS queueing time in milliseconds.

**Pend** Average pending time in milliseconds.

**Conn** Average connect time in milliseconds.

**Disc** Average disconnect time in milliseconds.

---

This data pertains to the current RMF interval. The IDEV command displays when the last RMF interval started, and how much of the current interval has elapsed.
You can analyze devices allocated to various control region address spaces. In addition, you can analyze devices allocated to dependent regions. Use an RGNC command to select the target dependent region. OMEGAMON analyzes devices allocated to the first region.

The following statements show how to filter control regions:

- iIDEV * control region
- fIDEV * control region
- dIDEV * DLS address space
- sIDEV * DLS address space
- rIDEV * DBRC address space
- cIDEV * DBRC address space
- lIDEV * IRLM address space
- mIDEV * IRLM address space
- oIDEV * volumes used for the current OLDS
- wIDEV * volumes used for the current WADS

The following statements show how to analyze dependent region devices:

- RGND IMSMPP01 IMSMPP03
- PIDEV
- PDSK * devices for region IMSMPP01

The following statement shows how to analyze DASD logging of OLDS devices:

**FIGURE 11. Typical analysis of OLDS activity using IDEV**

<table>
<thead>
<tr>
<th>OIDEV</th>
<th>Interval Start Time: 17:44:34</th>
<th>Elapsed: 3:41 MN</th>
<th>Length: 14:25 MN</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDSK</td>
<td>Volser I/O per second Util% Avg.Q Resp = IOSQ + Pend + Conn + Disc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>Total IMS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ 156</td>
<td>DISK01 4.8 2.1 13.0 .02 27.8 3.8 4.5 12.8 6.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ 153</td>
<td>DISK04 3.5 1.3 6.6 .01 25.0 3.0 5.1 15.2 1.7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following statement shows how to analyze DASD logging of WADS devices:
**FIGURE 12. Typical analysis of WADS activity using IDEV**

<table>
<thead>
<tr>
<th>WIDEV</th>
<th>Interval Start Time</th>
<th>Elapsed</th>
<th>Length</th>
<th>Volser</th>
<th>I/O per second</th>
<th>Util%</th>
<th>Avg.Q</th>
<th>Resp</th>
<th>IOSQ + Pend + Conn + Disc</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISK02</td>
<td>11.1</td>
<td>5.5</td>
<td>7.8</td>
<td>.01</td>
<td>24.3</td>
<td>3.8</td>
<td>20.5</td>
<td>0.00.0</td>
<td></td>
</tr>
</tbody>
</table>
Chapter overview

This chapter provides information about program specification blocks (PSB), which are internal IMS control blocks that define a site’s application programs. IMS knows each PSB by an 8-character name you choose during the IMS system generation process. As such, this manual uses the term PSB as a synonym for program or application. By creating PSB groups, you can target the information that OMEGAMON collects and reports. After you create groups, you can target service level analysis for threads, focusing your tuning efforts on the specific bottlenecks that impact your system.

Chapter contents

Defining PSB Groups .......................................................... 184
Specifying Maximum Number of PSB Groups ....................... 187
Activating New PSB Groups ................................................. 188
Defining PSB Groups

Introduction

Use the SETG command to change PSB assignments dynamically. To assign PSB groups permanently, define them in the KOIGBL module. (For information about the KOIGBL module, see the Configuration and Customization Guide.)

<table>
<thead>
<tr>
<th>SETG</th>
<th>Provides a list of criteria for inclusion of a given PSB in a group.</th>
</tr>
</thead>
</table>

Type: Immediate

To determine whether or not a PSB belongs in a given group, OMEGAMON compares the name of a PSB to the argument that you specify for the group when you issue one or more SETG commands.

Format:

```
xSETGnn PSB=<PSB names...>       (names are 1-8 characters)
  xSETGnn no keyword, if only listing a PSB group’s contents.

  +------>  PSB group number (01-30 or max group # defined at installation time or via the MAXG command. Use the number 99 to list, add, or delete from all groups.)
  +-------->  A  -  add an entry to group nn
               C  -  create group nn
               D  -  delete an entry from group nn
               L  -  list group nn
```

When SETG completes an add, delete, or change request, it converts the label field to L (list), so that SETG does not re-execute the change on the next cycle.

The SETG command allows generic PSB names. When an asterisk (*) appears in a PSB name, SETG accepts any character in that position. For example, the command

```
CSETG01 PSB=AR
```

causes PSB group 01 to consist of all PSBs whose names contain the letters AR in position two and four. Any character can fill positions one and three.

A trailing asterisk matches all remaining characters. Therefore, entering PSB=*AR* is the same as entering PSB=*AR*****. In this example, the PSBs selected for group 01 include both PART and MARK01.

Any PSB group definitions and alterations you make using the SETG command last only for the duration of the current OMEGAMON session. You can, however, set up screen spaces which contain any group definitions which you want to use again. For information on how to define or modify these
Defining PSB Groups

Defining PSB Groups

Creating a PSB group
To create a new PSB group, use the SETG command with the letter C in the label field.

**Note:** You must stop DEXAN before creating PSB groups.

To create a PSB group which contains PSB names PART, ADDPART, and DLT0, type the following command:

```
CSETG03 PSB=PART,ADDPART,DLT0
```

OMEGAMON alphabetizes the names and responds as follows:

```
PSB=ADDPART, DLT0, PART
```

Adding entries to a PSB group
To add PSBs to an existing group, use the SETG command with the letter A in the label field.

To add PSBs to PSB group 03 (which already contains the PSB names ADDPART, DLT0, and PART), type the following command:

```
ASETG03 PSB=DLETPART
```

To acknowledge your request, OMEGAMON displays the IDs of all PSBs in the redefined PSB group, as follows:

```
PSB=ADDPART, DLETPART, DLT0, PART
```

To make the add request effective for all PSB groups, specify 99 as the group number in the SETG command argument.

For example, to add a PSB name to all PSB groups, use the following SETG command:

```
ASETG99 PSB=ADDPART
```

Deleting entries from a PSB group
To delete PSBs from a group, use the SETG command with the letter D in the label field.

To delete PSBs from a group, type a delete command similar to the one shown below:

```
DSETG03 PSB=ADDPART,DLT8
```

groups permanently, see the description of the KOIGBL module in “Activating New PSB Groups” on page 188.
Defining PSB Groups

To acknowledge your request, OMEGAMON displays the IDs of all PSBs in the redefined PSB group, as follows:

```
LSETG03 PSB=ADDPART,DLT8
+ ID(s) not found - DLT8
+ PSB=DLETPART DLT0 PART
```

If you attempt to delete a PSB from a PSB group which does not include it, OMEGAMON displays an error message similar to the "not found" message above.

To delete a PSB from all PSB groups in which it appears, specify a group number of 99 using the following SETG command:

```
DSETG99 PSB=DLT0
```

Deleting all entries from a PSB group

To delete all the PSB names from a PSB group, use the SETG command with the letter X in the label field.

Listing the entries in a PSB group

To list the PSBs in a group, use the SETG command with the letter L or a blank in the label field:

```
LSETG03
```

In response, OMEGAMON displays the names of all the PSBs in the PSB group you select, as follows:

```
LSETG03
+ PSB=DLETPART PART
```

To request a list of all the PSB groups which currently have defined entries, use a group number of 99. A display similar to the following results:

```
LSETG99
+ Group #1 (Name=GROUP 01)
+ PSB=DFSXYZ
+ Group #3 (Name=BILLING )
+ PSB=DLETPART PART
```

The above display indicates that only PSB groups 1 and 3 have defined entries. The display does not include groups, such as PSB group 2, which have no defined entries.
Specifying Maximum Number of PSB Groups

Introduction

Use the MAXG command to specify the maximum number of transaction groups.

**MAXG**

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format:</td>
<td><strong>MAXG</strong> <strong>nn</strong></td>
</tr>
</tbody>
</table>

Dynamically controls the number of PSB groups that OMEGAMON supports.

The value *nn* is the number of PSB groups that OMEGAMON is to support. The valid range is 1 to 30 groups. You can also define the maximum number of PSB groups the KOIGBL module is to support.

If you enter the MAXG command without any operands, the current maximum group value appears in a display similar to the following:

```
MAXG >> 15 is the maximum number of PSB groups allowed <<
```

You cannot define a MAXG value lower than a currently active group ID number. If there are no active group ID numbers higher than the new MAXG value, the new value replaces the current maximum group number. If there are group IDs which are larger than the new value, the maximum group number does not change, and a warning message similar to the following appears:

```
MAXG 5
+ max groups value not changed; following groups exceed MAXG value specified
+ groups= 6 7 10 14
+ enter an X in column 1 to delete these groups and change MAXG value
+ >> 15 is the maximum number of PSB groups allowed <<
```

This display shows that the current maximum group value is 15 and that PSB groups 6, 7, 10, and 14 contain entries. If you still want to reduce the maximum PSB group number to 5, place an X in the label field of the MAXG command as follows:

**XMAXG 5**

The MAXG command then processes, and a display similar to the following one results:

```
XMAXG 5
+ following PSB groups have been deleted
+ groups= 6 7 10 14
+ >> 5 is the maximum number of PSB groups allowed <<
```
Activating New PSB Groups

Introduction

As mentioned above, the KOIGBL module contains the user-defined default definitions for the PSB groups. Rather than change these definitions dynamically via the SETG and MAXG commands, you can alter and reassemble the KOIGBL module. (For details, see the Configuration and Customization Guide.) Once you assemble and link the module, you need to activate the default definitions in the new KOIGBL module.

The GLBL command provides this capability. In addition to letting you reload a new version of the current KOIGBL module (same suffix), it also lets you dynamically switch between multiple KOIGBL modules, each with a different suffix.

Note: You must stop DEXAN before activating new PSB groups.

The format of the GLBL immediate command is as follows.

<table>
<thead>
<tr>
<th>GLBL</th>
<th>Displays or sets the 2-character suffix for the KOIGBL module name.</th>
</tr>
</thead>
</table>

Type: Immediate
Format: **GLBLmp**

*mp* Specifies the suffix.

If you enter the GLBL command without any operands, the suffix of the KOIGBL module currently in use appears. The following is a sample GLBL command output display:

```
GLBL >> mp is current KOIGBL module suffix <<
```

From this display you can see that the module currently in use is KOIGBGLmp.

If you want to reload a new copy of this module, enter the following command:

```
GLBLmp
```

OMEGAMON responds to this command with the following message:

```
>GLBL >> mp is current KOIGBL module suffix <<
```

Note: When a > appears in column 1, the command comments out, preventing re-execution of the command.

If OMEGAMON is unable to locate the KOIGBL module with the specified suffix in any of the load libraries allocated to OMEGAMON, in the link list datasets, or in the link pack area (LPA), the GLBL module does not change, and the following error message appears:

```
>GLBL44 >> OI901: Unable to locate module KOIGBL44<<
```
Chapter overview

This chapter provides information about DBCTL regions. The architecture of IMS requires that IMS assign multiple MVS address spaces to each DBCTL system you start. There is always a control region, a database recovery control (DBRC) region, and some combination of batch message regions (BMPs).

This chapter refers to all address spaces as regions.

Chapter contents

DBCTL Region Immediate Commands ........................................ 190
DBCTL Region Major Commands ............................................. 194
DBCTL Region Minor Commands ............................................. 198
### DBCTL Region Immediate Commands

#### List of commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>THIN</td>
<td>Defines the percentage of threads in use thresholds for CCTLs.</td>
</tr>
</tbody>
</table>

**Type:** Immediate  

**Format:**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>THIN</td>
<td><strong>ADD</strong> cccccccc <strong>THREADS=nnn</strong></td>
</tr>
<tr>
<td></td>
<td><strong>ADD</strong> Adds a threshold for a specific CCTL ID.</td>
</tr>
<tr>
<td></td>
<td><strong>DELETE</strong> Deletes a threshold for a specific CCTL ID.</td>
</tr>
<tr>
<td></td>
<td><strong>LIST</strong> Lists thresholds for specified CCTL IDs.</td>
</tr>
<tr>
<td></td>
<td><em>(blank)</em> Lists all defined thresholds. This is the default.</td>
</tr>
<tr>
<td></td>
<td><strong>ccccc</strong> Specifies the CCTL ID to which the threshold applies. You can use a wildcard (*) as either the first or last character of the CCTL ID only with the LIST operand.</td>
</tr>
<tr>
<td></td>
<td><strong>THREADS=</strong> Specifies the percentage of threads in use (1-100), that will cause the Tnnn exception to trip.</td>
</tr>
</tbody>
</table>

The THIN immediate command assigns a number to each CCTL ID that you have given a THREADS= threshold. This number becomes the last three characters of the Tnnn exception name. If a CCTL exceeds the percentage of threads in use threshold, the Tnnn exception trips to notify you that the CCTL has a high percentage of threads in use.

**Note:** Exception Analysis must be active for the Tnnn exception to provide data.

You can define up to 100 CCTL IDs for the Tnnn exception to monitor during your current session. To save the thresholds that you define with THIN in a user profile, use the PPRF command. If you want to create and save more than 100 thresholds, you can use a second user profile.

The following figure shows an example of adding a threshold for a specific CCTL ID.

```
THIN ADD RCICS321 THREADS=5
+ CCTL THRESHOLD HAS BEEN ADDED
```
The following figure shows the list of all defined thresholds for RCI*, after adding the threshold in this example.

```
THIN LIST RCI*
+ T001 RCICS001 THREADS=10
+ T002 RCICS002 THREADS=100
+ T003 RCICS003 THREADS=20
+ T004 RCICS004 THREADS=10
+ T005 RCICS321 THREADS=5
+ T006 RCICS322 THREADS=9
```

The following figure shows an example of deleting a threshold for a CCTL ID.

```
THIN DEL RCICS322
+ CCTL THRESHOLD HAS BEEN DELETED
```

<table>
<thead>
<tr>
<th>THRDI</th>
<th>Displays each thread’s CCTL ID, Pseudo-Recovery Token, Recovery Token, RGID (PST number), PSB name, and thread status.</th>
</tr>
</thead>
</table>

**Type:** Immediate

The thread status can be any of the following:

- **Active** The thread is actively processing a PSB.
- **Avail** The thread is available to schedule a PSB.
- **Indoubt** The thread has failed and indoubt data exists.
- **Term** The thread is terminating.
- **Unavill** The thread is terminating a scheduled PSB and is therefore unavailable to schedule a new PSB.
- **W-BlkMvr** The thread is waiting for the ACB block mover.
- **W-Intent** The thread is waiting for intent to resolve a conflict with a currently scheduled program’s use of the database.
- **W-PoolSp** The thread is waiting for DMS, PSB, or PSB work area pool space.
- **W-Switch** The thread is waiting to be switched to an alternate system.

The THRDI immediate command provides information about threads that you use with the PNR major command for thread analysis.

```
TTIM
```

<table>
<thead>
<tr>
<th>TTIM</th>
<th>Defines time thresholds for PSBs that IMS schedules on behalf of CCTLs.</th>
</tr>
</thead>
</table>

**Type:** Immediate
The TTIM immediate command compares each active thread with each TTIM threshold to see if a PSB exceeded the CPU or ELAPSED threshold. If a PSB exceeds the CPU or ELAPSED time threshold, the TPSB exception trips to notify you that the PSB may be using more CPU or ELAPSED time than you expected.

**Note:** Exception Analysis and the Transaction Reporting Facility (TRF) must be active for the TPSB exception to provide data. If you start OMEGAMON after IMS schedules a PSB, the TTIM command does not check the PSB against the TTIM thresholds.

You can define up to 100 PSB names for the TPSB exception to monitor during your current session. To save the PSB names that you define with TTIM in a user profile, use the PPRF command. If you want to create and save more than 100 PSB names, you can use a second user profile.

The following figure shows an example of adding a threshold for a CCTL and a PSB.

```
TTIM ADD RCICS410 PSB=ABC* CPU=1 ELAPSED=10
+ PSB THRESHOLD HAS BEEN ADDED
```
The following figure shows the list of all defined thresholds for RCICS410, after adding the threshold in this example.

```
TTIM LIST RCICS410
  + RCICS410 PSB=DFHSAM25 CPU=0001 ELAPSED=0001
  + RCICS410 PSB=DFHSAM2* CPU=0001 ELAPSED=0001
  + RCICS410 PSB=C* CPU=0001 ELAPSED=0001
  + RCICS410 PSB=B* CPU=0001 ELAPSED=0001
  + RCICS410 PSB=ABC* CPU=0001 ELAPSED=0010
  + RCICS410 PSB=*05 CPU=0001 ELAPSED=0001
```

The following figure shows an example of deleting a threshold for a CCTL and a PSB.

```
TTIM DEL RCICS410 PSB=ABC*
  + PSB THRESHOLD HAS BEEN DELETED
```
DBCTL Region Major Commands

Introduction

In order to provide flexibility, a number of different commands are available for selecting IMS regions.

The DBCTL region major commands select only those regions associated with the IMS system for which OMEGAMON is active. While there might be more than one IMS system (and, therefore, more than one DBCTL control region) active at the same time, an OMEGAMON session only associates with a single DBCTL system, and the major commands only select regions belonging to that system. For example, if you have two DBCTL systems called PROD and TEST, with OMEGAMON running in each, the OMEGAMON terminal monitoring the PROD control region completely ignores any DBCTL region associated with the TEST system.

OMEGAMON only recognizes an address space as being an DBCTL dependent region once it makes itself known to the control region via the DBCTL SVC and has an assigned PST. There is a brief period during both dependent region initialization and termination where the address space exists, but OMEGAMON does not see it. Once the IMS control region recognizes the dependent region, OMEGAMON can display it.

If a region major command selects the control region, it always appears first in the display. Dependent regions always appear in the same order you assign them to PSTs, with the exception of the RGNL major command. In the case of RGNL, you supply a list of region names, and OMEGAMON preserves your order.

Note: The region major commands have a special relationship with the IDEV (device activity analysis) command. You can use a region major command to display a list of regions. Then, you can enter an IDEV minor command with a P in the label field to analyze the devices allocated to the first region listed by the region major command. Figure 13 on page 194 shows an example of the analysis of the device activity for all devices allocated to region BMPRG01.

Figure 13. RGN command used with IDEV

<table>
<thead>
<tr>
<th>RGNL</th>
<th>BMPRG01</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Volser Pattern</td>
</tr>
<tr>
<td>+</td>
<td>------</td>
</tr>
<tr>
<td>+</td>
<td>14B</td>
</tr>
<tr>
<td>+</td>
<td>161</td>
</tr>
</tbody>
</table>

BMP Selects all batch message processing regions.
DBCTL Region Major Commands

Type: Major

BMP is an alias for RGNB.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTL</td>
<td>Selects the IMS control region address space.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

CTL is an alias for RGNC.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBRC</td>
<td>Selects the DBRC region only.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

DBRC is an alias for RGNR.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLS</td>
<td>Selects the DL/I subordinate address space if LSO=S is being used.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

DLS is an alias for RGNS.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IRLM</td>
<td>Selects the IRLM region if the IRLM is being used instead of program isolation as the single lock manager.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

IRLM is an alias for RGNI.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNRnnn</td>
<td>Provides access to all the RGNx minor commands for every DBCTL thread for thread analysis.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

The nnn parameter is the RGID of a thread displayed with the THRD command.

PNR displays the jobname of the CCTL associated with the thread identified by the PST. However, output is generated by entering the .EXM immediate command or any of the RGNx minor commands in conjunction with PNR.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDB2</td>
<td>Selects all BMP regions which have DB2® subsystems defined.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

RDB2 is an alias for RGN2.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGNA</td>
<td>Selects all IMS regions, including DBRC, DLISAS, and IRLM.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>
Dependent regions include message processing regions (MPRs), batch message processing regions (BMPs), and Fast Path regions (IFPs).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGNB</td>
<td>Selects all batch message processing (BMP) regions.</td>
</tr>
<tr>
<td>RGNC</td>
<td>Selects the IMS control region address space.</td>
</tr>
<tr>
<td>RGND</td>
<td>Selects all IMS dependent regions.</td>
</tr>
<tr>
<td>RGNH</td>
<td>Selects all IMS BMP regions using High Speed Sequential Processing (HSSP).</td>
</tr>
<tr>
<td>RGNI</td>
<td>Selects the IRLM region if IRLM is being used instead of PI as the single lock manager.</td>
</tr>
<tr>
<td>RGNL</td>
<td>Specifies up to seven regions by job name.</td>
</tr>
<tr>
<td>RGNN</td>
<td>Selects all non-dependent regions.</td>
</tr>
</tbody>
</table>

BMP is an alias for RGNB.

CTL is an alias for RGNC.

OMEGAMON displays all IMS regions except the control region.

IRLM is an alias for RGNI.

OMEGAMON automatically eliminates any invalid names in the list. OMEGAMON considers as valid only those address spaces associated with the IMS control region. Regions display in the same order in which you list them.

Non-dependent regions include the IMS control region, the DBRC region, the DLISAS or DLS region, and the IRLM region.
### DBCTL Region Major Commands

<table>
<thead>
<tr>
<th><strong>RGNP/n</strong></th>
<th>Selects all regions matching a pattern.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>

The .SPT command sets the pattern. See the .SPT command for information about setting patterns. If RGNP does not find a \( n \) value, it uses the first pattern supplied with the last .SPT command.

<table>
<thead>
<tr>
<th><strong>RGNR</strong></th>
<th>Selects the database recovery control region (DBRC).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>

DBRC is an alias for RGNR.

<table>
<thead>
<tr>
<th><strong>RGNS</strong></th>
<th>Selects the DL/I subordinate address space if LSO=S is being used.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>

DLS is an alias for RGNS.

<table>
<thead>
<tr>
<th><strong>RGNT</strong></th>
<th>Selects the RSR Transport Manager Subsystem address space.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>

TMS is an alias for RGNT.

<table>
<thead>
<tr>
<th><strong>RGN2</strong></th>
<th>Selects all BMP regions which have DB2 subsystems defined.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>
DBCTL Region Minor Commands

Introduction
The region minor commands provide more details for any region you select with a region major command. To see a list of minor commands for a particular major, enter .MIN (for minor command) under the major command. There are three categories of minor commands:

- Commands that relate to the IMS environment.
- Commands that provide information related to the MVS environment in which the region runs.
- Commands that relate to Fast Path regions.

DBCTL-related region minor commands
Many commands are available to display information about IMS-related regions. Some of these minor commands do not apply to all types of IMS regions. For example, IMS cannot schedule a PSB into the control region. The message --n/a-- indicates that the minor does not apply to the region the major selected.

CALL
Displays user parameter information for the first region displayed.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Minor of IMS region majors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format:</td>
<td>cCALL</td>
</tr>
</tbody>
</table>

H in column 1 displays the I/O PCB in hexadecimal and character format. CALL displays the call function code, the first 60 bytes of the current PCB, and up to 60 bytes of the I/O area. DB calls also show segment search arguments.
This command does not apply to the control region or non-dependent regions.

The following figure shows the output of a typical CALL command.

CDMB
Displays the name of the most recently referenced database (DMB).

<table>
<thead>
<tr>
<th>Type:</th>
<th>Minor of IMS region majors</th>
</tr>
</thead>
</table>

This command does not apply to the control region.
DBCTL Region Minor Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPCB</td>
<td>Displays the name of the current PCB.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRM</td>
<td>Displays the current CICS terminal id for this DBCTL thread. This information is retrieved from the CICS EXEC Interface Block in the CICS address space running the DBCTL thread.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRN</td>
<td>Displays the CICS transaction name for the current thread.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTSK</td>
<td>Displays the current CICS task number for this DBCTL thread. This information is retrieved from the CICS EXEC Interface Block in the CICS address space running the DBCTL thread. The task number can be used in a CICS operator command to cancel the unit of work for such reasons as a looping PSB or a deadly embrace situation.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBT</td>
<td>Displays the current total of all DL/I database calls performed since the application program (PSB) was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DGN</td>
<td>Displays the number of database get next (GN) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DGU</td>
<td>Displays the number of database Get Unique (GU) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLET</td>
<td>Displays the number of database delete (DLET) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region or Fast Path regions.
### DBCTL Region Minor Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DPCB</strong></td>
<td>Displays the name of the most recently referenced database PCB.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>DRCS</strong></td>
<td>Displays the size of the dependent inter-region communication area (DIRCA).</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>ETIC</strong></td>
<td>Displays the elapsed wait time for an intent conflict in seconds.</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
<tr>
<td><strong>ETIO</strong></td>
<td>Displays the elapsed time for database I/O in seconds.</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
<tr>
<td><strong>ETLK</strong></td>
<td>Displays the elapsed time for PI locking in seconds.</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
<tr>
<td><strong>ETPL</strong></td>
<td>Displays the elapsed wait time for pool space in seconds.</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
<tr>
<td><strong>ETSP</strong></td>
<td>Displays the elapsed time for the schedule process in seconds.</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
<tr>
<td><strong>GHN</strong></td>
<td>Displays the number of database get hold next (GHN) calls performed since the application program was last scheduled into the region.</td>
<td>Minor of IMS region majors.</td>
</tr>
<tr>
<td><strong>GHNP</strong></td>
<td>Displays the number of database get hold next within parent (GHNP) calls performed since the application program was last scheduled into the region.</td>
<td>Minor of IMS region majors.</td>
</tr>
</tbody>
</table>

This command does not apply to the control region, other non-dependent regions, or Fast Path regions.

The size is in bytes (decimal).

This command does not apply to the control region or other non-dependent regions.
This command does not apply to the control region, other non-dependent regions, or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GHU</td>
<td>Displays the number of database get hold unique (GHU) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors.</td>
</tr>
</tbody>
</table>

This command does not apply to the control region, other non-dependent regions, or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GNP</td>
<td>Displays the number of database get next within parent (GNP) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors.</td>
</tr>
</tbody>
</table>

This command does not apply to the control region, other non-dependent regions, or Fast Path regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSPB</td>
<td>Displays HSSP private area buffer pool byte statistics.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors.</td>
</tr>
</tbody>
</table>

Displays HSSP private area buffer pool statistics for the first region displayed by the major command. The display fields are:

- **Area**: Displays the DEDB area that HSSP is processing.
- **Alloc**: Displays the total number of bytes that were allocated for the private area buffer pool.
- **Used**: Displays the number of bytes used by the private area buffer pool.
- **%Used**: Displays the percentage of bytes used in the private area buffer pool.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IHLD</td>
<td>Displays the number of IRLM or PL locks held.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INUM</td>
<td>Displays the number of database I/Os.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of RGN majors and their aliases.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISRT</td>
<td>Displays the number of database insert (ISRT) calls performed since the application program was last scheduled into the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>
This command does not apply to the control region, other non-dependent regions, or Fast Path regions.

<table>
<thead>
<tr>
<th>LOCK</th>
<th>Displays PI or IRLM locks currently held by a dependent region.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

The following figure shows an example of the LOCK command display.

### FIGURE 14. LOCK command display

<table>
<thead>
<tr>
<th>RGND</th>
<th>MPP00131</th>
<th>MPP00121</th>
<th>MPP00132</th>
<th>MPP00150</th>
</tr>
</thead>
<tbody>
<tr>
<td>lock</td>
<td>Subsys</td>
<td>Workunit</td>
<td>PSBname</td>
<td>Tx/Rg ID</td>
</tr>
<tr>
<td>+</td>
<td>IMSA</td>
<td>CICS321</td>
<td>ACCNT010</td>
<td>1</td>
</tr>
<tr>
<td>+</td>
<td>IMSA</td>
<td>CICS321</td>
<td>UPDCUST1</td>
<td>2</td>
</tr>
</tbody>
</table>

The fields in the LOCK display are:

- **Subsys**: Displays the name of the subsystem that holds the lock.
- **Workunit**: Displays the name of the job that holds the lock.
- **PSBname**: Displays the program specification block (PSB) associated with the lock.
- **Tx/Rg ID**: Displays the transaction name or region number (if DBCTL is being monitored) associated with the lock.
- **Lterm ID**: Displays the logical terminal name associated with the lock.
- **Status**: Displays the status of the job (workunit) holding the lock and indicates whether it is an owner or a waiter.

These statuses are:

- **EX**: Exclusive control
- **UP**: Update
- **RD**: Read
- **SH**: Share
- **ER**: Erase
Owners and waiters

**WAT**  Workunit is waiting for access.

**OWN**  Workunit owns the lock.

**DB/Area or area name**
Displays the database name.

**Token**  Displays the relative byte address/relative block number (PI only).

**DCB**  Displays the DCB number within the named DMB (PI only).

**n/a**  Displays if data are not available because there is more than one IRLM and the secondary IRLM is running on a different CPU.

Only those locks held by the first region displayed by the major command are shown. For example, in Figure 14 on page 202, the lock for region MPP00131 is the only one displayed. To display the locks for the other dependent regions, use RGNL.

<table>
<thead>
<tr>
<th>PLAN</th>
<th>Displays the DB2 plan ID for the current thread.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PLSY</th>
<th>Displays the number of DB2 commits during this schedule.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PSBN</th>
<th>Displays the PSB name of a thread if one is currently scheduled.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

If there is not a thread in progress, OMEGAMON displays **--none--**. This command does not apply to non-dependent regions.

<table>
<thead>
<tr>
<th>PSTA</th>
<th>Displays the PST address.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>REPL</th>
<th>Displays the number of database replace (REPL) calls made since the application was activated.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

This command does not apply to the control region, other non-dependent regions, or Fast Path regions.
This ID is the same as the PST number and the region number the IMS DC monitor refers to. This command does not apply to non-dependent regions.

The region type can be BMP, Control, DLS, DBRC, Fast Path, HSSP-BMP, IRLM, or DBCTL.

SALL displays the subsystem name, current IMS/DB2 interface status, the language interface token (LIT), resource translation table (RTT) name, interface control module name, and the error option specified.

The current IMS/DB2 interface status is one of the following:

- **cre-thrd**: Create thread is in progress.
- **SQL-call**: SQL call is in progress.
- **ph1-sync**: Phase 1 commit is in progress.
- **ph2-sync**: Phase 2 commit is in progress.
- **trm-thrd**: Thread is terminating.
- **conn**: DB2 subsystem is connected to control region.
- **not-con**: DB2 subsystem is not connected to control region.
- **not-def**: DB2 subsystem is not defined to control region.
- **usable**: There is no connection to DB2 subsystem due to an error detected in subsystem, unavailable resources, or the connection was never established.

An IMS dependent region can only be active with one DB2 subsystem at a time.
**DBCTL Region Minor Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| SETO    | Displays high speed sequential processing options for a region.  
**Type:** Minor of IMS region majors  
SETO displays information that has been specified in the SETO control card in the DFSCTL dataset. The display shows the following information:  

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbname</td>
<td>Displays the database name specified in the SETO control card in the DFSCTL dataset.</td>
</tr>
<tr>
<td>PCB</td>
<td>PCB name specified in the SETO control card in the DFSCTL dataset.</td>
</tr>
</tbody>
</table>
| #I/C  | Displays the following image copy information:  
1 One image copy.  
2 Dual image copy.  
0 No image copy.  
I/C-Opt | Displays the following image copy options:  
CONTINUE Specifies that the program continue if the image copy cannot complete.  
1ABEND Specifies that the program abend if the image copy cannot complete for one dataset.  
2ABEND Specifies that the program abend if the image copy cannot complete for two datasets. |
| Area  | Displays the area.  
STAT  | Displays the current status of the IMS region.  
**Type:** Minor of IMS region majors  
STAT displays the current status of the IMS region.  

SETR displays the processing areas of DEDBs to which an application program is restricted during scheduling. The application program can only access data in the DEDB within the area or areas specified:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| dbname | Displays the database name.  
PCB | Displays the PCB name.  
Area | Displays the area.  

The output fields in the SETR display are:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| STAT    | Displays the current status of the IMS region.  
**Type:** Minor of IMS region majors  
STAT displays the current status of the IMS region.  

---

**DBCTL Regions 205**
The possible status values are:

**Region Inactive**
- **Idle** No application is scheduled in this region.

**Scheduler Waits**
- **Wt-Intnt** Waiting because of intent conflict.
- **Wt-DMB** Waiting for DMB to load.
- **Wt-Mover** Waiting for block mover.
- **Wt-PSB** Waiting for PSB to load.

**IWAITs**
- **Wt-Plenq** IWAITing because of program isolation enqueue conflict.
- **Wt-DL/I** IWAITing for DL/I processing.
- **Wt-DISP** IWAITing in IMS dispatcher.
- **Wt-LATCH** IWAITing for a latch.
- **Wt-TERM** IWAITing in termination.
- **IWAIT** Other IWAIT.

**Execution States**
- **Ex-DL/I** Active in DL/I processing,ｵlation enqueue conflict.
- **Ex-Term** Active in termination.
- **Ex-Drgn** Active in IMS dependent region.
- **Wt-IRLM** Wait for IRLM lock conflict.
- **Ex-Abend** Executing in ABDUMP manager. In this execution state, cancelling a dependent region could cause a U0113 abend.
- **Ex-Lum** Executing in LU 6.2 manager. In this execution state, cancelling a dependent region could cause a U0113 abend.

**Fast Path Status**
- **Wt-FxBuf** Waiting for a fixed buffer.
- **Wt-OCLth** Waiting for the OPEN/CLOSE LATCH.
- **Wt-DmLth** Waiting for the DMAC LATCH.
- **Wt-MSLth** Waiting for the MSDB LATCH.
- **Wt-DEOwn** Waiting for ownership within a DEDB.
**Fast Path Status**

- **Wt-OBA** Waiting for OBA interlock.
- **Wt-SYLock** Waiting for the SYNC LOCK.
- **Ex-Sync** Active in SYNC POINT processing.

**DB2 Status**

- **Cre-Thrd** Create thread in progress.
- **Ph1-Sync** Phase 1 syncpoint in progress.
- **Ph2-Sync** Phase 2 syncpoint in progress.
- **SQL-Call** SQL® call in progress.
- **Trm-Thrd** Terminate thread in progress.

This command does not apply to non-dependent regions.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TOKN</strong></td>
<td>Displays the 8-byte DBCTL thread recovery token in hexadecimal format.</td>
</tr>
<tr>
<td><strong>WTDE</strong></td>
<td>Displays the number of waits for DEDB buffers.</td>
</tr>
<tr>
<td><strong>WTTE</strong></td>
<td>Displays the number of waits in test enqueues.</td>
</tr>
<tr>
<td><strong>WTUE</strong></td>
<td>Displays the number of waits in update and enqueues.</td>
</tr>
</tbody>
</table>

If the PST selected by the RGN major is not a DBCTL thread, **--n/a--** displays.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>WTDE</strong></td>
<td>Displays the number of waits for DEDB buffers.</td>
</tr>
<tr>
<td><strong>WTTE</strong></td>
<td>Displays the number of waits for DEDB buffers.</td>
</tr>
<tr>
<td><strong>WTUE</strong></td>
<td>Displays the number of waits for DEDB buffers.</td>
</tr>
</tbody>
</table>

**MVS-related region minor commands**

This section describes the MVS-related region minor commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AENV</strong></td>
<td>Displays graphical environmental data about a given region.</td>
</tr>
</tbody>
</table>

AENV displays graphical environmental data about a given region over both the last interval and the last 20 intervals.
When the major command selects more than one region, AENV only operates on the first region it selects; it ignores the rest.

The example below shows the output of a sample AENV command for the IMSPROD control region.

<table>
<thead>
<tr>
<th>RGNC</th>
<th>BMPGPRP1 Ssys</th>
<th>System Data-</th>
<th>Short 0.1.2.3.4.5.6.7.8.9.0</th>
<th>Long 0.1.2.3.4.5.6.7.8.9.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>aenv</td>
<td></td>
<td>+ TCB Time(%)</td>
<td>25.3</td>
<td>-----&gt; . . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ SBB Time(%)</td>
<td>9.7</td>
<td>-&gt; . . . . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ PAGE-ins/s</td>
<td>3.8</td>
<td>------&gt; . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ PAGE-outs/s</td>
<td>2.8</td>
<td>------&gt; . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ I/Os/sec</td>
<td>7.2</td>
<td>-&gt; . . . . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ SU/s/seg</td>
<td>131.4</td>
<td>---&gt; . . . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ Working Set</td>
<td>1400K</td>
<td>---&gt; . . . . . . . . .</td>
</tr>
<tr>
<td>+</td>
<td></td>
<td>+ Pg-in/CPU-s</td>
<td>12.5</td>
<td>---&gt; . . . . . . . . .</td>
</tr>
</tbody>
</table>

The graph on the left represents the last OMEGAMON cycle. The graph on the right represents a running average of the previous 20 OMEGAMON cycles. The last column of the display indicates trends using the following symbols:

+  The trend is upward.
-  The trend is downward.
(blank)  No change.

**ASID**  Displays the MVS address space ID of the region.

**Type:**  Minor of IMS region majors

The address space ID displays in hexadecimal.

MVS numbers the regions beginning with the *MASTER* scheduler at ASID 0001. As MVS starts regions, it assigns them an ASID number, which it uses as an internal reference.

**CPU**  Displays the TCB CPU time for the current job step for the region in seconds.

**Type:**  Minor of IMS region majors

Use the CPU command with the .R suffix to show the CPU rate during the last OMEGAMON cycle.

**DISP**  Displays the type of MVS dispatching algorithm being used for a region.

**Type:**  Minor of IMS region majors
The algorithm types are:

- MTW (mean-time-to-wait)
- ROTATE
- TIME-SLC
- FIXED

The IEAIPSnn member of SYS1.PARMLIB defines and controls domains.

<table>
<thead>
<tr>
<th>DOM#</th>
<th>Displays the current domain number of the region with which it is associated.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

Ordinarily, MVS never swaps out IMS regions, but some users modify their IMS systems to allow it.

<table>
<thead>
<tr>
<th>DPRT</th>
<th>Displays the CPU dispatching priority of the region in both decimal and hexadecimal.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FIXF</th>
<th>Displays the fixed frame count when a region is swapped out.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FMCT</th>
<th>Displays the number of frames a region is using in main storage.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

**Note:** For MVS/ESA, expanded storage is included.

When a region executes under MVS, portions of the program can be in main storage and other temporarily inactive portions can be on disk (on a PAGE or SWAP dataset). A unit of main storage (4096 bytes) is a frame. The unit of disk storage that holds one frame is a slot.

The frame count is zero when the region is swapped out. Instead of displaying zero, OMEGAMON displays the last frame count (and marks it with an S).

Regions can be resident and still show an S for FMCT. This usually means that the region is page-stolen down to zero.

<table>
<thead>
<tr>
<th>FXFR</th>
<th>Displays the number of frames of real storage a region is using that are fixed and cannot move to disk (as slots) to make room for other regions.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

One frame equals 4096 bytes.
DBCTL Region Minor Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HUIC</strong></td>
<td>Displays the highest unreferenced interval count for the region.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>IODP</strong></td>
<td>Displays the I/O dispatching priority for the region.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>IOJ</strong></td>
<td>Displays the I/O counts for the region.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>JPIC</strong></td>
<td>Displays the job common area page-in counts.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>JPUI</strong></td>
<td>Displays the job private area page-in counts.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>JPUO</strong></td>
<td>Displays the job private area page-out counts.</td>
<td>Minor of IMS region majors</td>
</tr>
<tr>
<td><strong>JSTA</strong></td>
<td>Displays the job status indicator for the region.</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

A low number for HUIC indicates that the system is paging heavily and that there is a high demand for real storage frames.

Use the IOJ command with the .R suffix to show the I/O rate during the last OMEGAMON cycle.

Use with .R to display the common area page-in rate.

Use with .R to display the private area page-in rate.

Use with .R to display the private area page-out rate.

The following figure illustrates the JSTA command display.
DBCTL Regions

The job status indicator (I/O*NSW in the example above) contains three fields in the following format:

```
   aaabccc
```

**aaa** Dispatchability:
- **CPU** Region is CPU dispatchable.
- **WAT** Region is waiting.
- **DLY** Region is delayed.
- **I/O** Region has performed I/Os but is currently not CPU dispatchable.

**b** Transaction flag:
- * Region is in an MVS transaction.
- ( ) Region is not in an MVS transaction.

**ccc** Location:
- **RES** Region is resident.
- **NSW** Region is resident and nonswappable.
- **LSW** Region is logically swapped.
- **SWP** Region is swapped out.

<table>
<thead>
<tr>
<th>NVSC</th>
<th>Displays the number of non-VIO slots the region uses.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

These are the slots on a PAGE dataset used for portions of the program.

The example below shows the number of non-VIO slots each dependent region uses.

```
   RGNB Bmprgn01 Bmprgn02 Bmprgn03
   fmct  40 S  30  10 <frame count>
   wkst  160K S 120K 40K <working set size>
   fxfr  10  5 <fixed frames>
   nvsc  40  33  14 <non-VIO (program)slots>
   vsc   5  <VIO slots>
```

<table>
<thead>
<tr>
<th>PERF</th>
<th>Displays the performance group number of a region.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

The PERF and DOM# minor commands display the performance group number and domain number of a region. The SRM manages performance in the MVS operating system by swapping regions in and out according to defined rules. The SRM uses service units to track the amount of computer resources a region is using. These service units are a composite of CPU time used, I/Os performed, and main storage occupancy for the region. After examining a region’s activity, different parts of the SRM recommend whether
IMS should swap the region in or out. Usually, IMS marks all regions non-swappable unless the user took some action to circumvent this.

The IEAIPSc member of SYS1.PARMLIB defines and controls performance groups.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROC</td>
<td>Displays the current procedure step name for the region.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSTI</td>
<td>Displays a summary of parameters and data relating to storage isolation.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

The following example shows the output of a typical PSTI command.

<table>
<thead>
<tr>
<th>RGNS</th>
<th>BMPRGN01</th>
</tr>
</thead>
<tbody>
<tr>
<td>psti</td>
<td>Private: Working Page-ins</td>
</tr>
<tr>
<td>+</td>
<td>Maximum Set Size /Elap-sec</td>
</tr>
<tr>
<td>+</td>
<td>Target 60K -none-</td>
</tr>
<tr>
<td>+</td>
<td>Actual 72K 0.13</td>
</tr>
<tr>
<td>+</td>
<td>Minimum 40K -none-</td>
</tr>
</tbody>
</table>

**Note:** OMEGAMON labels the /CPU-sec field as /Elap-sec when it runs in internal monitoring.

In this example, the BMPRGN01 region is in a performance group which has a minimum of 40K and a maximum of 100K established as storage isolation limits. BMPRGN01’s actual current working set size is 72K, but the current target size storage isolation has set is 60K.

The actual value for page-ins/Elap-second is currently 0.13; -none- indicates the user is not fencing for page-ins. The ------ value indicates there is no such thing as target page-ins.

Only one PSTI display at a time can be active on an OMEGAMON display. If you attempt to display data from two different jobs at the same time, PSTI does not complete initialization.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEQN</td>
<td>Displays the sequence number of the region on the CPU dispatching queue.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>
The following figure shows the CPU dispatching queue sequence number of each dependent region.

<table>
<thead>
<tr>
<th>Region</th>
<th>CPU Dispatching Queue Sequence Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGNB</td>
<td>BMPRGN01</td>
</tr>
<tr>
<td>cpu</td>
<td>144.23</td>
</tr>
<tr>
<td>cpu .R</td>
<td>0.038</td>
</tr>
<tr>
<td>sbt</td>
<td>11.23</td>
</tr>
<tr>
<td>sbt .R</td>
<td>0.009</td>
</tr>
<tr>
<td>tcpu</td>
<td>155.46</td>
</tr>
<tr>
<td>tcpu .R</td>
<td>0.047</td>
</tr>
<tr>
<td>disp</td>
<td>MTW</td>
</tr>
<tr>
<td>seqn</td>
<td>18</td>
</tr>
</tbody>
</table>

**SRBT**
- Displays the SRB CPU time of the current job step for the region in seconds.
- **Type:** Minor of IMS region majors

**STEP**
- Displays the current step name for the region.
- **Type:** Minor of IMS region majors

The example below shows the output of a typical STEP command.

<table>
<thead>
<tr>
<th>Region</th>
<th>CPU Dispatching Queue Sequence Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>RGNB</td>
<td>BMPRGN01</td>
</tr>
<tr>
<td>asid</td>
<td>(0C) 12</td>
</tr>
<tr>
<td>dom#</td>
<td>1 1</td>
</tr>
<tr>
<td>perf</td>
<td>6 6</td>
</tr>
<tr>
<td>proc</td>
<td>GO</td>
</tr>
<tr>
<td>step</td>
<td>REGION</td>
</tr>
</tbody>
</table>

**SUAL**
- Displays all service units for the period.
- **Type:** Minor of IMS region majors

All service units = SUCP + SUIO + SUMS.

**SUCP**
- Displays the CPU service units for this period.
- **Type:** Minor of IMS region majors

Use with .R to display service units per second during the last interval.

**SUJO**
- Displays the I/O service units for this period.
- **Type:** Minor of IMS region majors

Use with .R to display service units per second during the last interval.
### DBCTL Region Minor Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SUMS</strong></td>
<td>Displays the main storage occupancy service units for this period.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of IMS region majors</td>
</tr>
</tbody>
</table>

Use with `.R` to display service units per second during the last interval.

| **TCPU** | Displays the total CPU utilization (TCB + SRB) for the region. |
| **Type:** | Minor of IMS region majors |

This should equal the sum of the CPU and SRBT commands.

You can use the commands CPU, SRBT and TCPU with the `.R` rate operator to display the percentage of the CPU that this particular region is using. Percentages are displayed as decimal values; for instance, 2.7% is displayed as .027. This percentage value assumes that 100 percent is available from each CPU in the system. For a 3081 dyadic processor or a 3033MP which contains two CPUs, this figure could, in theory, be as high as 200 percent.

| **TMTR** | Displays the time since the region began. |
| **Type:** | Minor of IMS region majors |

Transaction refers to an MVS rather than a CICS transaction. An MVS transaction begins every time the performance group changes. This field displays the time since the last transaction began. If the performance group changes during execution of the job, the TMTR reflects the time since that change. For instance, it may measure only the time since a STEP change. Otherwise, it measures the total job time.

If you execute the MVS operator command SET IPS, IMS does not reset the times of transactions which are swapped out at the time of execution until they are swapped in again.

| **TWSS** | Displays the target working set size in K (1024 bytes) for any fenced region. |
| **Type:** | Minor of IMS region majors |

This command is for sites using the MVS feature called storage isolation. If the maximum target working set size is seen, MAXIMUM appears.

| **VSC** | Displays the number of VIO slots the region used. |
| **Type:** | Minor of IMS region majors |

Virtual I/O (VIO) is a method of using virtual memory for temporary files.
DBCTL Regions Minor Commands

Active means the time since any CPU was last expended.

The working set size is calculated as follows:

\[ \text{WKST} = 4 \times \text{FMCT}. \]

The value is zero when the region is swapped out. Instead of displaying the zero value, OMEGAMON displays the last working set size (and marks it with an S). Regions can be resident and still show an S for WKST. This usually means that the region is page-stolen down to zero.

The following figures show several typical region major and minor commands.

**FIGURE 15. Region major command with several minor commands**

<table>
<thead>
<tr>
<th>RGNB</th>
<th>BMPRGN01</th>
<th>BMPRGN02</th>
<th>BMPRGN03</th>
</tr>
</thead>
<tbody>
<tr>
<td>jsta</td>
<td>CPU*RES</td>
<td>I/O*RES</td>
<td>WAT*SWP</td>
</tr>
<tr>
<td>ioj</td>
<td>24.1</td>
<td>12.1</td>
<td>0</td>
</tr>
<tr>
<td>iodp</td>
<td>(04)212</td>
<td>(63)99</td>
<td>(63)99</td>
</tr>
<tr>
<td>tmtr</td>
<td>3:20 MN</td>
<td>2:10 MN</td>
<td>1:15 MN</td>
</tr>
<tr>
<td>wait</td>
<td>1 SEC</td>
<td>1:23 MN</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 16. Region major command with service unit and page rate minor commands**

<table>
<thead>
<tr>
<th>RGNB</th>
<th>BMPRGN01</th>
<th>BMPRGN02</th>
<th>BMPRGN03</th>
</tr>
</thead>
<tbody>
<tr>
<td>sucp</td>
<td>50</td>
<td>182</td>
<td>883</td>
</tr>
<tr>
<td>suio</td>
<td>130</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>sums</td>
<td>27</td>
<td></td>
<td></td>
</tr>
<tr>
<td>suai</td>
<td>180</td>
<td>293</td>
<td>883</td>
</tr>
<tr>
<td>huic</td>
<td>4</td>
<td>12</td>
<td>17</td>
</tr>
<tr>
<td>jpui.R</td>
<td>0.2</td>
<td>0.7</td>
<td>5.1</td>
</tr>
<tr>
<td>jpuo.R</td>
<td>1.0</td>
<td>2.2</td>
<td>2.1</td>
</tr>
<tr>
<td>jpci.R</td>
<td>1.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter overview

This section describes OMEGAMON commands that display information about the following IMS internal and external resources:

- data management blocks (DMBs)
- program specification blocks (PSBs)
- IMS pool statistics
- VSAM database pool statistics
- ISAM/OSAM buffer pool statistics
- Fast Path resource commands

Chapter contents

Database Management Block Commands ........................................ 218
PSB Commands ........................................................................... 228
Pool Commands .......................................................................... 233
IMS Logging Analysis ................................................................. 254
Fast Path Resource Commands .................................................... 260
Extended Recovery Facility (XRF) Support ..................................... 266
Introduction

The database management block (DMB) is an internal IMS control block which defines an IMS database. IMS knows each database by an 8-character name you specify during the IMS system generation process in the DBO= parameter of the database macro.

Note: This manual uses the term DMB as a synonym for database.

DMB immediate command

<table>
<thead>
<tr>
<th>DBIO</th>
<th>Displays each IMS database and associated DDName with current I/O rates and EXCP counts. This information is not available to Fast Path databases.</th>
</tr>
</thead>
</table>

Type: Immediate

You can filter the display by setting a variable. The variables available are

- **OIDBTYP** Database type
  - Maximum of 8 characters
  - Valid values are HDAM, HIDAM, HISAM, HSAM, SSAM, DEDB, MSDB, INDEX

- **OIDBORG** Database organization
  - Maximum of 4 characters
  - Valid values are OSAM, VSAM

- **OIDBPAT** Database name pattern
  - Maximum of 8 characters.
  - Use an asterisk for wild characters. If an asterisk is followed by a space, the remainder of the 8-character name is wild.
OIDBST  DMB status
  Maximum of 16 characters
  Valid values are:
  '/DBR ACTIVE'
  '/DBD ACTIVE'
  'WAIT ON DMB POOL'
  'STOP PENDING'
  'RESTART PEND'
  'RECOVERY NEEDED'
  'DYN ALLOC ERROR'
  'DMB STOPPED'
  'DMB NOT LOADED'
  'DMB NOT FOUND'
  'DMB NOT ALLOC'
  'DMB LOCKED'
  'DMB AVAILABLE'
  'DMB ACTIVE'
  'DATA AREA EMPTY'
  'AREA STOPPED'
  'AREA OPEN'
  'AREA NOT OPEN'
  'AREA I/O ERROR'
  'USABLE'
  'UNUSABLE'

You must use single quotes to enclose the literal, because these literal values have imbedded spaces.

Once a variable has been set, it will stay set for the duration of your session or until it is changed.

To set a variable, issue the following command starting in column 1.

. VAR S &variable value
The variable name must be preceded by an ampersand (&). To disable a variable, set the variable value to asterisk (*).

**DMB major commands**

This section describes the major commands available to select DMBs.

**DMBA**

<table>
<thead>
<tr>
<th>DBD Name</th>
<th>DMB Status</th>
<th>Type/Org</th>
<th>DD Name</th>
<th>EXCP Count</th>
<th>EXCP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>BA$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BAHDLO01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BB$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BBHDLO01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BC$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BCCHDL01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BD$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BDHDM001</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BE$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BEHDM001</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BF$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BFHDM001</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BG$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BGHDLO01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BH$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BHHDLO01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>BI$HDLO0</td>
<td>DMB AVAILABLE</td>
<td>HDAM/OSAM</td>
<td>BIHDLO01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD1</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD2</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD4</td>
<td>DMB ACTIVE</td>
<td>HDAM/VSAM</td>
<td>LOAN</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DI21PAR1</td>
<td>DMB AVAILABLE</td>
<td>HISAM/VSAM</td>
<td>DI21PART</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1I</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB2</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB4</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD3</td>
<td>AREA NOT OPEN</td>
<td>DEDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB3</td>
<td>AREA OPEN</td>
<td>DEDB</td>
<td>DFSIVD32</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Type:** Major

I/O errors consist of read errors, write errors, hot I/O standby errors, DBRC user errors, and DBRC permanent errors.

**DMBI**

<table>
<thead>
<tr>
<th>DBD Name</th>
<th>DMB Status</th>
<th>Type/Org</th>
<th>DD Name</th>
<th>EXCP Count</th>
<th>EXCP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBFSAMD1</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD2</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD4</td>
<td>DMB ACTIVE</td>
<td>HDAM/VSAM</td>
<td>LOAN</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DI21PAR1</td>
<td>DMB AVAILABLE</td>
<td>HISAM/VSAM</td>
<td>DI21PART</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1I</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB2</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB4</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD3</td>
<td>AREA NOT OPEN</td>
<td>DEDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB3</td>
<td>AREA OPEN</td>
<td>DEDB</td>
<td>DFSIVD32</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Type:** Major

IMS only loads DMBs into memory if it has marked them resident or when the database they represent is actually open.

**DMBL**

<table>
<thead>
<tr>
<th>DBD Name</th>
<th>DMB Status</th>
<th>Type/Org</th>
<th>DD Name</th>
<th>EXCP Count</th>
<th>EXCP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBFSAMD1</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD2</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD4</td>
<td>DMB ACTIVE</td>
<td>HDAM/VSAM</td>
<td>LOAN</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DI21PAR1</td>
<td>DMB AVAILABLE</td>
<td>HISAM/VSAM</td>
<td>DI21PART</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB1I</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB2</td>
<td>DMB NOT LOADED</td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB4</td>
<td>DMB AVAILABLE</td>
<td>MSDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DBFSAMD3</td>
<td>AREA NOT OPEN</td>
<td>DEDB</td>
<td></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>IVDPB3</td>
<td>AREA OPEN</td>
<td>DEDB</td>
<td>DFSIVD32</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Type:** Major
The DMBN major command displays only non-recoverable databases that have been opened.

DMBP/n Selects all DMBs matching a pattern.

The .SPT command sets the pattern. See the .SPT command for information about setting patterns. If DMBP does not find a /n value in the argument field, it uses the first pattern supplied with the last .SPT command.

DMBU Selects DMBs that are unusable.

Use the SCHD minor command to display the status of a DMB and see why IMS considers the DMB unusable.

In some IMS systems, IMS does not find a large number of DMBs at IMS initialization, and OMEGAMON includes them in the unusable display. If you do not care that these DMBs are unavailable, you can exclude them from the DMBU display. To do this, enter an X in the argument field (DMBUX). The following example shows the database management block major commands.

```
DMBA BE2PCUST BE3ORDER BE3ORDRX BE3PARTS BE3PSID1 DI21PART  
schd  Availble  Active  Not-Fnd Availble Availble DMB-Stop  
DMBI  BE3ORDER  
DMBL  BE3ORDER BE3ORDRX  
.SPT  B**P*  <--- Enter pattern - Current pattern = 'B**P*'
DMBP  BE2PCUST BE3PARTS BE3PSID1  
DMBU  BE3ORDRX DI21PART  
schd  Not-Fnd DMB-Stop
```

The DMBN major command on line 1 of the above example shows that the IMS system has six defined DMBs. The SCHD minor command on line 2 displays the scheduling status of each DMB. The DMBI major selected the only DMB currently in memory (because it is the only one currently active). The DMBL major selected the two DMBs the user-specified list requested. The DMBP major selected those DMBs whose names match the pattern which the preceding .SPT immediate command set. The DMBU command selected the only two DMBs which are unschedulable, and the SCHD minor shows the reason for each.
**DMB minor commands**

The following minor commands display additional information about database management blocks (DMBs).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCS</td>
<td>Displays the highest database access intent defined for the resident (or currently open) DMB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

ACCS can display the following access intents:

- **Rd-Only** Read-only access.
- **Read** Read access.
- **Update** Update access.
- **Xclusive** Exclusive access.
- **Unknown?** Any other type appears as unknown.

If the DMB is not currently in virtual memory, ------- appears.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADMB</td>
<td>Displays the address of the DMB block in storage.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

If the block is not currently in storage, ------- appears.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASP</td>
<td>Displays the number of control area splits which have occurred for a VSAM database.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

This is the sum of control area (CA) splits for all VSAM datasets which make up the database since the last time VSAM reorganized the database. Before a database opens, the display reads **Not-Open**. If the DMB does not represent a VSAM database, **Not-VSAM** appears. If the DMB represents a Fast Path database, **FASTPATH** appears.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CISP</td>
<td>Displays the number of control interval splits which occurred for a VSAM database.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

This is the sum of control interval (CI) splits for all VSAM datasets which make up the database since the last time VSAM reorganized the database. Before a database opens, the display reads **Not-Open**. If the DMB does not represent
a VSAM database, Not-VSAM appears. If the DMB represents a Fast Path database, FASTPATH appears.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRES</td>
<td>Displays the residency status of the DMB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

The DMB status displays as **Resident**, **In-memry**, or **Not-in. In-memry** implies the DMB is currently in memory but IMS did not mark it resident.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSZE</td>
<td>Displays the size of a DMB in decimal bytes.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

This number should be the same as that on an ACBGEN listing.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID#</td>
<td>Displays either the DL/I ddname or the Fast Path area name for databases which have incurred I/O errors.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

**Format:** ID#nn

ID# accepts the following values in its argument field:

- **1-99** Represents the number of relative I/O errors.
- **A** Displays all I/O errors.

The default value is 1.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IR#</td>
<td>Displays the RBA of the I/O error in the database.</td>
</tr>
</tbody>
</table>

**Type:** Minor of DMB majors

**Format:** IR#nn

IR# accepts the following values in its argument field:

- **1-99** Represents the number of relative I/O errors.
- **A** Displays all I/O errors.

The default value is 1.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT#</td>
<td>Displays the type of I/O error in the database.</td>
</tr>
</tbody>
</table>
Database Management Block Commands

**Type:** Minor of DMB majors

**Format:** `IT#nn`

IT# accepts the following values in its argument field:

- **1-99** Represents the number of relative I/O errors.
- **A** Displays all I/O errors.

The default value is 1.

IT# displays the following error types:

- **RD-err** DMB has a read error.
- **WRT-err** DMB has a write error.
- **IOT-err** DMB has a hot standby error.
- **DBRC-err** DMB has a DBRC user error.
- **PRM-err** DMB has a DBRC permanent error.

**ODDN** Displays information about non-VSAM databases.

**Type:** Minor of DMB majors

ODDN displays the following information about non-VSAM databases:

- ddname
- blocksize
- volume serial number (volser)
- unit address of the device on which the database resides

If the database is on multiple volumes, the volser and unit address of each volume appear.

**PB#** Displays the PSB of the database which has a dynamic backout error.

**Type:** Minor of DMB majors

**Format:** `PB#nn`

PB# accepts the following values in its argument field:

- **1-99** Represents the number of relative dynamic backout errors.
- **A** Displays all dynamic backout errors.

The default value is 1.
### PD#
 Displays the Julian date when a dynamic backout error occurred against the database.

**Type:** Minor of DMB majors  
**Format:** PD#nn  
PD# accepts the following values in its argument field:   
- **1-99** represents the number of relative dynamic backout errors.  
- **A** displays all dynamic backout errors.  
The default value is 1.

### PSBC
 Displays the number of PSBs that are currently active and using this DMB.

**Type:** Minor of DMB majors

### PT#
 Displays the time when the dynamic backout error occurred against the database.

**Type:** Minor of DMB majors  
**Format:** PT#nn  
PT# accepts the following values in its argument field:   
- **1-99** represents the number of relative dynamic backout errors.  
- **A** displays all dynamic backout errors.  
The default value is 1.  
The time is in the format hh:mm:ss.

### RCOV
 Displays whether a database is recoverable or non-recoverable.

**Type:** Minor of DMB majors
RCOV displays the following information:

<table>
<thead>
<tr>
<th>Not-Fnd</th>
<th>DMB could not be found at IMS startup, and the database is not usable.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Rcov</td>
<td>The database is opened and non-recoverable.</td>
</tr>
<tr>
<td>Not-Opnd</td>
<td>The database is not opened. RCOV cannot determine the status.</td>
</tr>
<tr>
<td>Recovrbl</td>
<td>The database is recoverable.</td>
</tr>
</tbody>
</table>

RCOV displays non-recoverable databases only when they have been opened.

<table>
<thead>
<tr>
<th>SCHD</th>
<th>Displays the current scheduling status for this DMB.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of DMB majors</td>
</tr>
</tbody>
</table>

The possible messages are:

<table>
<thead>
<tr>
<th>/DBD-Act</th>
<th>A /DBD command is currently active, and the DMB is not available.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMB-Stop</td>
<td>A user issued a /STOP DATABASE command against the DMB.</td>
</tr>
<tr>
<td>DMB-Lock</td>
<td>A user issued a /LOCK DATABASE command against the DMB.</td>
</tr>
<tr>
<td>Not-Fnd</td>
<td>A user defined the DMB but IMS did not find it in ACBLIB at startup.</td>
</tr>
<tr>
<td>WF-Pool</td>
<td>The DMB is coming into memory, but is waiting for DMB pool space.</td>
</tr>
<tr>
<td>Active</td>
<td>A PSB which uses the DMB is currently scheduled (DMB is not necessarily open).</td>
</tr>
<tr>
<td>Available</td>
<td>The database is available but not currently in use.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SPD#</th>
<th>Displays the subpool ID of each OSAM dataset in the database.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of DMB majors</td>
</tr>
<tr>
<td>Format:</td>
<td><strong>SPD#nn</strong></td>
</tr>
</tbody>
</table>

The variable *nn* specifies an OSAM dataset number. This number is required. If the dataset is not assigned to a specific subpool, SPD# displays *default*.

<table>
<thead>
<tr>
<th>STAT</th>
<th>Displays the scheduling status of the database.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of DMB majors</td>
</tr>
</tbody>
</table>
STAT is an alias of the SCHD minor command.

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Displays the database type.</th>
</tr>
</thead>
</table>

**Type:** Minor of DMB majors

The possible types are:

- **VSAM** The database is stored as a VSAM file.
- **OSAM** The database is stored as an OSAM file.
- **DEDB MSDB** The database is a Fast Path DEDB.
- **MSDB** The database is a Fast Path MSDB.
- **Not in** The DMB control block is not currently in memory, so this information is not available.

<table>
<thead>
<tr>
<th>VDDN</th>
<th>Displays information about VSAM databases.</th>
</tr>
</thead>
</table>

**Type:** Minor of DMB majors

VDDN displays the following information:

- ddnames which make up the database
- number of CA and CI splits
- number of dataset extents
- whether VSAM writecheck is currently on or off
- CI-size of the data component
- CI-size of the index component (if any)
- volume serial number
- unit address of the device on which the dataset resides

If the database is on multiple volumes, VDDN displays the volser and unit address of each volume.

OMEGAMON only produces a display for the first major item you select.

If the DMB does not represent a VSAM database, VDDN displays

>> Not a VSAM Database >>.

If the database that DMB represents is not open, VDDN displays

>>Database not open<<.
PSB Commands

Introduction

The PSB is an internal IMS control block that defines a site’s application programs. IMS knows each PSB by an 8-character name you choose during the IMS system generation process in the PSB= parameter of the APPLCTN macro. This manual uses the term PSB as a synonym for program or application.

PSB major commands

The following major commands are available for selecting PSBs.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSBA</td>
<td>Selects all PSBs.</td>
</tr>
<tr>
<td>PSBI</td>
<td>Selects all PSBs that are currently loaded into virtual storage.</td>
</tr>
<tr>
<td>PSBL</td>
<td>Selects all specified PSBs.</td>
</tr>
<tr>
<td>PSBP/n</td>
<td>Selects all PSBs matching a pattern.</td>
</tr>
</tbody>
</table>

IMS only loads non-resident PSBs (those that the user did not mark resident at IMS generation time) into memory when IMS actually schedules the program they represent. The PSBI major selects all those PSBs which are currently loaded into virtual storage. This includes all resident PSBs and any PSBs that may be loaded into the PSB pool even though they are not currently scheduled.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSBS</td>
<td>Selects all PSBs that are currently running in at least one dependent region.</td>
</tr>
</tbody>
</table>

The .SPT command sets the pattern. See the .SPT command for information about setting patterns. If PSBP does not find a /n value, it uses the first pattern supplied with the last .SPT command.
These are a subset of PSBs which the PSBI major command selects.

<table>
<thead>
<tr>
<th>PSBU</th>
<th>Selects all PSBs that are unusable.</th>
</tr>
</thead>
</table>

**Type:** Major

Use the SCHD minor command to display the status of a PSB and see why IMS considers a PSB unusable. (See “PSB minor commands” for details.)

In some IMS systems, IMS does not find a large number of PSBs at IMS initialization. OMEGAMON includes these PSBs in the unusable display. If you do not care that these PSBs are unavailable, you can exclude them from the PSBU display. To do this, enter an X in the argument field (PSBUX).

**PSB minor commands**

The following minor commands are available to display additional information about program specification blocks (PSBs).

<table>
<thead>
<tr>
<th>APSB</th>
<th>Displays the address of the PSB if the block is in storage.</th>
</tr>
</thead>
</table>

**Type:** Minor of PSB majors

**NotInMem** appears if the block is not in storage.

<table>
<thead>
<tr>
<th>DB#</th>
<th>Displays DMBs associated with a selected PSB.</th>
</tr>
</thead>
</table>

**Type:** Minor of PSB majors

**Format:** DB#nnn

You can invoke the DB# minor command in two different ways:

- To request a multi-line display of all the DMBs associated with the first PSB selected, enter an A in the label field without nnn as shown in the following example:

<table>
<thead>
<tr>
<th>PSBL</th>
<th>PE4CPPUR PE4CODEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ad#</td>
<td>Database Highest Intent</td>
</tr>
<tr>
<td>+</td>
<td>BE3PSID1 Update</td>
</tr>
<tr>
<td>+</td>
<td>BE3PARTS Update</td>
</tr>
<tr>
<td>+</td>
<td>BE3ORDER No Intent</td>
</tr>
<tr>
<td>+</td>
<td>BE3ORDRX No Intent</td>
</tr>
</tbody>
</table>

In this example, PSB PE4CPPUR has two databases associated with it that have a highest intent of update. DB# does not produce any output for the PE4CODEL PSB. If the intent list for the PSB is not in memory, the DB# minor produces the message:

>> Intent list for PSB PE4CPPUR is not in memory <<
To display the name of the \textit{nnn}th DMB associated with the PSB, leave the label field blank and supply a number in the argument field as shown in the following example:

\begin{verbatim}
PSBL  PE4CPPUR PE4CODEL
  db# 1 BE3PSID1 BE2PCUST
  db# 2 BE3PARTS BE3ORDER
  db# 3 BE3ORDER BE3PARTS
  db# 4 BE3ORDRX BE3ORDRX
  db# 5 Max=4   BE3PSID1
\end{verbatim}

If the PSB you specify is not in memory, \textbf{NotInMem} appears.

<table>
<thead>
<tr>
<th>IRES</th>
<th>Displays the residency attribute of the intent list associated with a PSB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of PSB majors</td>
</tr>
</tbody>
</table>

The intent list can be either resident or non-resident. If non-resident, the intent list can be either in memory or out. \textbf{IRES} displays these three states as \textbf{Resident}, \textbf{In-memry}, and \textbf{Not-in}.

<table>
<thead>
<tr>
<th>LANG</th>
<th>Displays the program language associated with a PSB.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of PSB majors</td>
</tr>
</tbody>
</table>

The language is one of the following: ASM/CBL, Pascal, or PL/1. If the PSB is not currently in memory, the display shows \textbf{NotInMem}.

<table>
<thead>
<tr>
<th>PDLS</th>
<th>Displays the number of bytes used in the DPSB pool.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of PSB majors</td>
</tr>
</tbody>
</table>

When you choose the LSO=S option, each active PSB requires space in the DPSB pool in the DLISAS address space.

If the PSB is not currently in the pool, \textbf{NOT-IN} appears. If you did not choose the LSO=S option, PDLS displays the value \textbf{DLS inac}. If all PSBs are in CSA, PDLS displays \textbf{ALL-CSA}.

<table>
<thead>
<tr>
<th>PDRA</th>
<th>Displays the virtual address of the PSB directory entry (PDIR) associated with this PSB.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of PSB majors</td>
</tr>
</tbody>
</table>

This address displays in hexadecimal.
This address displays in hexadecimal.

The PSZE minor command gives the size of the PSB, but the actual amount of space it needs from the pool depends on the residency attributes of the PSB, its intent list, and whether the PSB is to be parallel scheduled. (A resident PSB with a resident intent list that is not parallel scheduled does not require any space from the PSB pool, because IMS pre-assigns all necessary memory from resident pools.)

If the PSB is parallel scheduled, IMS makes a copy of the PDIR in the PSB pool. PPUS displays the result, including the size of the PDIR.

To find out how much total PSB pool storage is currently in use, multiply the size PPUS reports by the current number of the PSB’s schedulings (see the PSBC minor command).

IMS marks a PSB RESIDENT, DOPT, or neither. If not RESIDENT, the PSB can be either currently in memory or out. PRES displays these states as Resident, DOPT-in, DOPT-out, In-memry, and Not-in. This has nothing to do with whether or not IMS schedules the PSB, because RESIDENT PSBs are always in memory, and non-RESIDENT PSBs can be left in the PSB pool even if they are not currently in use.

For a PSB that cannot be parallel scheduled, this number cannot be greater than 1.

Zero means that the PSB is not currently scheduled.

This number should be the same as that found on an ACBGEN listing.
**PSB Commands**

**Type** Minor of PSB majors

Program type is either online or batch.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCHD</td>
<td>Displays the current scheduling status for this PSB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of PSB majors.

SCHD displays the current scheduling status for this PSB. The possible messages are:

- **DMB-Stop** The user issued a /STOP DATABASE command against a PSB databases.
- **PSB-Lock** The user issued a /LOCK PROGRAM command against the PSB.
- **PSB-Stop** The user issued a /STOP PROGRAM command against the PSB, or the PSB abended.
- **Not-Fnd** The PSB was defined but not found in ACBLIB at IMS startup.
- **Scheduled** The PSB is currently scheduled (at least once).
- **Schdlble** The PSB is usable but not currently in use.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STAT</td>
<td>Displays the current status of the PSB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of PSB majors

STAT is an alias of the SCHED minor command.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STYP</td>
<td>Displays the type of scheduling associated with this PSB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of PSB majors

Scheduling type is either parallel or serial.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRCE</td>
<td>Displays the status of the trace option for this PSB.</td>
</tr>
</tbody>
</table>

**Type:** Minor of PSB majors

Trace option status is one of the following: off (Off), on (Trace On), or on with the compare option (TraceCmp).
Pool Commands

Introduction

A set of major commands are available to display the characteristics and
utilization of various IMS pools. The following major commands display
statistics about the indicated IMS pools.

Automated operator interface pool

This command displays information about the automated operator interface
pool (AOIP).

<table>
<thead>
<tr>
<th>AOIP</th>
<th>Stores messages, commands and responses instead of the message queues.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

The AOIP is available only with IMS 6.1 and above.

**FIGURE 17. Typical AIOP major and SIZE minor output**

<table>
<thead>
<tr>
<th>AOIP</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>Total Alloc Free High</td>
</tr>
<tr>
<td>+ Size</td>
<td>81920 56033 25887 62587</td>
</tr>
<tr>
<td>+ Percent</td>
<td>na 68.4 31.6 76.4</td>
</tr>
<tr>
<td>+ Blocks</td>
<td>na 17 1 na</td>
</tr>
<tr>
<td>+ Numbers</td>
<td>na 1024 25887 na</td>
</tr>
<tr>
<td>+ Sml size</td>
<td>na 4096 25887 na</td>
</tr>
<tr>
<td>+ Big size</td>
<td>na 3296 25887 na</td>
</tr>
<tr>
<td>+ Avg size</td>
<td>na 3296 25887 na</td>
</tr>
</tbody>
</table>

Field descriptions:

- **Total** Total size of the AOIP pool.
- **Alloc** Currently allocated pool storage.
- **Free** Currently available pool storage.
- **High** Highest amount of pool storage allocated as sampled by OMEGAMON.
- **Size** Amount of storage as represented by a specific column in the display.
OMEGAMON provides the following command to select the communications external subsystem (CESS) pool.

**CESS**

 Displays statistics about the use of the CESS pool.

**Type:** Major

CESS displays information about the communications external subsystem (CESS) pool. It displays:

- the total size of the pool in bytes
- the free space available within the pool in bytes
- the current utilization percentage
- the usage high water mark (the largest amount used since IMS startup)
- the number of free blocks
- the size of the largest free block

The last two items indicate the extent to which IMS can fragment the pool.
See the following figure for a sample display.

<table>
<thead>
<tr>
<th>CESS</th>
<th>Total Size in bytes = 4096</th>
<th>Free space in bytes = 4096</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Utilization = 0.00%</td>
<td>Usage Highwater mark = 0</td>
</tr>
<tr>
<td>+ Number of Free blocks = 1</td>
<td>Largest free block = 4096</td>
<td></td>
</tr>
</tbody>
</table>

ALOC
Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.

Type: Minor of CESS

EXTD
Displays extended storage pool information.

Type: Minor of CESS

FREE
Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.

Type: Minor of CESS

BUFS
Displays allocation and usage statistics for buffers in the CESS pool.

Type: Minor of CESS

Main work area pool
OMEGAMON provides the major command WKAP to display information about the IMS main work area pool.

WKAP
Displays size, free space, utilization, and the usage high water mark of the general work pool.

Type: Major
The following example shows the results of the WKAP command.

<table>
<thead>
<tr>
<th>WKAP</th>
<th>Total Size in bytes = 39056</th>
<th>Free space in bytes = 37152</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Utilization = 4.87%</td>
<td>Usage Highwater mark= 7112</td>
</tr>
</tbody>
</table>

**ALOC** | Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.

**FREE** | Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.

**Database pool commands**

OMEGAMON provides the following commands to select and display information about the database pools.

**DBWP** | Displays statistics about the utilization of the database work pool.

**Type:** Major

DBWP displays statistics about the utilization of the database work pool. It displays:

- the total size of the pool in bytes
- the free space available within the pool in bytes
- the current utilization percentage
- the usage high water mark (the largest amount used since IMS startup)
- the number of free blocks
- the size of the largest free block.

The last two items indicate the extent to which IMS can fragment the pool.
See the following figure for a sample display.

<table>
<thead>
<tr>
<th>Pool</th>
<th>Total Size in bytes</th>
<th>Free space in bytes</th>
<th>Utilization</th>
<th>Usage Highwater mark</th>
<th>Number of Free blocks</th>
<th>Largest free block</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBWP</td>
<td>8192</td>
<td>8192</td>
<td>.00%</td>
<td>0</td>
<td>1</td>
<td>8192</td>
</tr>
</tbody>
</table>

**ALOC**
Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.

**Type:** Minor of DBWP

The dump is in hexadecimal.

This dump provides size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

**FREE**
Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.

**Type:** Minor of DBWP

The dump is in hexadecimal.

This dump provides size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

**DMPL**
Displays statistics about the database management block (DMB) pool.

**Type:** Major

**ALOC**
Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.

**Type:** Minor of DMPL

The dump is in hexadecimal.

This dump provides size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

**FREE**
Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.

**Type:** Minor of DMPL
The dump is in hexadecimal.

This dump provides size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEstokeeptrackoftheirmemory.

<table>
<thead>
<tr>
<th>SIZE</th>
<th>Displays pool storage usage statistics for a specific storage pool.</th>
</tr>
</thead>
</table>

**Type:** Minor of DMPL

**FIGURE 18. DMPL major and SIZE minor commands**

<table>
<thead>
<tr>
<th>DMPL size</th>
<th>Utilization, Current: 5.2% Highest: 5.2%</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Size Total Alloc Active Free Res Res-act G-total</td>
<td></td>
</tr>
<tr>
<td>+ 16384</td>
<td>856</td>
</tr>
<tr>
<td>+ 15528</td>
<td>1040</td>
</tr>
<tr>
<td>+ 17424</td>
<td></td>
</tr>
<tr>
<td>+ Percents</td>
<td>5.2</td>
</tr>
<tr>
<td>+ Blocks</td>
<td>94.7</td>
</tr>
<tr>
<td>+ Numbers</td>
<td>5.9</td>
</tr>
<tr>
<td>+ Sml Size</td>
<td></td>
</tr>
<tr>
<td>+ Big Size</td>
<td>856</td>
</tr>
<tr>
<td>+ Avg Size</td>
<td>15528</td>
</tr>
</tbody>
</table>

The fields are:

- **Current** Percentage of the storage pool space currently allocated.
- **Highest** Highest percentage of the storage pool space ever allocated.
- **Total** Total storage within the DMB pool.
- **Alloc** Amount of storage allocated within the pool.
- **Active** Amount of storage associated with active DMBs.
- **Free** Amount of unallocated storage within the pool.
- **Res** Storage outside of the pool associated with resident DMBs.
- **Res-act** Storage associated with resident active DMBs.
- **G-total** Total IMS storage for DMBs; a sum of the Total and Res values.
- **Percents** Percentage of storage as represented by a specific column in the display.
- **Blocks** These fields display information about the pieces of contiguous storage in the pool:
  - **Numbers** Number of blocks for a specific column in the display.
  - **Sml size** Smallest current block size for a specific column in the display.
  - **Big size** Largest current block size for a specific column in the display.
  - **Avg size** Average current block size for a specific column in the display.
VSAM database buffer pool statistics

<table>
<thead>
<tr>
<th>DBVS</th>
<th>Displays statistics relating to individual VSAM buffer subpools.</th>
</tr>
</thead>
</table>

**Type:** Immediate  
**Format:** cDBVSnn  
- c Enter S to display summary information for VSAMn subpools.  
- nn VSAM subpool ID. If you do not specify an ID, DBVS displays data for all subpools.

DBVS displays the subpool number, the buffer size for the subpool, the number of buffers allocated, and whether the buffers or IOBs are page-fixed. Buffer statistics similar to those from the /DISPLAY POOL DBAS command also appear, such as retrieves by key, records altered, and VSAM reads. If a the subpool ID has specific databases assigned to it, DBVS also lists the subpool ID and its assigned databases.

The following example shows the summary displayed with SDBVS:

<table>
<thead>
<tr>
<th>SDBVS</th>
<th>VSAM Subpool Summary</th>
<th>20 buffers totaling 63488 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+ Retrieves by RBA</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>+ Records altered</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>+ VSAM reads</td>
<td>3231</td>
</tr>
<tr>
<td></td>
<td>+ Found in pool</td>
<td>2396</td>
</tr>
<tr>
<td></td>
<td>+ Buffers in error</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>+ Hit ratio</td>
<td>74.1%</td>
</tr>
<tr>
<td></td>
<td>+ VSAM strings active</td>
<td>1</td>
</tr>
</tbody>
</table>

DBVS displays whether the buffers are in Hiperspace™, and it displays LSR pool information. The following figure shows an example of the display:

**FIGURE 19. Display VSAM buffer subpool statistics**

<table>
<thead>
<tr>
<th>DBVS03 VSAM Subpool # 3</th>
<th>4 buffers of 2048 bytes each</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Pool id: POL1</td>
<td>Pool number: 0</td>
</tr>
<tr>
<td>+ Subpool type: Index</td>
<td>Subpool number within pool: 2</td>
</tr>
<tr>
<td>+ Buffers are not page-fixed</td>
<td>Blocks are not page-fixed</td>
</tr>
<tr>
<td>+ Retrieves by RBA</td>
<td>10</td>
</tr>
<tr>
<td>+ Records altered</td>
<td>0</td>
</tr>
<tr>
<td>+ VSAM reads</td>
<td>531</td>
</tr>
<tr>
<td>+ Found in pool</td>
<td>396</td>
</tr>
<tr>
<td>+ Buffers in error</td>
<td>0</td>
</tr>
<tr>
<td>+ Hit ratio</td>
<td>74.5%</td>
</tr>
<tr>
<td>+ Buffers are backed by hiperspace</td>
<td></td>
</tr>
<tr>
<td>+ Data bases assigned to this subpool:</td>
<td>HSSP01FX(2) KD#IGW03(1) KD#MEM01(3) VDBSINDEX(1)</td>
</tr>
</tbody>
</table>

The number in parentheses after the database name denotes the dataset as specified in DFSVSMMxx. For example, the number 2 in parentheses after HSSP01FX in Figure 19 on page 239 means that the second dataset statement in the database is assigned to the subpool.
**VSAM database subpool statistics**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSUB</td>
<td>Displays VSAM subpool statistics.</td>
</tr>
</tbody>
</table>

**Type:** Major  
**Format:** VSUB n n n . . .

The variable n is the number of each subpool you want to analyze. The minors of VSUB are described below.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BALT</td>
<td>Displays the number of logical records altered.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSIZ</td>
<td>Displays the size of the buffers in this subpool.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CERR</td>
<td>Displays the number of error buffers currently in the subpool.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWRT</td>
<td>Displays the number of writes forced by VSAM.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

This is the number of non-user-initiated writes.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GETS</td>
<td>Displays the number of VSAM GET calls issued.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISES</td>
<td>Displays the number of logical inserts to ESDS.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISKS</td>
<td>Displays the number of logical inserts to KSDS.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IWRT</td>
<td>Displays the number of user-initiated writes by VSAM.</td>
</tr>
</tbody>
</table>

**Type:** Minor of VSUB
<table>
<thead>
<tr>
<th>Type:</th>
<th>Minor of VSUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NBFR</td>
<td>Displays the number of buffers in this subpool.</td>
</tr>
<tr>
<td>RKEY</td>
<td>Displays the number of VSAM retrieves by key.</td>
</tr>
<tr>
<td>RRBA</td>
<td>Displays the number of VSAM retrieves by RBA.</td>
</tr>
<tr>
<td>SCBF</td>
<td>Displays the number of schedule buffer calls issued.</td>
</tr>
<tr>
<td>SFND</td>
<td>Displays the number of successful buffer finds.</td>
</tr>
<tr>
<td>TERR</td>
<td>Displays the largest number of error buffers during this execution of IMS.</td>
</tr>
<tr>
<td>TIO</td>
<td>Displays the total number of VSAM I/Os in this subpool.</td>
</tr>
<tr>
<td>TWRT</td>
<td>Displays the total number of writes.</td>
</tr>
<tr>
<td>VRDS</td>
<td>Displays the number of reads performed by VSAM.</td>
</tr>
</tbody>
</table>
**ISAM/OSAM buffer pool commands**

The IMS systems programmer defines various numbers and sizes of subpools that are used for database I/O. Specifying the size and number of buffers in a subpool has a direct effect on the performance of database I/O.

<table>
<thead>
<tr>
<th>SPAL</th>
<th>Displays database names and dataset numbers of those databases assigned to specific subpools.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
<tr>
<td>Format:</td>
<td><strong>SPAL</strong> cccc</td>
</tr>
<tr>
<td></td>
<td>Specifies a subpool ID. If you do not specify a subpool ID, SPAL displays all databases and dataset numbers that have been assigned to a buffer subpool in the DFSVSMnn member.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DBOS</th>
<th>Displays statistics about ISAM/OSAM buffer subpools.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td><strong>DBOS</strong> nn</td>
</tr>
<tr>
<td></td>
<td>Specifies a subpool. If you omit the argument nn, data appears for all subpools, and OMEGAMON produces a summary consisting of totals for all subpools at the end.</td>
</tr>
</tbody>
</table>

DBOS displays the subpool number, the buffer size for the subpool, the number of buffers allocated, and whether the buffers or buffer prefixes were page-fixed. Buffer statistics similar to those from the /DISPLAY POOL DBAS command appear (locate calls, locates found in pool, and read requests). If a subpool ID has specific databases assigned to it, DBOS also lists the subpool ID and its assigned databases. The hit ratio field is a ratio of locates found in the pool to locate calls.

The following figure shows the results of the DBOS command.

**FIGURE 20. ISAM/OSAM buffer pool statistics**

<table>
<thead>
<tr>
<th>Subpool id: AYCD</th>
<th>4 buffers of 2048 bytes each</th>
</tr>
</thead>
<tbody>
<tr>
<td>Locate Calls =</td>
<td>36033</td>
</tr>
<tr>
<td>Locates found in pool =</td>
<td>3835</td>
</tr>
<tr>
<td>Single block writes =</td>
<td>0</td>
</tr>
<tr>
<td>Records altered =</td>
<td>510</td>
</tr>
<tr>
<td>Blocks written by purge =</td>
<td>0</td>
</tr>
<tr>
<td>Buffers locked by errors =</td>
<td>0</td>
</tr>
<tr>
<td>Hit ratio =</td>
<td>5.0%</td>
</tr>
<tr>
<td>Data bases assigned to this subpool</td>
<td>AYCD01FX(1) DB$HFS01(1) DB$MDL03(2) SDBHINDEX(4)</td>
</tr>
</tbody>
</table>
**Note:** The number in parentheses after the database name denotes the dataset as specified in the DBDGEN. For example, the number 2 in parentheses after DB$MDL03 in Figure 20 on page 242 means that the second dataset statement in the database is assigned to the subpool.

<table>
<thead>
<tr>
<th>ALOC</th>
<th>Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of DBOS</td>
</tr>
</tbody>
</table>

The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

<table>
<thead>
<tr>
<th>FREE</th>
<th>Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of DBOS</td>
</tr>
</tbody>
</table>

The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

**ISAM/OSAM subpool commands**

The OSUB command and its minors provide detailed statistics about the OSAM subpool.

<table>
<thead>
<tr>
<th>OSUB</th>
<th>Displays detailed OSAM subpool statistics.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>Format:</td>
<td><strong>OSUB n n n . . .</strong></td>
</tr>
</tbody>
</table>

The variable n is the number of each subpool you wish to analyze.

Following are the OSUB minors.

<table>
<thead>
<tr>
<th>ABWP</th>
<th>Displays the average number of blocks written per purge operation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of OSUB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AWAT</th>
<th>Displays the average number of waits per subpool I/O request.</th>
</tr>
</thead>
</table>
### Pool Commands

A subpool I/O request is a read, write, or purge.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSZE</td>
<td>Displays the size of buffers in this subpool.</td>
</tr>
<tr>
<td>EIO</td>
<td>Displays the number of I/O errors.</td>
</tr>
<tr>
<td>LBFR</td>
<td>Displays the number of buffers locked due to I/O errors.</td>
</tr>
<tr>
<td>LFND</td>
<td>Displays the number of locates found in this pool.</td>
</tr>
<tr>
<td>NBUF</td>
<td>Displays the number of buffers in this pool.</td>
</tr>
<tr>
<td>PSZE</td>
<td>Displays the amount of storage being used for prefixes and subpool control blocks.</td>
</tr>
<tr>
<td>PURG</td>
<td>Displays the number of purge requests within the subpool.</td>
</tr>
<tr>
<td>PWRT</td>
<td>Displays the number of purge writes.</td>
</tr>
<tr>
<td>RIO</td>
<td>Displays the number of read I/O requests.</td>
</tr>
<tr>
<td>SEAR</td>
<td>Displays the number of buffers searched by locate.</td>
</tr>
</tbody>
</table>
Type: Minor of OSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWRT</td>
<td>Displays the number of single write requests.</td>
</tr>
</tbody>
</table>

Type: Minor of OSUB

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TLS</td>
<td>Displays number of buffers searched to find a buffer at level n.</td>
</tr>
</tbody>
</table>

Format: **TLSn**

The variable n is one of the following levels:

0  The buffer is available for use.
1  The buffer is empty.
2  A PST referenced the buffer.
3  A PST referenced the buffer, and the buffer is busy, because the PST is doing I/O.
4  A PST referenced the buffer and altered its contents.
5  Something other than a PST referenced the buffer and altered its contents.
6  A PST is using the buffer, but the PST did not alter the buffer’s contents.
7  A PST is using the buffer, and the PST altered the buffer’s contents.
8  The buffer is currently busy reading.

Format: **TLVn**

The variable n is one of the following levels:

0  The buffer is available for use.
1  The buffer is empty.
2  A PST referenced the buffer.
3  A PST referenced the buffer, and the buffer is busy, because the PST is doing I/O.
4  A PST referenced the buffer and altered its contents.
5  Something other than a PST referenced the buffer and altered its contents.
6  A PST is using the buffer, but the PST did not alter the buffer’s contents.
Sequential buffering (SBUF) information

<table>
<thead>
<tr>
<th>SBUF</th>
<th>Displays sequential buffering information.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Immediate</td>
</tr>
</tbody>
</table>

Sequential buffering is a buffering technique IMS can use to speed I/O processing for OSAM databases.

<table>
<thead>
<tr>
<th>Maximum storage</th>
<th>Current utilization</th>
<th>Current storage</th>
<th>HWM utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>76120</td>
<td>28.04%</td>
<td>21346</td>
<td>37.76%</td>
</tr>
</tbody>
</table>

A PST is using the buffer, and the PST altered the buffer’s contents.
The buffer is currently busy reading.
Field descriptions:

**Maximum storage**
The maximum amount of storage which can be used for sequential buffering. SBUF displays **-no max**- if there was no maximum value supplied in the IMS SBONLINE control card.

**Current utilization**
Percentage of maximum storage currently being used.

**Current storage**
Amount of storage currently being used for sequential buffering.

**HWM utilization**
Largest percentage of maximum storage used for sequential buffering.

**Fast path buffer pool commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPDB</td>
<td>Displays Fast Path buffer pool statistics.</td>
</tr>
</tbody>
</table>

**Type:** Major

**FIGURE 21. FPDB major and BSTS and SIZE minor commands**

<table>
<thead>
<tr>
<th>FPDB Fast Path Buffer Pool Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bsts</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Size</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>+</td>
</tr>
<tr>
<td>+</td>
</tr>
<tr>
<td>+</td>
</tr>
</tbody>
</table>

| BSTS Displays Fast Path buffer pool (FPDB) statistics. |

**Type:** Minor of FPDB

The statistics include:

**DBBF**
Total number of Fast Path buffers defined.

**DBFX**
Total number of Fast Path buffers defined as a cushion that allows for system and/or output thread usage.

**BSIZE**
VSAM control interval (CI) size of each Fast Path buffer. Possible CI sizes are: 512, 1024, 2048 and 4096 bytes and so on, up to 28K. Each buffer is the same size.

Also displayed are the number of regions waiting for a free Fast Path database buffer for the reading of a CI. This condition is commonly referred to as a
"hard luck wait". Don’t confuse this condition with a region’s waiting for a CI currently in use by another region, a PST, or an output thread.

<table>
<thead>
<tr>
<th>SIZE</th>
<th>Displays the Fast Path buffer pool allocation and usage sizes.</th>
</tr>
</thead>
</table>

**Type:** Minor of FPDB

Sizes are by number of CIs, buffers, bytes, or percentages, as follows:

<table>
<thead>
<tr>
<th>Size Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total or DBBF</strong></td>
<td>Total buffers in the Fast Path buffer pool.</td>
</tr>
<tr>
<td><strong>Unfix new PSTs</strong></td>
<td>Total number of Fast Path buffers defined that are not page-fixed for use by regions or by output threads. These buffers are available for the normal buffer allocations (NBA) for additional regions (IFP or MPP and BMP regions with NBA specified). If this number is less than the normal buffer allocation that the dependent region JCL specifies, the additional region abends.</td>
</tr>
<tr>
<td><strong>Pgfix SDEPS</strong></td>
<td>Number of page-fixed AREAs that are open and have a sequential dependent control interval (SDEP CI) defined. Each SDEP CI resides in a buffer. This buffer is acquired at AREA open. It is written back to DASD when it is full (another buffer replaces it for the next SDEP CI) or when the AREA is closed. Buffers are taken from the page-fixed available allocation, but when buffers for the current SDEP CI are newly acquired, they cause Fast Path to recalculate the boundary between page-fixed and unfixed buffers. If this boundary is adjusted, an unfixed buffer will be given to the page-fixed available allocation. The opening of AREAs with SDEPs defined ultimately reduces the number of unfixed buffers available for the normal buffer allocation (NBA parameter on the dependent region JCL) for new PSTs. In determining the total number of Fast Path buffers (DBBF parameter on the control region JCL), consider the largest number of concurrent open AREAs with SDEPs defined.</td>
</tr>
<tr>
<td><strong>Pgfix PST Use</strong></td>
<td>Buffers which dependent regions are currently using for normal buffer allocation (NBA) and overflow buffer allocation (OBA).</td>
</tr>
<tr>
<td><strong>Pgfix Avail</strong></td>
<td>Buffers allocated but not currently in use for dependent region NBA and OBA. Pgfix Avail also includes fixed buffers left (not allocated for dependent region NBA or OBA) for system or output thread usage, but not currently being used.</td>
</tr>
<tr>
<td><strong>Pgfix Int Q</strong></td>
<td>Number of buffers that were previously used and released, and are now in the intermediate queue, where they await return to the page-fixed available allocation. The output thread that writes the update back to DASD releases these buffers to the intermediate queue. Unmodified buffers that the application process used are released to the intermediate queue during PST sync point.</td>
</tr>
<tr>
<td><strong>Pgfix OTHR</strong></td>
<td>Buffers which are output threads currently writing back to DASD.</td>
</tr>
</tbody>
</table>
See Figure 21 on page 247 for a sample display.

**EPCB pool information**

<table>
<thead>
<tr>
<th>EPCB</th>
<th>Displays EPCB (extended PCB) pool information.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

**FIGURE 22. Typical EPCB major and SIZE minor output**

<table>
<thead>
<tr>
<th>EPCB size</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
</tr>
<tr>
<td>Size</td>
<td>81920</td>
</tr>
<tr>
<td>Percent</td>
<td>na</td>
</tr>
<tr>
<td>Blocks</td>
<td></td>
</tr>
<tr>
<td>Numbers</td>
<td>na</td>
</tr>
<tr>
<td>Sml size</td>
<td>na</td>
</tr>
<tr>
<td>Big size</td>
<td>na</td>
</tr>
<tr>
<td>Avg size</td>
<td>na</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SIZE</th>
<th>Displays the extended PCB pool allocation and usage sizes.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of EPCB</td>
</tr>
</tbody>
</table>

**PSB work pool commands**

The commands in this section display information about the PSB pool.

<table>
<thead>
<tr>
<th>PSBW</th>
<th>Displays statistics about the utilization of the PSB work pool.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
</tbody>
</table>

PSBW displays:
- the total size of the pool in bytes
- the free space available within the pool in bytes
- the current utilization percentage
- the usage high water mark (the largest amount used since IMS startup)
- the number of free blocks and the size of the largest free block.

The last two items indicate the extent to which IMS can fragment the pool. See Figure 23 on page 250 for a sample display.

<table>
<thead>
<tr>
<th>ALOC</th>
<th>Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Minor of PSBW</td>
</tr>
</tbody>
</table>
The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

| FREE | Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool. |
| Type: Minor of PSBW |

The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

FIGURE 23. IMS PSBW pool information

| PSBW | Total Size in bytes = 12288 | Free space in bytes = 10040 |
| + Utilization = 18.29% | Usage Highwater mark = 4496 |
| + Number of Free blocks = 1 | Largest free block = 10040 |

| PSPL | Displays PSB pool statistics. |
| Type: Major |

The minor commands SIZE and DSIZ show in detail how IMS utilizes pool storage.

| ALOC | Displays a dump of all the free-allocated queue elements (FAQEs) for allocated space within each pool. |
| Type: Minor of PSPL |

The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

| DSIZ | Displays pool private storage usage statistics for a DL1-SAS PSB storage pool. |
| Type: Minor of PSPL |
Pool Commands

### FREE

| Displays a dump of all the free-allocated queue elements (FAQEs) for free space within each pool.

**Type:** Minor of PSPL

The dump is in hexadecimal.

This dump provides the size and virtual addresses of each free and allocated area within the pool. Some pools, such as MFP, QBUF, and WKAP, do not use the FAQEs to keep track of their memory.

### REQU

| Displays pool request statistics, rates, and deltas for a specific pool.

**Type:** Minor of PSPL

The types of conflicts grouped under the REQU subparm OTHER are:

- SMB locked or stopped
- PSB locked or stopped
- database stopped
- PSB permanently bad

### SIZE

| Displays pool CSA storage usage statistics for a specific storage pool.

**Type:** Minor of PSPL

Figure 24 on page 251 shows an example of how to use the PSPL major command and its minors, SIZE and DSIZ. SIZE displays storage information for PSB pool (in CSA). DSIZ displays storage information for PSB space (in DLISAS).

**FIGURE 24. DSIZ and SIZE minors of PSPL**

<table>
<thead>
<tr>
<th>PSPL</th>
<th>Utilization, Current: 59.2%, Highest: 69.8% (CSA storage)</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>Size Total Alloc Active Free Res Res-act G-total</td>
</tr>
<tr>
<td>+ 65536</td>
<td>38832 12204 26704 1884 67420</td>
</tr>
<tr>
<td>+ Percents</td>
<td>59.2 18.6 40.7 2.7</td>
</tr>
<tr>
<td>Blocks</td>
<td>Numbers 14 2 4 1</td>
</tr>
<tr>
<td>+ Sml Size</td>
<td>336 3932 64 1876</td>
</tr>
<tr>
<td>+ Big Size</td>
<td>9520 8272 20928 1876</td>
</tr>
<tr>
<td>+ Avg Size</td>
<td>2773 6102 6676 1876</td>
</tr>
<tr>
<td>dsiz</td>
<td>Utilization, Current: 94.6%, Highest: 99.7% (DLISAS storage)</td>
</tr>
<tr>
<td>+ Size Total Alloc Active Free Res Res-act G-total</td>
<td></td>
</tr>
<tr>
<td>+ 368640</td>
<td>349064 145584 19576 58348 426988</td>
</tr>
<tr>
<td>+ Percents</td>
<td>94.6 39.4 5.3 13.6</td>
</tr>
<tr>
<td>Blocks</td>
<td>Numbers 13 2 3 1</td>
</tr>
<tr>
<td>+ Sml Size</td>
<td>5896 40696 1248 58340</td>
</tr>
</tbody>
</table>
The fields are:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current</td>
<td>Percentage of storage pool space currently allocated.</td>
</tr>
<tr>
<td>Highest</td>
<td>Highest percentage of storage pool space ever allocated.</td>
</tr>
<tr>
<td>Total</td>
<td>Total storage within the PSB pool.</td>
</tr>
<tr>
<td>Alloc</td>
<td>Amount of storage allocated within the pool.</td>
</tr>
<tr>
<td>Active</td>
<td>Amount of storage associated with active PSBs.</td>
</tr>
<tr>
<td>Free</td>
<td>Amount of unallocated storage within the pool.</td>
</tr>
<tr>
<td>Res</td>
<td>Storage outside of pool associated with resident PSBs.</td>
</tr>
<tr>
<td>Res-act</td>
<td>Storage associated with resident active PSBs.</td>
</tr>
<tr>
<td>G-total</td>
<td>Total IMS storage for PSBs; a sum of the Total and Res values.</td>
</tr>
<tr>
<td>Percent</td>
<td>The percentage of storage for a specific column in the display.</td>
</tr>
<tr>
<td>Blocks</td>
<td>These fields display information about the pieces of contiguous storage in the pool:</td>
</tr>
<tr>
<td>Numbers</td>
<td>The number of blocks for a specific column in the display.</td>
</tr>
<tr>
<td>Sml size</td>
<td>The smallest current block size for a specific column in the display.</td>
</tr>
<tr>
<td>Big size</td>
<td>The largest current block size for a specific column in the display.</td>
</tr>
<tr>
<td>Avg size</td>
<td>The average current block size for a specific column in the display.</td>
</tr>
</tbody>
</table>

OMEGAMON provides analysis about the partitioning of the PSB pool, with PSBs partitioned in the DLISAS address space, as well as common storage (CSA). The SIZE minor command of the PSPL command contains expanded information about the CSA PSB storage areas, and DSIZ shows information about DLISAS storage.

Program isolation enqueue commands

<table>
<thead>
<tr>
<th>PIEP</th>
<th>Displays information about PI enqueue pool utilization.</th>
</tr>
</thead>
</table>

Type: Major

The CORE parameter of the IMSCTF macro specifies the size of the PI enqueue pool at IMS generation time. Because the pool can grow, the user specifies both an increment and a maximum size.

IMS starts out with one increment of space and adds more increments as required until it reaches the maximum. PIEP displays the percentage of the theoretical maximum which is currently in use.
The PIEP command displays utilization relative to both current and maximum pool sizes. You can set the APIE exception to produce a warning when this utilization exceeds some threshold. The following example shows the result of the PIEP major command before initialization.

<table>
<thead>
<tr>
<th>PIEP</th>
<th>Maximum</th>
<th>Current</th>
<th>Used(bytes)</th>
<th>Unused(bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Size</td>
<td>16K</td>
<td>0K</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>+ Percent of maximum</td>
<td>100.00</td>
<td>.00</td>
<td>.00</td>
<td>100.00</td>
</tr>
<tr>
<td>+ Percent of current</td>
<td>-NA-</td>
<td>100.00</td>
<td>.00</td>
<td>.00</td>
</tr>
<tr>
<td>+ Pool increment =</td>
<td>2K</td>
<td>Avg. Length of Search =</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+ Maximum Search =</td>
<td>0</td>
<td>Request Rate/<strong>Initialized</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following example shows the result of the PIEP major command after initialization.

<table>
<thead>
<tr>
<th>PIEP</th>
<th>Maximum</th>
<th>Current</th>
<th>Used(bytes)</th>
<th>Unused(bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Size</td>
<td>16K</td>
<td>2K</td>
<td>8</td>
<td>2040</td>
</tr>
<tr>
<td>+ Percent of maximum</td>
<td>100.00</td>
<td>12.50</td>
<td>.04</td>
<td>99.95</td>
</tr>
<tr>
<td>+ Percent of current</td>
<td>-NA-</td>
<td>100.00</td>
<td>.39</td>
<td>99.60</td>
</tr>
<tr>
<td>+ Pool increment =</td>
<td>2K</td>
<td>Avg. Length of Search =</td>
<td>.33</td>
<td></td>
</tr>
<tr>
<td>+ Maximum Search =</td>
<td>1</td>
<td>Request Rate/Sec =</td>
<td>.03</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delta/Cycle =</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

After initialization, the Request Rate/Sec and Delta/Cycle fields might display blanks or zeros, depending on IMS activity.

**Save area prefix (SAP) pool command**

<table>
<thead>
<tr>
<th>SAPP</th>
<th>Displays number of dynamic and privileged SAPs allocated and currently in use, along with resulting utilization.</th>
</tr>
</thead>
</table>

Type: **Major**

This buffer pool contains a fixed number of buffers.

The following example shows information about the save area prefix (SAP) sets.

<table>
<thead>
<tr>
<th>SAPP</th>
<th>Dynamic SAPs allocated</th>
<th>Dynamic SAPs in use =</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Utilization =</td>
<td>.00%</td>
<td>Privileged SAPs =</td>
<td>10</td>
</tr>
</tbody>
</table>

A pre-assigned SAP is one that IMS permanently assigns to a certain function, and a privileged SAP is one that IMS can only use for privileged events.
**Introduction**

OMEGAMON provides IMS logging analysis.

**DASD logging analysis**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSYS</td>
<td>Displays an analysis of the DASD logging feature.</td>
</tr>
</tbody>
</table>

**Type:** Major

LSYS is a major command which selects information about the DASD logging feature. LSYS minor commands display the following information:

- data about the logging environment
- logging statistics
- OLDS and WADS dataset data

The following minor commands display detailed information about the DASD logging environment.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBUF</td>
<td>Displays OLDS buffer definition and usage statistics.</td>
</tr>
</tbody>
</table>

**Type:** Minor of LSYS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LENV</td>
<td>Displays DASD logging environment data.</td>
</tr>
</tbody>
</table>

**Type:** Minor of LSYS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTA</td>
<td>Displays DASD logging statistics.</td>
</tr>
</tbody>
</table>

**Type:** Minor of LSYS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLDS</td>
<td>Displays information about the active primary and secondary OLDS log datasets.</td>
</tr>
</tbody>
</table>

**Type:** Minor of LSYS

The OLDS command displays information about the currently active OLDS dataset. To display other OLDS datasets, enter the command OLDSnn where nn is the numeric suffix of the dataset you want to see.

If dual OLDS logging is in effect, the OLDS command displays information from both the primary and secondary datasets. To display only OLDS primary dataset information, enter **POLDS** where **P** (primary dataset) is in the label.
field (column 1). To display OLDS secondary dataset information only, enter **SOLDS** where **S** (secondary dataset) is in the label field (column 1).

<table>
<thead>
<tr>
<th>WADS</th>
<th>Displays information about the active primary and secondary WADS log datasets.</th>
</tr>
</thead>
</table>

**Type:** Minor of LSYS

The WADS command is similar to the OLDS command, but it displays information about the current WADS dataset. To display other WADS datasets, enter the command **WADSn** where **n** is the numeric suffix (0-9) of the dataset.

To display the primary WADS dataset, enter **PWADS** where **P** (primary dataset) is in the label field (column 1). To display the secondary WADS dataset, enter **SWADS** where **S** (secondary dataset) is in the label field (column 1).

**FIGURE 25. LSYS and its minor commands**

```
LSYS IMS/VS DASD Logging Environment and Statistics
lenv OLDS Logging = DUAL Auto Archive Limit = 1
  + OLDS Defined = 3 OLDS Active = 3
  + OLDS Stopped = 0 OLDS in ERROR = 0
  + WADS Logging = DUAL WADS Defined = 2
  + WADS In Use = 2 Spare WADS left = 0
------------------------------------------------------------------
                  Hardware assisted Log Compression/Expansion is being performed
lsta Totals Rates/Sec. Delta
+ Total Log Records 30927 .00 0
+ Total Log Blocks 236 .00 0
+ Write Ahead Requests 4065 .00 0
+ DC Waits for Write Ahead 0 .00 0
+ Output Buffer Waits 43 .00 0
+ Output Buffer Checkpoint Wait 15 .00 0
+ # System Checkpoints 26 n/a 0
+ EXCPVRs to the WADS 4190 .00 0
+ 2K Blocks Written to WADS 4915 .00 0
+ WRITEs to the OLDS 235 .00 0
+ READs from OLDS 0 .00 0
```
FIGURE 26. (Page 2 of 2) LSYS and its minor commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ODDS</td>
<td>Displays the OLDS datasets.</td>
</tr>
<tr>
<td>OBLB</td>
<td>Displays last block number in OLDS (logging end of file).</td>
</tr>
<tr>
<td>OBLK</td>
<td>Displays OLDS capacity in blocks.</td>
</tr>
<tr>
<td>OBLW</td>
<td>Displays latest block number written to the OLDS.</td>
</tr>
<tr>
<td>OBLZ</td>
<td>Displays OLDS block size.</td>
</tr>
</tbody>
</table>

**Online log datasets**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>DDNAME</th>
<th>Status</th>
<th>DSNAME</th>
<th>Unit address</th>
<th>Volume</th>
<th>Logical record length</th>
<th>Blocksize</th>
<th>I/O Count</th>
<th>I/O Rate</th>
<th>Number of BLOCKs</th>
<th>Starting BLOCK</th>
<th>Ending BLOCK</th>
</tr>
</thead>
<tbody>
<tr>
<td>olds</td>
<td>DFSOLP00</td>
<td>Current - Open</td>
<td>IMS.V130.OLP00</td>
<td>153</td>
<td>OMON22</td>
<td>22524</td>
<td>22528</td>
<td>235</td>
<td>.00 per second</td>
<td>600</td>
<td>235</td>
<td>600</td>
</tr>
<tr>
<td>wads</td>
<td>DFSWADS0</td>
<td>Current - Open</td>
<td>IMS.V130.WADS00</td>
<td>755</td>
<td>IMS120</td>
<td>2080</td>
<td>2080</td>
<td>4190</td>
<td>.00 per second</td>
<td>600</td>
<td>235</td>
<td>600</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Minor of ODDS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>--------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OBST</strong></td>
<td>Displays first block number in the OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OCIO</strong></td>
<td>Displays the current I/O count of the OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OLRL</strong></td>
<td>Displays OLDS logical record length.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OPDN</strong></td>
<td>Displays the primary OLDS ddname.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OPST</strong></td>
<td>Displays the status of the primary OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OPUA</strong></td>
<td>Displays the unit address of the primary OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OPVL</strong></td>
<td>Displays the volume of the primary OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ORDR</strong></td>
<td>Displays the OLDS sequence of use.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OSDN</strong></td>
<td>Displays the secondary OLDS ddname.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OSST</strong></td>
<td>Displays the status of the secondary OLDS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Write-ahead log datasets

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSUA</td>
<td>Displays the unit address of the secondary OLDS.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of ODDS</td>
</tr>
<tr>
<td>OSVL</td>
<td>Displays the volume of the secondary OLDS.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of ODDS</td>
</tr>
<tr>
<td>WDDS</td>
<td>Displays the WADS datasets in their order of use.</td>
</tr>
<tr>
<td>Type:</td>
<td>Major</td>
</tr>
<tr>
<td>WBLK</td>
<td>Displays the WADS block size.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of WDDS</td>
</tr>
<tr>
<td>WCIO</td>
<td>Displays the WADS I/O count.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of WDDS</td>
</tr>
<tr>
<td>WLRL</td>
<td>Displays the logical record length of the WADS dataset.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of WDDS</td>
</tr>
<tr>
<td>WSTA</td>
<td>Displays the status of the WADS.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of WDDS</td>
</tr>
<tr>
<td>WUCB</td>
<td>Displays the unit address of the volume containing the WADS.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of WDDS</td>
</tr>
</tbody>
</table>

The WDDs major command selects and displays the write-ahead datasets (WADS) in the order of their use.

The following minor commands display additional information about the write-ahead log datasets.
<table>
<thead>
<tr>
<th><strong>WUSE</strong></th>
<th>Displays use of the WADS.</th>
</tr>
</thead>
</table>

**Type:** Minor of WDDS

The use of the WADS is primary, secondary, or spare.

<table>
<thead>
<tr>
<th><strong>WVOL</strong></th>
<th>Displays the volume on which the WADS resides.</th>
</tr>
</thead>
</table>

**Type:** Minor of WDDS
Fast Path Resource Commands

Introduction

There are a number of major, minor, and immediate commands which analyze Fast Path resources.

Summary of Fast Path information

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSYS</td>
<td>Displays summary information about Fast Path in your system.</td>
</tr>
</tbody>
</table>

**Type:** Immediate

This immediate command displays the number of each type of Fast Path (IFP) region currently active.

Field descriptions:

- **IFPs**: Fast Path regions
- **MD**: Message-driven
- **UT**: Fast Path utility

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FSYS</td>
</tr>
<tr>
<td></td>
<td>IFPs (MD) active = 0</td>
</tr>
<tr>
<td></td>
<td>IFPs (UT) active = 0</td>
</tr>
<tr>
<td>+</td>
<td>BALG Msgs processed = 0</td>
</tr>
<tr>
<td>+</td>
<td>BALG Msgs queued = 0</td>
</tr>
<tr>
<td>+</td>
<td>BALG Msg Enq Rate = .00/sec</td>
</tr>
<tr>
<td></td>
<td>BALG Msg Deq Rate = .00/sec</td>
</tr>
</tbody>
</table>

This command also displays the total number of Fast Path messages based on all balancing groups (BALGs) queued and processed. The number of Fast Path messages queued equals the number processed plus the number of messages waiting to process.

The Fast Path message counts and rates are based on current Fast Path balancing group counters, and only represent current values. These values can change, because IMS clears Fast Path balancing group messages processed counters to zero when the last Fast Path region processing the application is stopped.

Data entry database areas

The following major commands select data entry database areas.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEDA</td>
<td>Selects all DEDB areas.</td>
</tr>
</tbody>
</table>

**Type:** Major
### Fast Path Resource Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DEDH</strong></td>
<td>Displays all DEDB areas that have an HSSP BMP active.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
<tr>
<td><strong>DEDL</strong></td>
<td>Displays all specified DEDB areas.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
<tr>
<td><strong>DEDU</strong></td>
<td>Displays all DEDBs that are closed or stopped.</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
</tbody>
</table>

A DEDB is unusable if it is closed or stopped.

| **DSME** | Displays Fast Path VSO DEDB area I/O rates for all DEDB areas allocated to dataspaces. By using the optional dataspace name, the list is limited to only those DEDB areas allocated to the specified dataspace. |
| **Type:** | Immediate |

This command applies to IMS 6.1 and above.

| **DSML** | Displays all the Fast Path VSO dataspaces allocated to the IMS system as well as datasource and DASD I/O statistics. Each datasource is allocated as a result of defining the database as a VSO database in the Database Recovery Control (RECON) dataset. |
| **Type:** | Immediate |

This command applies to IMS 6.1 and above.

| **DSPS** | Displays summary information about the datasource and the areas allocated to datasources. By using the optional datasource name, the display is limited to a single datasource. |
| **Type:** | Immediate |

This command applies to IMS 6.1 and above.
Fast Path Resource Commands

**DSPX**
Displays the number of dataspaces, number of Fast Path DEDB areas allocated to the dataspaces, the dataspaces with the highest and lowest dataspace I/O rate, and the areas with the highest and lowest I/O rate.

**Type:** Immediate

This command applies to IMS 6.1 and above.

**AINF**
Shows information about the selected DEDB area.

**Type:** Minor of data entry database majors

The information appears in the following format:

- **CL .. ..** Area closed.
- **OP .. ..** Area opened.
- **ST .. ..** Area stopped and closed.
- **.. SD ..** Area has sequential dependent segments defined.
- **.. SF ..** Area has sequential dependent segments defined and, during Fast Path synchronization point, there was not enough room in the sequential dependent portion of the area to insert a segment.
- **.. .. ER** Area has an I/O error and requires recovery.

**CIAB**
Shows the number of CIs in the root addressable portion of each unit of work.

**Type:** Minor of data entry database majors

**CIDO**
Shows the number of CIs in the dependent overflow portion of each unit of work.

**Type:** Minor of data entry database majors

**CIIF**
Shows the number of free CIs in independent overflow.

**Type:** Minor of data entry database majors

**CIIO**
Shows the number of CIs in independent overflow.

**Type:** Minor of data entry database majors
### Fast Path Resource Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIRA</td>
<td>Shows the number of CIs in the root addressable portion of the area.</td>
</tr>
<tr>
<td>CISF</td>
<td>Shows the number of free CIs in sequential dependent overflow.</td>
</tr>
<tr>
<td>CISO</td>
<td>Shows the number of CIs in sequential dependent overflow.</td>
</tr>
<tr>
<td>CISZ</td>
<td>Shows the size of a CI in this area.</td>
</tr>
<tr>
<td>CIUW</td>
<td>Shows the total number of CIs in this UOW.</td>
</tr>
<tr>
<td>CUOW</td>
<td>Shows the number of DEDB resource request conflicts.</td>
</tr>
</tbody>
</table>

**Type:** Minor of data entry database majors

Each conflict comprises two or more requests.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPUT</td>
<td>Shows the Fast Path utility currently active on the DEDB area.</td>
</tr>
<tr>
<td>COMPARE</td>
<td>Fast Path online DEDB area dataset compare (DBFUMMH0).</td>
</tr>
<tr>
<td>CREATE</td>
<td>Fast Path online DEDB area dataset create (DBFUMRI0).</td>
</tr>
<tr>
<td>DELETE</td>
<td>Fast Path online sequential dependent delete (DBFUMDL0).</td>
</tr>
<tr>
<td>HSSP</td>
<td>High Speed Sequential Processing.</td>
</tr>
<tr>
<td>none</td>
<td>No Fast Path online utility active on the area.</td>
</tr>
<tr>
<td>REORG</td>
<td>Fast Path online reorganization (DBFUMDR0).</td>
</tr>
<tr>
<td>SCAN</td>
<td>Fast Path online sequential dependent scan (DBFUMSC0).</td>
</tr>
</tbody>
</table>

The possible values are as follows:
The XCRB control block represents the active requests. If the major command shows more than one area, XCRB only shows the first area you list. The XCRB fields are:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Shows the name of the DEDB of which this is an area.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of data entry database majors</td>
</tr>
<tr>
<td>XCRB</td>
<td>Shows the active DEDB resource requests in this area.</td>
</tr>
<tr>
<td>Type:</td>
<td>Minor of data entry database majors</td>
</tr>
</tbody>
</table>

- **Jobname**: Name of region holding or waiting for resource. If name is OTHR, an output thread is holding the resource (control interval).
- **PSBName**: Name of the PSB.
- **Trxname**: Name of the transaction.
- **DBname**: Name of the Fast Path DEDB.
- **AREAname**: Name of this partition of the DEDB.
- **R.B.A.**: Relative byte address—the address of the resource (control interval).
- **Status**: Status of the resource request. Possible values are:
  - **EX/OWNER**: Exclusive/owner.
  - **EX/WAITING**: Exclusive/waiting-highlighted.
  - **NE/OWNER**: Non-exclusive/owner.
  - **NE/WAITING**: Non-exclusive/waiting-highlighted.
  - **P.I.**: YES/NO depending on whether the resource control request is known to IMS Program Isolation.
Fast Path output thread command

<table>
<thead>
<tr>
<th>OTHR</th>
<th>Displays information about Fast Path output threads.</th>
</tr>
</thead>
</table>

**Type:** Immediate

OTHR is an immediate command which displays the following information:

- number of defined OTHRs
- number of OTHRs active
- number of OTHRs idle
- count of buffers waiting for an output thread
- total number of buffers queued on output threads

```
+       Defined =  3   Active =    0   Idle =    3
+       Buffers waiting for an OTHR =    0
+       Total buffers queued on OTHRs =    0
```
Extended Recovery Facility (XRF) Support

Introduction

The following commands provide information about the XRF environment.

**XRFS**

| Displays summary information about the XRF system. |

**Type:** Immediate

Figure 27 on page 266 and Figure 28 on page 266 show examples of the XRFS command executed on the active and standby IMS system:

**FIGURE 27. XRFS command executed on the active IMS system**

```
XRFS Active system
+     Class 1 Terminals                  = 350
+     Class 2 Terminals                  =  24
+     RSE name = IMSPROD          AVM is active
```

**FIGURE 28. XRFS command executed on the standby IMS system**

```
XRFS Backup system
+     Terminals in backup mode           = 374
+     Class 1 Terminals                  = 350
+     Class 2 Terminals                  =  24
+     RSE name = IMSPROD          AVM is active
+     Active IMSID = IMSA
+     Active CPUID = 0208570103084
```

<table>
<thead>
<tr>
<th>System status</th>
<th>Indicates whether this IMS system is currently an active or standby IMS system.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eligible terminals</td>
<td>The number of terminals defined in the IMS generation as Class 1 or Class 2 terminals.</td>
</tr>
<tr>
<td>Backup terminals</td>
<td>The number of terminals logged on to the active IMS system with backup sessions on the standby IMS system. This is the number of terminals which must be switched from the active system to the standby system in the event of a takeover.</td>
</tr>
<tr>
<td>RSE name</td>
<td>The name of the recoverable service element (RSE). The RSE is the basic unit of control for XRF. The two IMS systems (active and standby) comprise the RSE.</td>
</tr>
<tr>
<td>AVM status</td>
<td>Indicates whether the availability manager (AVM) is currently active or inactive. To insure integrity of datasets shared between the active and standby systems, XRF must stop all I/O activity when a takeover begins. This is a function of the AVM. The AVM is a component of the MVS system.</td>
</tr>
</tbody>
</table>
Extended Recovery Facility (XRF) Support

**Active IMSID**  The IMSID of the active IMS system. The XRFS command displays this field only when it is executed from the OMEGAMON monitoring the standby IMS system.

**Active CPUID**  The CPUID of the active IMS system. The XRFS command displays this field only when it is executed from the OMEGAMON monitoring the standby IMS system.

XRFS only applies to IMS systems which were installed with XRF.

<table>
<thead>
<tr>
<th>XRFT</th>
<th>Displays general information related to a pending XRF takeover.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Immediate</td>
<td></td>
</tr>
</tbody>
</table>

**XRFT** displays the following information:

**Takeover status**  Indicates whether the IMS standby system is currently taking over the active IMS system.

**Held PSTs**  The number of dependent regions held on the standby system due to backout.

**Sessions not switched**  The number of terminal sessions which still must be switched from the active to the standby system.

OMEGAMON can only execute this command when it is monitoring the standby IMS system. XRFT only applies to IMS systems which were installed with XRF.

**XRSV**  Displays the XRF surveillance status.

<table>
<thead>
<tr>
<th>Type: Immediate</th>
<th></th>
</tr>
</thead>
</table>

The following figures show two examples of XRSV command output:

**FIGURE 29. XRSV command output (RDS mode inactive)**

<table>
<thead>
<tr>
<th>XRSV</th>
<th>Type</th>
<th>Status</th>
<th>Interval</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td></td>
<td>----</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>+</td>
<td>LINK</td>
<td>Active</td>
<td>2 sec</td>
<td>8 sec</td>
</tr>
<tr>
<td>+</td>
<td>LOG</td>
<td>Active</td>
<td>3 sec</td>
<td>9 sec</td>
</tr>
<tr>
<td>+</td>
<td>RDS</td>
<td>Inact</td>
<td>--na--</td>
<td>--na--</td>
</tr>
</tbody>
</table>
FIGURE 30. XRSV command output (LINK surveillance mode not used)

<table>
<thead>
<tr>
<th>XRSV</th>
<th>Type</th>
<th>Status</th>
<th>Interval</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>LINK</td>
<td>Not sel</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>LOG</td>
<td>Active</td>
<td>3 sec</td>
<td>9 sec</td>
</tr>
<tr>
<td>+</td>
<td>RDS</td>
<td>Active</td>
<td>4 sec</td>
<td>7 sec</td>
</tr>
</tbody>
</table>

The headings in the display indicate:

**Type**
- **Link**: Surveillance is done using an ISC link.
- **Log**: Surveillance is done using the IMS system log.
- **RDS**: Surveillance is done using the RDS dataset.

**Status**
- **Active**: Surveillance is in progress.
- **Inact**: Surveillance is not in progress.
- **No sel**: Surveillance is not selected.

**Interval**
The monitoring interval (in seconds).

**Timeout**
The timeout value (in seconds).

XRSV only applies to IMS systems that were installed with XRF.
Chapter overview

This chapter describes the commands available to monitor external subsystems (ESS) defined to BMP regions.

Chapter contents

Major Commands ............................................. 270
Minor Commands ............................................. 271
Introduction

To monitor external subsystems, use the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUBA</td>
<td>Selects all external subsystems.</td>
<td>Major</td>
</tr>
<tr>
<td>SUBL</td>
<td>Selects all external subsystems listed.</td>
<td>Major</td>
</tr>
<tr>
<td>SUBP/n</td>
<td>Selects all external subsystems matching a pattern.</td>
<td>Major</td>
</tr>
</tbody>
</table>

Use the .SPT command to set the pattern.
Minor Commands

Introduction

The following minor commands are available:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECRC</td>
<td>Displays the command recognition character used to pass commands to the external subsystem.</td>
</tr>
</tbody>
</table>

**Type:** Minor of external subsystem majors

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARM</td>
<td>Displays all dependent regions which have the first displayed DB2 subsystem defined and their associated parameters.</td>
</tr>
</tbody>
</table>

**Type:** Minor of external subsystem majors

The PARM command displays the following information for the first subsystem the major command lists:

- IMS region name
- current interface status
- language interface token
- resource translation table name
- interface control module name
- error option specification

The current interface status is one of the following:

- **Cre-thrd** Creating a thread.
- **SQL-call** SQL call in progress.
- **ph1-sync** Phase 1 commit in progress.
- **ph2-sync** Phase 2 commit in progress.
- **trm-thrd** Terminating a thread.
- **conn** Subsystem connected to control region.
- **not-con** Subsystem not connected to control region.
- **not-def** Subsystem not defined to control region.
- **unable** Dependent region has no connection to DB2 subsystem due to an error detected in subsystem, unavailable resources, or the connection was never established.
Minor Commands

<table>
<thead>
<tr>
<th>STAT</th>
<th>Displays the status of the DB2 subsystem.</th>
</tr>
</thead>
</table>

**Type:** Minor of external subsystem majors

The status is one of the following:

- **Active** The DB2 subsystem is doing work for an BMP dependent region.
- **Conn** The DB2 subsystem is not doing work for any BMP dependent region.
- **Not-Con** The DB2 subsystem is not connected to any BMP dependent region.
- **Not-Def** The DB2 subsystem is not defined to any BMP dependent region.

Use the PARM minor command to determine which BMP dependent regions are using the DB2 subsystem.
Chapter overview

This section gives you basic information about using the authorized command facility, such as system access considerations, supplying the action character, accessing other address spaces, and specifying addresses.

In this manual, the term *authorized commands* refers to certain sensitive or powerful commands that Candle Corporation ships with a security level of 3. These commands can be accessed only by entering a password. We ship all other commands with a security level of 0.

Chapter contents

- Using Authorized Commands ........................................... 274
- Displaying and Modifying Storage .................................. 281
- Console-related Commands ............................................ 300
- Online Facility for Logging IMS Messages ....................... 320
- Dynamically Allocating and Unallocating Datasets .............. 323
Using Authorized Commands

Introduction

Authorized commands can alter data in storage and display data from restricted storage areas. Misuse of authorized commands can jeopardize your system and the integrity of your data. Use them with caution.

OMEGAMON’s internal security facility provides security for all OMEGAMON commands. Each OMEGAMON command can have a security level of 0, 1, 2, or 3. Security level 3 provides the highest degree of protection; levels 2 and 1 provide successively lower degrees of protection. A setting of 0 means that any user can access the command.

OMEGAMON allows you to secure OMEGAMON commands by setting up an interface with your installation’s external security system (for example, RACF™ or ACF2®). The security facility is described in the Configuration and Customization Guide.

Supplying the password

To access authorized commands, you need to enter your password via the /PWD INFO-line command. For information on how to use /PWD, see “Operational Commands” on page 37.

Considerations for access

The execution of authorized commands may fail because the Resource Measurement Facility (RMF) Monitor I is not active. To see if the RMF Monitor I is active, enter the .RMF immediate command.

<table>
<thead>
<tr>
<th>.RMF</th>
<th>Displays Resource Measurement Facility information.</th>
</tr>
</thead>
</table>

Type: Immediate

The .RMF immediate command displays whether the Resource Measurement Facility (RMF) Monitor I is active and, if it is, the version number and release level.

**Note:** The amount of information displayed by .RMF varies according to the level of RMF that you are running.
How to use the action character

You must enter most OMEGAMON commands with an action character in column 1 before OMEGAMON can execute the command. The action character is a hyphen (-). Some commands require this character whenever you invoke them; others require the action character only on certain occasions.

If you omit the action character from a command that requires it, OMEGAMON displays a No Act message at the far right of the line. To remove the No Act message, type a hyphen in column 1 and press Enter. OMEGAMON then executes the command.

Table 7: Commands that require the action character on page 275 shows the authorized commands that require the action character and when they require it.

Table 7. Commands that require the action character

<table>
<thead>
<tr>
<th>Command</th>
<th>Action Character Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONS</td>
<td>Never required.</td>
</tr>
<tr>
<td>CONU</td>
<td>Never required.</td>
</tr>
<tr>
<td>ACTN</td>
<td>Never required.</td>
</tr>
<tr>
<td>LINE</td>
<td>Never required.</td>
</tr>
<tr>
<td>MNT</td>
<td>Never required.</td>
</tr>
<tr>
<td>DYNA</td>
<td>Required to allocate dataset.</td>
</tr>
<tr>
<td>DYNU</td>
<td>Required to allocate dataset.</td>
</tr>
<tr>
<td>ICMD</td>
<td>Always required.</td>
</tr>
<tr>
<td>ICNS</td>
<td>Always required.</td>
</tr>
<tr>
<td>ILST</td>
<td>Required if storage is not in your key.</td>
</tr>
<tr>
<td>DYNU</td>
<td>Required if storage is not in your key.</td>
</tr>
<tr>
<td>ISCN</td>
<td>Required if storage is not in your key.</td>
</tr>
<tr>
<td>IZAP</td>
<td>Required if storage is not in your key.</td>
</tr>
<tr>
<td>MCHN</td>
<td>Never required.</td>
</tr>
<tr>
<td>MDEF</td>
<td>Never required.</td>
</tr>
<tr>
<td>MLOG</td>
<td>Always required.</td>
</tr>
<tr>
<td>MLST</td>
<td>Never required.</td>
</tr>
<tr>
<td>MSCN</td>
<td>Never required.</td>
</tr>
<tr>
<td>MZAP</td>
<td>Required if storage is not in your key.</td>
</tr>
<tr>
<td>OCMD</td>
<td>Always required.</td>
</tr>
<tr>
<td>OSPC</td>
<td>Never required.</td>
</tr>
</tbody>
</table>
OMEGAMON SRB/XMS facility

OMEGAMON has several authorized commands that must schedule an SRB in an IMS region. (for example, ICMID, PEEK, XMLS, XMSC, and XMZP).

**Note:** OMEGAMON completely recovers from all abnormal conditions; it should never cause a system dump or affect the IMS region in any way (other than to swap it into memory).

**SRB time-outs**

OMEGAMON only waits 10 seconds for an SRB to be dispatched. If the SRB does not dispatch in this period, OMEGAMON purges it automatically and continues. If this occurs, OMEGAMON displays the following message.

**WARNING**  **SRB TIMED OUT AFTER 10.00 SECONDS (RC=8)**

This message indicates that the IMS region could not service OMEGAMON’s SRB within the 10-second interval, so the SRB was purged. The problem might be only temporary, so you should retry whatever command causes the condition.

### Table 7. Commands that require the action character

<table>
<thead>
<tr>
<th>Command</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>PEEK</td>
<td>Required the first time you look at a particular job or need to collect new information. See “Collecting data about address spaces” on page 307 for complete information.</td>
</tr>
<tr>
<td>AMAP</td>
<td>Never required.</td>
</tr>
<tr>
<td>DATA</td>
<td>Never required.</td>
</tr>
<tr>
<td>DDNS</td>
<td>Never required.</td>
</tr>
<tr>
<td>JOBS</td>
<td>Never required.</td>
</tr>
<tr>
<td>MODS</td>
<td>Never required.</td>
</tr>
<tr>
<td>STEP</td>
<td>Never required.</td>
</tr>
<tr>
<td>SUBP</td>
<td>Never required.</td>
</tr>
<tr>
<td>TCBS</td>
<td>Never required.</td>
</tr>
<tr>
<td>SLST</td>
<td>Always required.</td>
</tr>
<tr>
<td>SSCN</td>
<td>Always required.</td>
</tr>
<tr>
<td>SCHN</td>
<td>Always required.</td>
</tr>
<tr>
<td>SZAP</td>
<td>Always required.</td>
</tr>
<tr>
<td>XMCH</td>
<td>Always required.</td>
</tr>
<tr>
<td>XMLS</td>
<td>Always required.</td>
</tr>
<tr>
<td>XMSC</td>
<td>Always required.</td>
</tr>
<tr>
<td>XMZP</td>
<td>Always required.</td>
</tr>
</tbody>
</table>

---

**Using Authorized Commands**

---
SRB/XMS routine program checks

If the IMS region is damaged, some of its control blocks may be overlaid or contain invalid pointers or addresses. OMEGAMON monitors this condition by checking the ID field of each control block before use. If it finds a suspect control block, it discontinues the analysis, and displays an error message.

In some cases, this situation can cause the SRB/XMS routine to program check when it runs in the IMS region. OMEGAMON’s SRB/XMS routines recover via a functional recovery routine (FRR) when a program check occurs, and the issuing command displays a warning message like the following:

```
WARNING      SRB ROUTINE TERMINATED ABNORMALLY - RC=S0C4
PSW AT TIME OF ERROR = xxxxxxxxxx xxxxxxxxxx
+ R0-8 xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx
+ R9-F xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx
+ A0-8 xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx
+ A9-F xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx xxxxxxxxxx
```

This hexadecimal display shows the PSW and general registers at the time of the program check.
Address Specification for Storage Commands

Introduction

You can specify, modify, or pre-define an address (addr) for commands that display or modify storage or data-only spaces (ESA). An address consists of an anchor, optional modifiers, and an optional pre-defined name.

Anchors

An anchor is the base address of an address specification. It can be:

- **Absolute** The hexadecimal address.
- **Symbolic** Up to eight alphanumeric characters, including @, #, and $. See “Pre-definitions” on page 279.

Modifiers

You can supply one or more modifiers to change the location that the anchor points to. A modifier can be:

- **Offset** A plus sign (+) or minus sign (-), followed by a hexadecimal number.
  This modifier specifies a location at a known offset (positive or negative) from the anchor address.
- **Indirect** Use a question mark (?) as the symbol for 31-bit (XA or ESA) addressing.
  This modifier indicates that the location pointed to is itself an address.

You can use these modifiers to create very powerful and versatile address expressions. For example, the following address expression locates the TIOT of the currently executing MVS task:

\[ 10\% +4\% +C\% \]

This expression has the following components:

- **10\%** Treats the data at location X’10’ as a 24-bit address. This is the address of the MVS CVT.
- **%** The second % treats the data at the start of the CVT as a 24-bit address. The first word of the CVT contains the address of a double word (8 bytes). The double word contains:
  - the address of the next TCB to dispatch (bytes 0 to 3)
  - the address of the currently dispatched TCB (bytes 4 to 7)
Address Specification for Storage Commands

You can specify or reference an address by a name consisting of up to eight alphanumeric characters, including @, #, and $. The following command specifies address names.

| MDEF    | Defines names for addresses. |

The MDEF command labels an address in storage with a name so that you do not have to repeatedly enter complex or frequently used storage addresses.

**Pre-definitions**

Consider the following points:

- OMEGAMON saves the names you define in a table, but does not save the table from session to session. Save address names that you want to use from session to session in a screen space.
- OMEGAMON provides a table of predefined names, but if you define an address with one of the predefined names, OMEGAMON uses the address you define during the session.
- OMEGAMON places a comment character (>) in front of MDEF after it executes.

**Type:** Immediate

**Format:** `MDEFcc addr.name`

- `cc` The type of operation.
  - CL Clears the address name table, and resets it to initial status.
  - LP Lists pre-defined address names.
  - LS Lists the address names in the table.
  - DE Deletes an address name.
  - bb Assigns an address name.

- `addr` The address you want to name. See “Address Specification for Storage Commands” on page 278 for more information.

- `name` The address name. Up to eight alphanumeric characters, including @, #, and $.
In the next example, MDEF gives the name XYZ to the address at offset X'4A' in the CSA.

\textbf{MDEF CSA+4A,XYZ}

You can also use a pre-defined control block identifier to specify an anchor. OMEGAMON comes with pre-defined names for control blocks in a table. Use the MLST command to display the table.

\begin{table}[h]
\centering
\begin{tabular}{|l|l|}
\hline
Identifier & Description \\
\hline
CVT & MVS communications vector table. \\
\hline
ESCD & IMS extended system contents directory. \\
\hline
IMSTCB & Physical logger TCB. \\
\hline
SCD & IMS system contents directory. \\
\hline
\end{tabular}
\end{table}
Displaying and Modifying Storage

Introduction

This section describes the authorized commands that display, scan, or modify address names for storage locations.

Storage display commands

The following commands display storage locations. You can use the ILST and MLST commands to look at IMS control blocks and memory locations. An important use of the ILST and MLST commands, especially when you use them with the DUMP command from the regular OMEGAMON command package, is for IMS education. Certain IMS education courses ask students to look through IMS control blocks in sample memory dumps. Students find that they can use OMEGAMON commands to look at those control blocks in their own system while it is running and increase the usefulness of such courses.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILST</td>
<td>Lists memory from the IMS address space.</td>
</tr>
</tbody>
</table>

ILST displays the contents of the private area of:

- the IMS control region
- control region
- the DLS region
- the DBRC region
- the IRLM address space

Note: Some IMS private storage requires no authorization or special key to display; however, some areas are store-protected. To display these areas you must supply the action character in the label field of the ILST command.
Type: Immediate

Format: **aILStcc addr,len**

- **a** If required, an action character (\(-\)) in column 1.
- **cc** The format and source address space of the display.

The first character specifies the display format:

- **b, B, or .** Hex and character (default).
- **C** Character only.
- **X** Hex only.

The second character specifies the address space OMEGAMON displays:

- **b or I** The IMS control region (default).
- **D** The IMS DLS region.
- **R** The IMS DBRC region.
- **L** The IMS IRLM region.

**addr** The first address of storage that OMEGAMON displays. See “Address Specification for Storage Commands” on page 278 for more information.

**len** The number (up to eight hex digits) of bytes that OMEGAMON displays. Default is 16 (X'10') bytes.

In the following example, **ILST** displays 80 (X'50') bytes starting at hex address 15547C:

**ILST 15547C,50**

This command defaults to displaying the IMS control region in both hex and character formats as shown in Figure 31 on page 283.
Displaying and Modifying Storage

FIGURE 31. Typical output for ILST

<table>
<thead>
<tr>
<th>MLST</th>
<th>Displays bytes of memory from either the common area or the OMEGAMON private storage area.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `aMLSTc addr,dlen`

- `a`: A K in the label position requests a display showing the fetch protection key for each virtual block in the range specified, and whether fetch protection is ON or OFF.
- `c`: Specifies the format of the output. Possible values are:
  - `B` or `b`: Hex and character (default).
  - `C`: Character only.
  - `X`: Hex only.

- `addr`: The first address of storage that OMEGAMON displays. See “Address Specification for Storage Commands” on page 278 for more information.

- `dlen`: The number (up to eight hex digits) of bytes that OMEGAMON displays. The default is 16 (X’10’) bytes.

In the following display, MLST lists 32 (X’20’) bytes, starting at address 1EB0, in character format.

In the next display, MLST lists 16 (X’10’) bytes, starting at address FF32D6, in both hex and character formats.
The next example shows MLST with a K in the label field, which displays fetch protection information.

```
KMLST 7EF000,4000                                Addr= 007EF000
+      Virtual Block Number: 07EF000   KEY: 5  Fetch Protection: OFF
+      Virtual Block Number: 07F0000   KEY: 1  Fetch Protection: ON
+      Virtual Block Number: 07F1000   KEY: 1  Fetch Protection: OFF
+      Virtual Block Number: 07F2000   KEY: 0  Fetch Protection: OFF
```

The next three figures show uses of the XMLS command.

<table>
<thead>
<tr>
<th>XMLS</th>
<th>Displays storage from IMS address spaces.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Immediate</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>aXMLSc targ,addr,dlen</td>
</tr>
<tr>
<td>a</td>
<td>An action character in column 1.:</td>
</tr>
<tr>
<td>-</td>
<td>Changes to a comment character (&gt;) after command executes.</td>
</tr>
<tr>
<td>&lt;</td>
<td>Does not change to a comment character after command executes. Use this action character to repeat the command.</td>
</tr>
<tr>
<td>c</td>
<td>The format of the output. Possible values are:</td>
</tr>
<tr>
<td>B or b</td>
<td>Hex and character (default).</td>
</tr>
<tr>
<td>C</td>
<td>Character only.</td>
</tr>
<tr>
<td>X</td>
<td>Hex only.</td>
</tr>
<tr>
<td><strong>targ</strong></td>
<td>The IMS region. It can be:</td>
</tr>
<tr>
<td>nnnn</td>
<td>Decimal ASID number.</td>
</tr>
<tr>
<td>cccccccc</td>
<td>Jobname.</td>
</tr>
<tr>
<td><strong>addr</strong></td>
<td>The first address of storage that OMEGAMON displays. See “Address Specification for Storage Commands” on page 278 for more information.</td>
</tr>
<tr>
<td><strong>dlen</strong></td>
<td>The number (up to eight hex digits) of bytes that OMEGAMON displays. The default is 16 (X'10') bytes. The maximum is 4096 (X'1000') bytes.</td>
</tr>
</tbody>
</table>

The next three figures show uses of the XMLS command.
In the following figure, XMLS displays 32 (X'20') bytes from address space 21 (starting at address 1EB0) in character format.

**FIGURE 32. XLMS output in character format**

```xml
>XMLSC 21,1EB0,20
>storage at 00001EB0 in IMS210FP ASID=21:
> 0000 *ABCDEFGH IJKLMNOP QRSTUVWX Z0123456
```

In the following figure, XMLS displays 16 (X'10') bytes from the IMS region specified by the jobname USER14, starting at 1EB0, in both hex and character formats.

**FIGURE 33. XLMS output in hex and character formats**

```xml
>XMLS USER14,1EB0,10
>storage at 00001EB0 in USER14 ASID=21:
> 0000 C1C2C3C4 C5C6C7C8 C9D1D2D3 D4D5D6D7 *ABCDEFHIJKLMNOP*
```

In the following figure, XMLS displays 16 (X'10') bytes from address space 21, starting at FF32D6, in hex and character format. The less-than symbol (<) prevents OMEGAMON from commenting out the command.

**FIGURE 34. Output from XMLS command with an action character**

```xml
<XMLSB 21,FF32C1+15,10
+storage at 00FF32D6 in TSOA ASID=21:
+ 0000 4AA800F7 D3700000 00000000 000000F8 *y..7L..........8*
```

**Storage scan commands**

The following commands scan storage locations for a specified string of values and display the string if it is found.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ISCN</strong></td>
<td>Scans memory in the IMS address space.</td>
</tr>
</tbody>
</table>

**Type:** Immediate

ISCN scans storage for a string of hexadecimal or character values. If the scan is successful, OMEGAMON displays the string.
ISCN scans the private areas of:

- the IMS control region
- the DLS region
- the DBRC region
- the IRLM address space

**Note:** Some IMS private storage requires no authorization or special key to scan; however, some areas are store-protected. To scan these areas, you must supply the action character in the label field of the ISCN command.
Displaying and Modifying Storage

Format:  

aISCNcc addr,string,slen,dlen

a  
If required, an action character (- or <) in column 1.

c  
The format and source address space of the display. Possible values are:

b or .l  
The IMS control region in hex and character format (default).

C  
The IMS control region in character format only.

X  
The IMS control region in hex format only.

.D  
The IMS DLS region in hex and character format.

.R  
The IMS DBRC region in hex and character format.

.L  
The IMS IRLM region in hex and character format.

addr  
The first address of storage that OMEGAMON scans. See “Address Specification for Storage Commands” on page 278 for more information.

string  
The hex string OMEGAMON uses for the scan. If you enclose it in single quotes, OMEGAMON assumes it is a character string.

Note: OMEGAMON interprets two single quotes ("') within a character string as a single quote (').

slen  
The number (up to eight hex digits) of bytes that OMEGAMON scans. Default is 256 (X’100’) bytes.

dlen  
The number (up to eight hex digits) of bytes that OMEGAMON displays if the scan is successful. The display starts at the beginning of string. Default is 16 (X’10’) bytes.

In the following example, ISCN scans 1280 (X’500’) bytes of storage, starting at hex address 155000, for the hex string 00000020 and displays 80 (X’50’) bytes starting at that point. This command defaults to displaying the IMS control region in both hex and character formats.

ISCN  155000,00000020,500,50
Typical output (if the string is found) is:

<table>
<thead>
<tr>
<th>Type: Immediate</th>
</tr>
</thead>
</table>

MCHN scans the elements of a table for a string of hex or character values. If the scan is successful, OMEGAMON displays the table element that contains the string.

Use MCHN to examine the following:

- Common Storage Area (CSA)
- System Queue Area (SQA)
- nucleus

If you want to search private storage areas other than OMEGAMON’s, use the XMCH command.
You must make sure that \texttt{addr} is the starting point of a table element. The address at \texttt{addr + chain} points to the next table element. The scan ends when the value at \texttt{addr + chain} is one of the following: 0, -1, or \texttt{addr} (the table is a ring).

The following example shows a typical MCHN command.

\textbf{MCHN AAB6C8,D6C30199,8,4}

In this example, MCHN scans a table that starts at location AAB6C8 and looks for the string D6C30199 that begins at the eighth byte of the table element; the address of the next table element is at offset 4. By default, this command displays 16 bytes of the table element in hex and character notation.
The following output appears if the scan is successful.

```
Addr=007DA000
MCHN AAB6C8,D6C30199,8,4
+ 0000 22E2C3E3 00000000 D6C30199 00000000 *SSCT OC r *
```

| MSCN   | Scans storage for a string of data and displays the location. |

MSCN scans the address space in which OMEGAMON resides for a string of hex or character values. If the scan is successful, OMEGAMON displays the string.

**Type:** Immediate

**Format:** `MSCN addr,string,slen,dlen`

- `c` The format of the output. Possible values are:
  - `B` or `b` Hex and character (default).
  - `C` Character only.
  - `X` Hex only.

- `addr` The first address of storage that OMEGAMON scans. See “Address Specification for Storage Commands” on page 278 for more information.

- `string` The hex string OMEGAMON uses for the scan. If you enclose it in single quotes, OMEGAMON assumes it is a character string.

  **Note:** OMEGAMON interprets two single quotes (""") within a character string as a single quote (‘’).

- `slen` The number (up to eight hex digits) of bytes that OMEGAMON scans. The default is 256 (X'100') bytes.

- `dlen` The number (up to eight hex digits) of bytes that OMEGAMON displays if the scan is successful. The display starts at the beginning of the string. The default is 16 (X'10') bytes.

In the next example, MSCN scans the first 1000 bytes of the TIOT entry for the character string OIHELP (see “Address Specification for Storage Commands” on page 278 for an explanation of the addressing), and displays 14 hex bytes starting at that point. The display is in both hex and character formats.

```
MSCN 10%%+4%+C%,’OIHELP’,1000,14
```
Typical output of the MSCN command is shown here.

XMCH scans the elements of a table for a string of hex or character values. If the scan is successful, OMEGAMON displays the table element that contains the string.

Use XMCH to search IMS address spaces. Use MCHN to search the address space in which OMEGAMON resides.

XMCH scans the elements of a table in an IMS region for a string of hex or character values. If the scan is successful, OMEGAMON displays the table element that contains the string.

Use XMCH to search IMS regions. Use MCHN to search the address space in which OMEGAMON resides.

**XMCH**

Scans tables in the IMS address space.

XMCH scans the elements of a table for a string of hex or character values. If the scan is successful, OMEGAMON displays the table element that contains the string.

Use XMCH to search IMS address spaces. Use MCHN to search the address space in which OMEGAMON resides.

XMCH scans the elements of a table in an IMS region for a string of hex or character values. If the scan is successful, OMEGAMON displays the table element that contains the string.

Use XMCH to search IMS regions. Use MCHN to search the address space in which OMEGAMON resides.

**Type:** Immediate

**Format:**

\[aXMCHc\text{ targ,addr,string,olen,chain,dlen}\]

- \(a\): An action character in column 1:
  - \(\_\): Changes to a comment character (>) after command executes.
  - \(<\): Does not change to a comment character after command executes. Use this action character to repeat the command.

- \(c\): The format of the output. Possible values are:
  - \(B\) or \(b\): Hex and character (default).
  - \(C\): Character only.
  - \(X\): Hex only.

**targ**: The IMS region. It can be:

- \(nnnn\): Decimal ASID number.
- \(cccccccc\): Jobname.
You must make sure that `addr` is the starting point of a table element. The address at `addr + chain` points to the next table element. The scan ends when the value at `addr + chain` is one of the following: 0, -1, `addr` (the table is a ring).

The following example shows a typical XMCH command.

```
-XMCH USER14,AAB6C8,D6C30199,8,4
```

In this example, XMCH scans a table in the IMS region that starts at location AAB6C8 and looks for the string D6C30199 that begins at the eighth byte of the table element. The address of the next table element is at offset 4. By default, this command displays 16 bytes of the table element in hex and character format.

The following output appears if the scan is successful.

```
>XMCH USER14,AAB6C8,D6C30199,8,4
>Storage at 007DA000 in USER14 ASID=21:  
> 0000  E2E2C3E3 00000000 D6C30199 00000000  *SSCT OC r *
```
XMSC scans an IMS address space for a string of hex or character values. If the scan is successful, OMEGAMON displays the string.

XMSC scans an IMS region for a string of hex or character values. If the scan is successful, OMEGAMON displays the string.

<table>
<thead>
<tr>
<th><strong>Type:</strong></th>
<th>Immediate</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Format:</strong></td>
<td>aXMSCc targ,addr,string,slen,dlen</td>
</tr>
</tbody>
</table>

- **a** | An action character in column 1.: |
  - | Changes to a comment character (>) after command executes. |
  - | Does not change to a comment character after command executes. Use this action character to repeat the command. |
- **c** | The format of the output. Possible values are: |
  - **B or b** | Hex and character (default). |
  - **C** | Character only. |
  - **X** | Hex only. |
- **targ** | The IMS region. It can be: |
  - **nnnn** | Decimal ASID number. |
  - **cccccccc** | Jobname. |
In the next example, XMSC scans 1000 bytes in the IMS region specified by the jobname USER14, starting at location 515988 for the character string WORKAREA. If the scan is successful, OMEGAMON displays 14 hex bytes in hex and character format, starting at WORKAREA.

-XMSC USER14,515988,'WORKAREA',1000,14

The following output appears if the scan is successful.

Storage modification commands

The following commands modify storage locations. Use these commands with caution.

Caution

IZAP is very powerful. Use it with extreme care.

IZAP | Zaps memory in the IMS address space.

Type: Immediate

IZAP modifies the contents of the private areas of:

- the IMS control region
- the DLS region
- the DBRC region
- the IRLM address space

Consider the following points:

- Some IMS private storage requires no authorization or special key to modify; however, some areas are store-protected. To modify these areas, you must supply the action character in the label field of the IZAP command.

- If you use IZAP to modify storage in the pageable link pack area (PLPA), IZAP automatically does a long-term page-fix to ensure that the storage remains modified. If this is necessary, OMEGAMON displays this message:

  PAGE(S) FIXED
Displaying and Modifying Storage

IZAP changes a fullword at location 6764 from '0000000A' to '00000064'.

IZAP 6744+20,0000000A,00000064

In the following example, IZAP changes a byte at location B46E08 from '80' to '40'.

IZAP B46DE8+20,80,40

The next figure is a typical output from a successful IZAP.

>IZAP B46DE8+20,80,40
> >> MEMORY ZAP SUCCESSFUL. <<
Displaying and Modifying Storage

<table>
<thead>
<tr>
<th>MZAP</th>
<th>Modifies the contents of the common area or the OMEGAMON private storage area.</th>
</tr>
</thead>
</table>

**Type:** Immediate

MZAP modifies the contents of the common area:

- Common Storage Area (CSA)
- System Queue Area (SQA)
- nucleus

Consider the following points:

- Some commonly-addressable storage requires no authorization or special key to modify; however, some areas are store-protected. To modify these areas, you must supply the action character in the label field of the MZAP command. You can also use MZAP to zap storage in the OMEGAMON address space for debugging purposes.
- If you use MZAP to modify storage in the pageable link pack area (PLPA), MZAP automatically does a long-term page-fix to ensure that the storage remains modified. If this is necessary, OMEGAMON displays the following message:

```
PAGE(S) FIXED
```

**Format:** `-MZAP addr,ver,rep`

- A required action character in column 1. The hyphen changes to a comment character (>) after the command executes.

- `addr` The address of the string that OMEGAMON may modify. See “Address Specification for Storage Commands” on page 278 for more information.

- `ver` The verify string. OMEGAMON modifies storage only if OMEGAMON finds this string at `addr`. If OMEGAMON does not find the string, it displays what is actually at `addr`.

- `rep` The replacement string. If OMEGAMON finds `ver` at `addr`, `rep` replaces `ver`.

**Note:** The verify and replacement strings must be the same length.

In the next example, MZAP changes a fullword at location 6764 from X’A’ to X’64’

```
MZAP 6744+20,0000000A,00000064
```
The next example shows how MZAP changes X’FF’ to X’00’ at location EA65C0.

MZAP EA65C0,FF,00

<table>
<thead>
<tr>
<th>XMZP</th>
<th>Modifies another user’s private storage area.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `-XMZP  targ,addr,ver,rep`

- A required action character in column 1. The hyphen changes to a comment character (>) after the command executes.
- **targ** The IMS region. It can be

  - **nnnn** Decimal ASID number.
  - **cccccccc** Jobname.

- **addr** The address of the string that OMEGAMON may modify. See “Address Specification for Storage Commands” on page 278 for more information.

  The verify string. OMEGAMON modifies storage only if OMEGAMON finds this string at `addr`. If OMEGAMON does not find the string, it displays what is actually at `addr`.

- **rep** The replacement string. If OMEGAMON finds `ver` at `addr`, `rep` replaces `ver`.

**Note:** The verify and replacement strings must be the same length.

In the next example, XMZP changes a byte at location A0160 in the master scheduler address space.

-XMZP  *MASTER*,A0160,0A,64

In the next example, XMZP changes a halfword at C4834 in the PRODJOB address space from X’1854’ to X’0700’.

-XMZP  PRODJOB,C4834,1854,0700

In the example below, XMZP changes a halfword at location D7E30 from X’18C0’ to X’18C1’.

-XMZP  BMP01,D7E30,18C0,18C1
Typical output (if the zap is successful) is:

```
>XMZP BMP01,D7E30,18C0,18C1
>    >> MEMORY ZAP SUCCESSFUL <<
```
Introduction

This section discusses the OMEGAMON commands available for use at the IMS MTO console (ICMD and ICNS) and the MVS operator console (OCMD, CONS, ACTN, LINE, and MNT).

IMS MTO console support

OMEGAMON provides commands to issue IMS commands or transactions from the OMEGAMON console and to display IMS messages and IMS-related MVS messages at the OMEGAMON console.

IMS commands and transactions

Use the ICMD command to enter any IMS command or transaction.

<table>
<thead>
<tr>
<th>ICMD</th>
<th>Enters an IMS command or transaction.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `-ICMD cc...cc`

- OMEGAMON requires the action character (\(-\)) in column 1 to execute the ICMD command.
- `cc...cc` Any string that you could enter at an MVS console in response to the IMS READY WTOR. ICMD is much more convenient to use because you do not have to know the WTOR reply ID number.

The next example is a typical ICMD command.

**-ICMD /DIS CCTL ALL**

The ICMD command uses the WTO macro to echo commands you enter; this echo appears on the ICNS display. (See “IMS messages and IMS-related MVS messages (ICNS)” on page 301.) Because OMEGAMON uses the IMS WTOR, you must end the IMS transaction message with a period (.), or IMS thinks you only entered the first segment of a multi-segment input message, and prompts you to enter the next segment. The MVS (WTOR) console, and therefore ICMD, always has the same authorization as the IMS master terminal. See Figure 35 on page 303 for an example. The output also appears on the MVS master console, because ICMD uses IMS’s WTOR console support. OMEGAMON passes the text you enter via ICMD to IMS by REPLYing to the outstanding IMS WTOR (IMS READY message). In response, IMS displays any reply via WTO.

OMEGAMON executes only one ICMD command per cycle and defers any other ICMDs on the screen until later cycles. This is because IMS has only one WTOR available.
If two users at two different OMEGAMON terminals try to issue ICMD at the same instant, it is possible (although unlikely) for OMEGAMON to execute only one of the commands. This is because, for a very short interval, there is no IMS reply ID to respond to. If this happens, ICMD displays the following message:

>> IMS Reply ID not found: RC=4 <<

In this case, issue the ICMD command again. If ICMD fails repeatedly or displays a return code other than 4, contact Candle Customer Support.

ICMD uses an SRB to extract the current IMS reply ID number. The overhead is not significant, but “OMEGAMON SRB/XMS facility” on page 276 describes the operational factors which apply in this case.

**IMS messages and IMS-related MVS messages (ICNS)**

The ICNS command displays the WTO and MTO message traffic that the IMS control region(s) and its dependent regions produce. These messages include any responses to commands and transactions you enter using the ICMD support described in the previous section. OMEGAMON does not include DFS996I IMS/VS READY messages on the display, because these have no value. Similarly, OMEGAMON removes the prefix DFS000I from informational messages that begin with this identifier.

ICNS shows a consolidated display of IMS and MVS messages associated with IMS control and dependent regions. The command has a buffer to retain 99 messages for display in chronological order. OMEGAMON only generates null replies as long as the ICNS command remains on the OMEGAMON screen.

During JES hotstarts, ICNS displays the following message:

>> ICNS unable to receive messages from JES at this time <<

This message appears temporarily until the hotstart is complete, when ICNS resumes proper functioning. Every time an IMS region issues a WTO, ICNS captures and stores the message in a buffer accessible to OMEGAMON sessions. (ICNS does not disturb the messages; they still appear on MVS consoles as usual.) The buffer lets you display as many as the most recent 99 WTO messages. It also primes the buffer with a message giving the date and time the support was activated.

<table>
<thead>
<tr>
<th>ICNS</th>
<th>Displays IMS WTO and MTO message traffic.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** -ICNSnn

- A required action character (-) in column 1.

- nn The number (01-99) of lines OMEGAMON displays. ICNS displays the last nn lines of intercepted messages.
**Note:** Because ICNS is only a display command, it does not automatically convert to a comment after one execution; ICNS can execute as many times as you want it to.

The following ICNS command displays the last 18 message lines:

-ICNS18

You can use the ICNS command with the ICMD authorized command (see Figure 35 on page 303 for an example).

ICNS captures messages only from active IMS regions. For example, when a new message region or BMP starts, ICNS does not see the **JOB STARTED** messages. This is because at the time JES issued those messages, the region had not yet told IMS of its existence. ICNS intercepts messages only if the issuing address space currently has an entry in the IMS SVC directory table; that is, if IMS initialized the address space and assigned it a PST.

Similarly, when a dependent region terminates, ICNS does not see the **JOB ENDED** messages, because JES issued them after the region disassociated itself from IMS. Between startup and shutdown, however, ICNS sees every message that appears on the JES job log of the control region or any of its dependents. However, ICMD suppresses the **DFS996I IMS READY WTOR** messages. Because the ICMD authorized command automatically extracts the IMS reply ID number, the user does not need this message, and ICNS removes it to eliminate clutter.

ICNS observes messages originating from the DLS, DBRC, IRLM, and Virtual Fetch address spaces.

ICNS performs another useful function with ICMD. While the output generated by IMS commands issued at the MVS console appears immediately, output from transactions entered at the MVS console and unsolicited (for example, application program ABEND) messages do not appear on the MVS console until you reply again to the next IMS WTOR message. IBM’s **IMS/VS Operator’s Reference Manual** states that this function prevents important IMS messages from getting lost in the heavy traffic to the MVS operator console. During each OMEGAMON cycle, the ICNS command checks whether there are any queued messages for the WTOR LTERM. If there are, it automatically issues a null reply (R xx, ’‘), which forces the output to appear on the MVS console (and, of course, on the ICNS display during the next cycle). This prevents you from having to issue the dummy reply yourself, or worse, not seeing an important IMS message until it is too late. Remember that this automatic reply feature is only active when ICNS is on the OMEGAMON screen.
The following figure shows a typical ICNS command used with the ICMD command.

**FIGURE 35. ICNS command with the ICMD authorized command**

OCMD

Issues MVS and JES2 operator commands from an OMEGAMON terminal.

<table>
<thead>
<tr>
<th>Type:</th>
<th>Immediate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format:</td>
<td>`-OCMDnn &lt;CONS=&lt;conid</td>
</tr>
</tbody>
</table>

OMEGAMON issues the command (cccc) that you supply via SVC 34.

OCMD requires the action character (-) in column 1. The action character changes to a comment character after execution.

For commands that accept a return destination, the variable `nn` indicates which operator’s console will receive the response. If you omit `nn`, the response goes to the master console.

The variable `cccc` is an MVS or JES2 command.

`conid` specifies which operator’s console issued the command. This operand overrides the `nn` operand.

`conname` specifies the console name from which the command originated.

The following screen display shows an example of the OCMD command.
OMEGAMON has the same MVS console authority as the console you indicate. Therefore, if you want to issue a command that requires master console authority (such as VARY CHANNEL) you must specify the MVS console ID of your current master console. If you do not specify a console ID, the master console ID is used.

<table>
<thead>
<tr>
<th>RCMD</th>
<th>Routes MVS and JES2 commands to a desired system in a sysplex.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>-RCMDnn sysname &lt;CONS=&lt;conid</td>
</tr>
</tbody>
</table>

OMEGAMON issues the command (cccc) that you supply via SVC 34.

RCMD requires the action character (-) in column 1. The action character changes to a comment character after execution.

The variable *nn* indicates which operator’s console will receive the response. If you omit *nn*, the response goes to the master console.

*sysname* specifies the system name where the command executes.

The variable *cccc* is an MVS or JES2 command.

*conid* specifies which operator’s console issued the command. This operand overrides the *nn* operand.

*conname* specifies the console name from which the command originated.

<table>
<thead>
<tr>
<th>CONS</th>
<th>Displays the console image for the specified console.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>Major</td>
</tr>
<tr>
<td><strong>Format:</strong></td>
<td>CONSnn &lt;CONS=&lt;conid</td>
</tr>
</tbody>
</table>

The variable *nn* is an MVS console number. This number ranges from 01 to the maximum number of consoles generated for your installation. If *nn* is omitted, OMEGAMON selects the master console.

*conid* selects the MVS operator’s console by ID number. This operand overrides the *nn* operand.

*conname* selects the MVS operator’s console by name.

CONS displays three types of information on the command line: the type of console (CONSOLE or MASTER CONSOLE), the console’s device number, and the console’s ID number (as in ID=3).

To see each MVS console’s number (also known as the UCMID), issue the D CONSOLES command at a real MVS console.

The CONS minor commands enable you to display selected types of messages from the console.
The CONS command also displays the output that MVS or JES commands generate when the OCMD authorized command issues them. (See Figure 36 on page 305 for an example.)

Note that OMEGAMON accesses the CONSOLE address space using cross memory services each time you ask for a display. It uses a small amount of memory in the MVS common area (approximately 2K) for the period that the CONS command is on the screen.

Figure 36 on page 305 is an example of an OMEGAMON screen that uses the MVS console support.

**FIGURE 36. OCMD and CONS output**

```
>OCMD R 83,ABEND RC = 0
=================================================================
CONS01 MASTER CONSOLE 01F
line99  *STC  721 *82 DFS9961 *IMS READY* IP01
+  - STC 1113 $HASP395 IMSRDR ENDED
+  - STC 1113 $HASP250 IMSRDR IS PURGED
+  - JOB 1114 $HASP373 BMP01 STARTED - INIT 4 - CLASS I - SYS A430
+  - JOB 1111 $HASP395 TS0003G ENDED
+  + $HASP309 INIT 3 INACTIVE ******** C=I
+  + JOB 1111 $HASP250 TS0003G IS PURGED
+  - STC 721 DFS25001 *MDA00 IOP1
+  - DATABASE BE3PARTS SUCCESSFULLY ALLOCATED
+  - STC 721 DFS25001 *MDA00 IOP1
+  - DATABASE BE3PSID1 SUCCESSFULLY ALLOCATED
+  + JOB 1114 @83 DFS3125A PRIMER SAMPLE TEST, REPLY CONT, LOOP, ABEND, 0
+  + CANCEL JOB
+  + TSU 1115 $HASP100 TS0045 ON TSOMEGAMON/IMSNRDR
+  + TSU 1115 $HASP373 TS0045 STARTED
+  00 IEE600I REPLY TO 83 IS;ABEND
=================================================================
```

**CONU**

Locates the output buffer for an MVS operator console by device address.

**Type:** Major

**Format:** CONU xxxx

**Note:** CONU Command now accepts input in either a 3-digit or 4-digit format.

CONU functions like CONS, except that you supply the three-byte hex device address as an operand instead of the console ID. If you omit xxx, OMEGAMON automatically locates the output buffer for the master console. CONU displays the address and the console ID as shown:
The following minor commands display information about the selected console.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ACTN</strong></td>
<td>Displays only those lines from the screen that require some action.</td>
</tr>
</tbody>
</table>

**Type:** Minor of CONS and CONU

ACTN displays any line that begins with an asterisk (*). You can use this command to display any outstanding Mount or Reply messages that need attention. ACTN has no arguments.

<table>
<thead>
<tr>
<th><strong>LINE</strong></th>
<th>Displays the last ( nn ) lines from the console you select.</th>
</tr>
</thead>
</table>

**Type:** Minor of CONS and CONU

**Format:** LINEnn

To display the last 12 lines of the screen of console 11, enter this command:

```
CONS11   Master Console (ID=11)
line12
```

The LINE minor command displays only in-line messages, such as the display produced by the following command.

```
D A,L,L=Z
```

LINE does not display out-of-line messages, such as those produced by the following command.

```
D A,L,L=A
```

To avoid this situation, either use the \( L=Z \) operand where appropriate, or issue the following console control command to remove the out-of-line display areas:

```
K A,NONE
```

<table>
<thead>
<tr>
<th><strong>MNT</strong></th>
<th>Displays mount messages that require operator action.</th>
</tr>
</thead>
</table>

**Type:** Minor of CONS and CONU

MNT has no arguments.

For example, to display mount messages for console 4, enter the following command:

```
CONS04   Console 660 (ID=4)
mnt
+ 2000 09.07.47 JOB 2225 *13 IEC701D M 370,VOLUME TO BE LABELED CA1759
```
Collecting data about address spaces

This section describes the PEEK major command and its minors.

<table>
<thead>
<tr>
<th>PEEK</th>
<th>Collects information about a single address space.</th>
</tr>
</thead>
</table>

**Type:** Major

After you issue the PEEK command to collect information from the target address space, you can format and display this information with various PEEK minor commands.

**Format:** `aPEEK targ`

- **a** An action character in column 1:
  - - Specifies that new data be collected from the target address space.
  - < Specifies that the command be re-executed on succeeding cycles.
- **b** Enables minor commands to execute with previously collected data.

- **targ** The target address space. It can be:
  - `cccccccc` jobname
  - `nnnn` decimal ASID number
  - `*` OMEGAMON address space

For example, to gather data from a job named PAYROLL (with an ASID of decimal 25), enter:

- **-PEEK PAYROLL**

or

- **-PEEK 25**

PEEK accesses the PAYROLL address space, removes the action character from column 1, and displays the following:

```plaintext
PEEK PAYROLL ASID=25 >> OB8112: Data Collection Initiated <<
```

When PEEK collects the data from the target address space and stores it in the work area, it displays this information:

```plaintext
PEEK PAYROLL ASID=25, collected at 15:39:39
```

Any PEEK minor commands that you issue now examine this work area. Each time you issue PEEK with the action character in the label field, PEEK collects
current information. If you issue PEEK without an action character, it uses the data in the work area from the previous update.

Note that PEEK uses cross memory services to access the target address space for a job that runs non-swappable. For a swappable job, it uses an SRB routine. Since SRBs run at the highest priority and increase the swapping load, overuse can degrade performance.

The .SET command contains two keywords to set OMEGAMON profile parameters for the PEEK command. The keywords are LOOPCOUNT and PEEKSIZE.

**LOOPCOUNT**
Sets the maximum number of control blocks that the PEEK command tests before it detects a loop. The valid range is 1 to 60000.

The PEEK command traces control block chains. If OMEGAMON encounters a damaged target address space, some of the control blocks examined may have chained into a loop, and OMEGAMON issues a warning message. The warning may also appear when you chain through an address space that has a complex TCB structure. In this case, the cause may not be a loop, but rather the large amount of processing that is necessary to scan all of the TCBs.

**PEEKSIZE**
Sets the work area size (in bytes) for the PEEK command. The maximum is 204800.

The first time you use PEEK in an OMEGAMON session, it obtains a work area (32K by default) from the private area to hold the collected data. OMEGAMON gives you a warning message if the data does not fit within the work area.

Issue .SET and increase the value of PEEKSIZE = to increase the work area size. Then reissue the PEEK command and it will collect the data. You can save the new PEEKSIZE definition in a user profile.

The following minor commands can be used with PEEK:

<table>
<thead>
<tr>
<th>AMAP</th>
<th>Displays a map of virtual storage utilization within the private area.</th>
</tr>
</thead>
</table>

**Type:** Minor of PEEK

The AMAP minor command of PEEK displays a map of virtual storage utilization within the private area. This map indicates the maximum region available, the portion currently in use, and various areas within the region.
The AMAP display for XA and ESA shows all virtual storage, or you can limit the map to storage above or below the 16M line with an A or B argument as shown.

```
FIGURE 37. AMAP Display Format (XA and ESA)
```

```
PEEK USER01 ASID=46, collected at 15:39:39
amap <map all virtual storage>
amapA <map virtual storage above the 16M line> (XA and ESA)
amapB <map virtual storage below the 16M line>
```
Area descriptions for above the 16M line:

<table>
<thead>
<tr>
<th>Area Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Highest 31-bit address</strong></td>
<td>The highest possible address in 31-bit architecture.</td>
</tr>
<tr>
<td><strong>Top of extended private</strong></td>
<td>Highest address within the extended private area.</td>
</tr>
<tr>
<td><strong>ELSQA,SWA unallocated</strong></td>
<td>The amount of storage not currently allocated within the extended system area.</td>
</tr>
<tr>
<td><strong>Fragmented free space</strong></td>
<td>The amount of free storage within allocated pages of the extended system area.</td>
</tr>
<tr>
<td><strong>Current bottom of ELSQA,SWA</strong></td>
<td>Lowest address allocated within the extended private area for the extended system area.</td>
</tr>
<tr>
<td><strong>Avail. for ELSQA/SWA only</strong></td>
<td>The amount of unallocated storage between the current bottom of the extended system area and the limit of the extended user area.</td>
</tr>
<tr>
<td><strong>Extended User Area Limit</strong></td>
<td>Highest address possible for the extended user area.</td>
</tr>
<tr>
<td><strong>Avail. for ELSQA/SWA/USER</strong></td>
<td>The amount of unallocated storage between the extended user area limit and the current top of extended user area. Note that the extended system area can allocate storage within this area.</td>
</tr>
<tr>
<td><strong>Current Top of Ext. User Area</strong></td>
<td>The highest address currently allocated within the extended private area for the extended user area.</td>
</tr>
<tr>
<td><strong>Largest free block</strong></td>
<td>The largest contiguous piece of unallocated storage within the extended user area.</td>
</tr>
<tr>
<td><strong>Extended User unallocated</strong></td>
<td>The amount of storage not allocated within the extended user area.</td>
</tr>
<tr>
<td><strong>Fragmented free space</strong></td>
<td>The amount of free storage within allocated pages of the extended user area.</td>
</tr>
<tr>
<td><strong>Bottom of Extended Private</strong></td>
<td>The lowest address currently allocated within the extended private area for the extended user area.</td>
</tr>
</tbody>
</table>

Area descriptions for below the 16M line:

<table>
<thead>
<tr>
<th>Area Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Highest 24-bit address</strong></td>
<td>The highest possible address in 24-bit architecture.</td>
</tr>
<tr>
<td><strong>Top of Private</strong></td>
<td>Highest address below the common area (start of CSA).</td>
</tr>
<tr>
<td><strong>LSQA/SWA unallocated</strong></td>
<td>Total of contiguous 4K areas. The numbers include LSQA, SWA and subpools 229/230.</td>
</tr>
</tbody>
</table>
### Console-related Commands

#### DATA

<table>
<thead>
<tr>
<th><strong>Fragmented free space</strong></th>
<th>Total of areas within LSQA which are each less than the 4K available for allocation as defined by FQEs.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Current bottom of LSQA/SWA</strong></td>
<td>Lowest address allocated to LSQA/SWA subpools.</td>
</tr>
<tr>
<td><strong>Avail. for LSQA/SWA only</strong></td>
<td>Total space available for LSQA/SWA allocation. This includes the LSQA/SWA unallocated value and the amount of space in the region available area.</td>
</tr>
<tr>
<td><strong>User Area Limit</strong></td>
<td>Highest address available for user allocation (region size plus IEALIMIT).</td>
</tr>
<tr>
<td><strong>Avail. for LSQA/SWA/USER</strong></td>
<td>Amount of space available for problem program allocations, not including unallocated areas within the region used.</td>
</tr>
<tr>
<td><strong>Current top of User Area</strong></td>
<td>Highest address currently allocated for problem program use.</td>
</tr>
<tr>
<td><strong>Largest free block</strong></td>
<td>Largest contiguous area available within the region used.</td>
</tr>
<tr>
<td><strong>User unallocated</strong></td>
<td>Total of the contiguous 4K areas within the region used which are available for problem program use.</td>
</tr>
<tr>
<td><strong>Bottom of Private</strong></td>
<td>Lowest address within the private area (above the resident nucleus rounded up to the next 64K boundary).</td>
</tr>
<tr>
<td><strong>Prefixed Storage Area</strong></td>
<td>Fixed storage location starting with absolute zero.</td>
</tr>
</tbody>
</table>

**DATA**

Displays data space and Hiperspace utilization for a given address space.

**Type:** Minor of PEEK

The DATA minor of PEEK requires an APF-authorized environment. For data space information display, MVS/SP™ 3.1 (ESA) must be installed. For Hiperspace™ information display, DFP 3 must be installed.

Here is an example of the DATA minor display.

```
-PEEK TSO07 ASID=48 >> OB8112: Data Collection Initiated <<
data Name Type   Owning Task    Current Size Maximum Size
+ SDUMPSWA Basic CR8SPACE (007FE380) 64K       256K
+ MYHIPER Scroll HIPERPGM (007FD468) 1000K 4096K
+ HIPER2  Cache HIPERPGM (007FDA50)  40K    100K
+ SDUMPALL Basic DSPPGM1 (007ED900)  4K      4K
```
If you put an X in the label field, DATA displays extended information.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Owning Task</th>
<th>Current Size</th>
<th>Maximum Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDUMPSWA</td>
<td>Basic</td>
<td>CR8SPACE (007FE380)</td>
<td>64K</td>
<td>256K</td>
</tr>
<tr>
<td>MYHIPER</td>
<td>SCROLL</td>
<td>HIPERPGM (007FD468)</td>
<td>1000K</td>
<td>4096K</td>
</tr>
<tr>
<td>HIPER2</td>
<td>CACHE</td>
<td>HIPERPGM (007FDA50)</td>
<td>40K</td>
<td>100K</td>
</tr>
<tr>
<td>SDUMPALL</td>
<td>Basic</td>
<td>DPPGM1 (007ED900)</td>
<td>4K</td>
<td>4K</td>
</tr>
</tbody>
</table>

**Name**
Name of the data-only space.

**Type**
Type of data-only space. Valid types are basic, scroll, and cache.

**BASIC**
BASIC data space.

**SCROLL**
SCROLL-type Hiperspace.

**CACHE**
CACHE-type Hiperspace.

**Owning Task**
Program name and TCB address associated with the owning task.

**Size**
Current size of the data-only space in K.

**Max Size**
Maximum allowable size of the data-only space in K.

**Key**
Storage protect key of the data-only space.

**Fprot**
Storage fetch protection indicator of the data-only space.

**Dref**
Disabled reference storage indicator for the BASIC data space.

**Scope**
Specifies whether the BASIC data space is shareable (ALL) or non-shareable (SINGLE) with other address spaces.

**Castout**
Indicates whether the CACHE-type Hiperspace is being given special consideration when the system searches for pages to remove from expanded storage when a shortage arises.

**DDNS**
Displays information about allocated ddnames.

**Type:** Minor of PEEK

**Format:**

- `[b]DDNS[nn]`
- `[X]DDNS[nn]`
**Console-related Commands**

**b** Displays all ddnames allocated to a jobstep and their corresponding device addresses, dataset names, and volume serial numbers.

**X** Requests extended information. For each TCB group of ddnames, XDDNS shows:

- **LRECL** Logical record.
- **BLKSZ** Blocksize.
- **RECFM** Record format.
- **DSORG** Dataset organization.
- **PWD** Password protection. A blank indicates that no dataset password is in effect.
- **EXCP** Execute channel program. This figure represents the number of I/Os (EXCPs) issued.
- **TIOT** Task I/O table address.

**nn** Suppresses the first nn lines of the display. This option is useful if all of the ddnames do not fit on one screen.

Here is an example of the DDNS display:

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOBS</td>
<td>Displays values available from the private area.</td>
</tr>
</tbody>
</table>

**Type:** Minor of PEEK
The following screen shows a typical JOBS display.

When the **Programmer Name** field is blank in the job statement, the JOBS minor command does not display that line. JOBCLASS information only appears for batch jobs. The job class is the class originally associated with the job. If a JES command was requeued prior to execution, the original job class from the CLASS= parameter in the JCL JOB statement will appear.

**MODS** Displays information about the modules currently loaded into the user’s jobpack area.

**Type:** Minor of PEEK

The following screen display shows a typical example of the MODS command.

The name of each module currently in the user’s jobpack area appears along with its entry point address, length, use count, and load module attributes. For the definition of the attributes, see the *IBM MVS Linkage Editor Manual*. You can use the entry point address with the cross-memory list or zap (XMLS or XMZP) commands.

To suppress the display of the first *nn* or *nnn* modules, you can optionally specify a two- or three-digit number in the operand field of MODS (MODSnn or MODnnn). This is useful if all of the names do not fit on one screen.

**STEP** Displays private area storage utilization.

**Type:** Minor of PEEK
Console-related Commands

The following screen shows an example of the STEP command for MVS/XA and MVS/ESA.

```
PEEK USER01  ASID=46, collected at 15:39:39
step  Job Step Pgm: cccccccc; 5 TCBs, 3 datasets, and 3 Modules
+  Step started at 18:29:04 , now in step # 1 of 1
+  For the region below the 16M line:
+  Total private region = 8172K Unused = 3104K
+  Region requested = 5120K Region limit = 5184K
+  Low PVT in use = 4876K Unallocated = 204K Free = 40K
+  High PVT in use = 192K Unallocated = 24K Free = 42K
+  Start of SYSREG: 00001000 End of SYSREG: 00004FFF
+  Start of low PVT: 00005000 End of low PVT: 004FAFFF
+  Current top: 004FAFFF Limit of region: 00514FFF
+  Start of high PVT: 007D0000 End of high PVT: 007FFFFF
+
+  For the extended region above the 16M line:
+  Total private region = 2055168K Unused = 2046116K
+  Region limit = 32768K
+  Low PVT in use = 96K Unallocated = 4K Free = 7K
+  High PVT in use = 8956K Unallocated = 8948K Free = 6K
+  Start of low PVT: 02900000 End of low PVT: 02918FFF
+  Current top: 02918FFF Limit of region: 048FFFFF
+  Start of high PVT: 7F741000 End of high PVT: 7FFFFFFF
===========================================================================
```

cccccccc
The current program name of the topmost JOBSTEP TCB. (This value corresponds to what is in the EXEC statement, unless XCTL was used to transfer control to another load module.) The other counts indicate how many lines of output you may expect from the TCBS, DDNS, and MODS minor commands.

**Total private region**
The total size of the private area, including areas that can’t be allocated.

**Region requested**
The amount you specify on the REGION JCL parameter.

**Region limit**
The region limit that the IEALIMIT exit imposes.

**Low PVT**
The storage that the REGION parameter limits. This includes all of the user subpools.

**High PVT**
Includes LSQA, SWA, and subpools 229 and 230. This value is allocated from the top of the user’s region downward and is not limited by the REGION JCL parameter.

**In use**
The storage allocated to subpools.

**Unallocated**
The storage not allocated to subpools.

**Free**
The storage allocated to subpools but not currently GETMAINed.
The SUBP display consists of two parts for each TCB: one part shows detailed information on the allocation of storage and one part shows a summary of the virtual storage. The third section shows totals. Notice that the totals displayed include all subpools in the address space, whether or not the X label was entered in front of the SUBP command. Consequently, the totals could be larger than the sums of the private area statistics.

Note that the SUBP display shows only allocation statistics for private area subpools and not common area subpools such as Subpool 241.

Here is an example of the display (for both XA and ESA).

```
PEEK CPSTEP6A ASID=15, collected at 15:39:39
subp
+ SBP-K Alloc Real #Blks Addr Free #Blks Mxfree Program
  251-8  28K  28K  10 0000DD000  6184   3  000968 IFOX&zz Own
  0-8  2012K 2012K  6 0000E4000  5608   3  000AF8 Shr
  230-5   4K  1K  1 007CA000  3966   5  000488 Own
  237-1  188K  41K  39 007CC000  3156  38  000440 Shr
----------------------------------------------------------------------------
+ PVT-Hi:   4K  1K  1  3K  5
+ PVT-Lo:  28K  7K  3  4K  3
+ Subtot:  32K  8K  4  7K  8
----------------------------------------------------------------------------
+ Tot-Hi:  360K  78K  70  40K  68
+ Tot-Lo:  2044K 29K  8  13K  7
+ Totals: 2404K 107K  78  53K  75
----------------------------------------------------------------------------
```
The first portion of the display for each TCB shows detailed information about the subpool:

| SBP-K | Subpool number and protect key (decimal). |
| Alloc | Amount of virtual storage currently allocated to the subpool (in 4K increments). The storage is not necessarily contiguous. |
| Real  | Amount of real storage backing the virtual allocation. |
| #Blks | Number blocks allocated to the subpool. |
| Addr  | Address of the block with the lowest address (hex). In this example, there are 10 blocks and the lowest starts at 000DD000 in subpool 251. |
| Free  | Number of free bytes (hex) within the subpool that no one has yet GETMAINed. Anything in a free area is available for a GETMAIN for the same subpool, but not for other subpools. |
| #Blks | Number of free non-contiguous blocks within the subpool, where each block can be any number of bytes (in 8-byte units). |
| Mxfree | Size (hex) of the largest free block within the subpool. |
| Pgmname | Program name of the TCB described associated with these subpools. |
| Own|Shr | Allocations marked SHR are displayed for each TCB that shares the allocation. Allocations marked OWN appear only for the owning TCB. |

The next portion of the display for each TCB is a summary of the virtual storage allocated.

| PVT-Hi | Summary of LSQA allocated for the address space. |
| PVT-Lo | Summary of user storage allocated for the address space. |
| Subtot | Summary of LSQA and user virtual storage allocated. |

The final section, which appears after all TCBs are listed, is a summary for the Private Area:

| Tot-Hi | Summary of LSQA allocated. |
| Tot-Lo | Summary of user virtual storage allocated. |
| Totals | Summary of LSQA and user virtual storage allocations. |

This example shows that the program IFOX00 currently has 28K bytes of storage allocated to subpool 251, key 8. It is allocated in three blocks and the storage defined by the last block begins at DD000. Of all subpool 251 storage, EF0 bytes are free (not in use). The EF0 freebytes are made up of three blocks. The largest of these blocks is 968 (hex) bytes long.

| TCBS | Displays the current TCB structure for the target user. |
This next screen shows a typical TCBS display.

```
PEEK PAYROLL ASID=25, collected at 15:39:39
==============================================================================
tcbs  Program  Mother  Daughter  Sister  Jobstep
+    IEAVAR00  IEESB605  IEFIIC    IEAVTSDT (SELF) (Region Control Task)
+    IEFIIC  IEESB605  PAY1             (SELF) (Initiator)
+    PAY1     IEFIIC                      (SELF)
+    IEAVTSDT IEAVAR00                    (SELF) (SVC Dump Task)
==============================================================================
```

This is a typical TCB structure for a batch job.

**Program**
Load module name of the most recently created RB for each TCB. In this case, PAY1 indicates the name on the EXEC PGM= parameter.

**Mother Daughter Sister**
Program names for the mother, daughter, and oldest sister TCBs of the Program TCB.

Note that most address spaces on this level point to themselves as the Jobstep TCB. You can use this information to easily construct a picture of the current TCB structure:

**FIGURE 38. Example of TCB structure**
If you place an A in the label field of TCBS, OMEGAMON displays the actual TCB address under each TCB program name. This information may be useful if several TCBs in the same step invoke the same program.

If you place an X in the label field of TCBS, two extra lines appear for each TCB. The first line shows the storage protect key for the TCB and indicates whether the address space is APF authorized.

The second line indicates either DISPATCHABLE or NON-DISPATCHABLE. For those TCBs that cannot be dispatched, a short explanation appears to indicate which non-dispatchability bit was found set.

For example, the following message indicates that the TCB is merely waiting to post an ECB.

**NON-DISPATCHABLE: TOP RB WAITING ON ECB**

This is by far the most common reason for a TCB to be non-dispatchable.

You may optionally specify a 2-digit number in the operand field of TCBS to suppress the display of the first nn TCBs. This number is useful if all of the TCBs do not fit on one screen.

---

**Important**

If the job being PEEKed has many TCBs, you may need to increase the maximum number of control blocks tested with the LOOPCOUNT keyword of the .SET command.
Online Facility for Logging IMS Messages

Introduction

OMEGAMON contains a facility to copy IMS MTO messages and IMS-related MCS (multiple console support) messages into SYSOUT files.

This facility, known as the OMEGAMON message logging facility, allows you to browse current IMS messages with IBM’s SDSF® or similar products, to print IMS messages, or to create permanent historical copies of IMS messages using the external writer facilities of JES.

The message logging facility does not require any additional DD statements in the OMEGAMON procedure. All SYSOUT message log files are dynamically allocated. The SYSOUT class to be used for the message log files is defined during message logging facility startup.

Installation

Although the message log file does not require any additional installation steps, some planning should be done to make effective use of the facility.

The user must select a JES SYSOUT class for use by the message logging facility. External writer procedures can then be created to capture SYSOUT files written to that class by the message logging facility. However, the user should ensure that the selected SYSOUT class is used exclusively by the message logging facility. This is to avoid collision between message logging facility output and other miscellaneous system output during the external writing process.

The user must also determine the destination of output captured by the external writer procedure. This output may go to tape, generation data groups, sequential disk datasets, or microfiche. This destination should be specified in the external writer task used to capture output from the message logging facility.

You can start the external writer task at any time. It is advisable to start when output from the message logging facility should be removed from the JES spool.

A sample external writer procedure is shown in the following figure.

**FIGURE 39. Typical JCL for external writer procedure**

```
//MLOGXWTR PROC
//IEFPROC EXEC PGM=IASXWR00,REGION=64K,
//          PARM='PW' <<<<<<------------ CLASS W IS FOR MLOG
//IEFRDER DD  DSN=CANDLE.MLOG,DISP=(MOD,PASS),
//         DCB=(BLKSIZE=13030,LRECL=137,BUFL=13030,BUFNO=2,RECFM=VBM),
//         UNIT=TAPE,VOL=SER=TVOL01
```
To start an external writer task, issue the following (or similar) command at an MVS console:

```
START MLOGXWTR.MLOGWTR
```

To stop the external writer task, issue the following (or similar) command at an MVS console:

```
STOP MLOGWTR
```

For further information about the external writer facility, consult IBM’s OS/VS2 MVS System Programming Library: Job Management.

### MLOG command

The MLOG authorized command examines or alters the online facility for logging IMS messages.

<table>
<thead>
<tr>
<th>MLOG</th>
<th>Displays or alters status of facility for logging IMS messages.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `-MLOG opr`

The operand (`opr`) is one of the following:

**FLUSH=n**

If specified, this operand sets up the automatic FLUSH threshold. When the message logging facility reaches this threshold, it will automatically close the SYSOUT dataset, make it available to the external writer task, and reallocate it.

The `=n` portion of the operand is optional. When `=n` is present, `n` can be a number between 0 and 65,535 inclusive. When `n` is set to zero, all flush processing is cancelled immediately. When `n` is set to a number between 1 and 65,535, flush processing occurs after the message logging facility logs `n` messages.

If `=n` is not specified, the FLUSH operand causes the current SYSOUT message log file to be closed and reallocated immediately by the message logging facility. This makes the message log file available to be processed by the external writer task.

**START**

The START operand causes the message logging facility to start.

The format is:

```
-MLOG START, CLASS=c
```

The optional CLASS suboperand specifies the default SYSOUT class (c) for the message log. If you do not specify an output class with the CLASS suboperand, it defaults to your installation’s default output class.
**STOP**  
The STOP operand causes the message logging facility to initiate shutdown processing. The current message log file will be closed and made available to the external writer task. The message logging facility will then be rendered inactive.

**STATUS**  
The STATUS operand displays the status of the message logging facility. This is the default.

The status provided includes the following information:

- The status of the message logging facility:
  - active
  - inactive
  - disabled
  - pending critical operations
- The SYSOUT class.
- The number of records written to the current SYSOUT dataset by the message logging facility since startup or since the last message log file flush.
- The automatic flush threshold, if automatic flushing is in effect.

This example shows a typical MLOG command:

```
-MLOG START, CLASS=D
```
Dynamically Allocating and Unallocating Datasets

Introduction

Use the following commands to dynamically allocate or unallocate a dataset.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DYNA</td>
<td>Dynamically allocates a dataset.</td>
</tr>
</tbody>
</table>

**Type:** Immediate

You must execute DYNA twice to allocate a dataset: once to specify the dataset name and once to allocate the dataset.

**Format:** aDYNA c...c

- **a** Specifies the action of the command. Possible values are:
  - **b** Defines the dataset to be allocated. If this field is blank, c...c must specify the dataset allocation parameters.
  - **-** Allocates the dataset. If this field contains the action character (-), c...c must be blank.

- **c...c** Specifies the dataset to be allocated, using the following parameters:
  - **DD=ccccccc** Specifies the ddname.
  - **UNIT=c** Specifies the unit name. An asterisk (*) allocates a cataloged dataset.
  - **VOLSER=cccccc** Specifies the volume serial number. To allocate a cataloged dataset, enter an asterisk (*).
  - **DISP=** Specifies the initial disposition of the dataset (OLD, SHR, or MOD).
  - **DSN=cccccccc** Specifies the dataset name.

If this field contains the dataset description, the label field (a) must be blank.
If this field is blank (if the dataset name has already been specified), you must specify the action character (-) in the label field to allocate the dataset.
In the next example, DYNA specifies a cataloged dataset, with a ddname of MYDD, an initial disposition of SHR, and a dataset name of MY.DATASET.

\[ \text{DYNA DD=MYDD,UNIT=*,VOLSER=*,DISP=SHR,DSN=MY.DATASET} \]

Enter DYNA again to allocate the dataset specified above:

\( \text{-DYNA} \)

---

Dynamically deallocates a dataset.

**Type:** Immediate

You must execute DYNU twice to deallocate a dataset: once to specify the dataset name and once to deallocate the dataset.

**Format:** \[ a\text{DYNU}\; c...c \]

- **a** Specifies the action of the command. Possible values are:
  - **b** Defines the dataset to be deallocated. If this field is blank, c...c must specify the dataset name.
  - **-** Deallocates the dataset. If this field contains the action character (-), c...c must be blank.

- **c...c** Specifies the dataset to be deallocated:
  - **NAME=ddname** Specifies the dataset name. If this field contains the dataset name, the label field (a) must be blank.
  - **b** If this field is blank (if the dataset name has already been specified) you must specify the action character (-) in the label field (a) to deallocate the dataset.

In the next example, DYNU specifies a cataloged dataset, with a ddname of MYDD.

\[ \text{DYNU DD=MYDD} \]

Enter DYNU again to deallocate the dataset specified above:

\( \text{-DYNU} \)
Displaying and Modifying Data Space and Hiperspace Storage

Introduction

OMEGAMON provides commands to display or modify IMS data space and hiperspace storage for systems that have used the capabilities of MVS/ESA and DFP 3.1 to create these data-only spaces. The commands in this section parallel the functions of the MLST, MCHN, MSCN, and MZAP storage commands.

Because of the potential security risk associated with using these commands, the .DSA command exists to provide an extra level of protection.

<table>
<thead>
<tr>
<th>.DSA</th>
<th>Sets and displays authorization to list and/or zap non-shareable data-only spaces.</th>
</tr>
</thead>
</table>

**Type:** Immediate

The .DSA command provides a mechanism to limit the scope of the listing and zapping commands to shareable data-only spaces (data spaces or hiperspaces that have been defined by the owner as able to be shared by other address spaces).

**Command operands:**

- **ON**  
  Turns on data-only space authorization, (access is allowed to all data-only spaces).

- **OFF**  
  Turns off data-only space authorization, that is, access is restricted to shareable spaces only.

Entering .DSA with no operand displays the current status of data-only space authorization.

<table>
<thead>
<tr>
<th>OSPC</th>
<th>Lists the attributes of the owner of a data-only space.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** OSPC spacename

where spacename is the name of the data-only space you want to list. If you do not enter a name, OSPC will list all data spaces and hiperspaces. You can also enter any number of characters from 1 to 7, and OSPC will display any space names beginning with the character string entered.
OSPC provides the following information about the specified data space or hiperspace:

- type of data-only space
- ASID of owning TCB
- jobname of owning TCB
- address of owning TCB

The following is an example:

```
>OSPC MYSPACE
+ TYPE ASID JOBNAME TCB address SPACE NAME
+ Data space 12 MYJOB 007FFA10 MYSPACE
+ Hiperspace 22 HISJOB 007B7CB0 MYSPACE
```

### SLST
Displays bytes of memory from data-only space storage.

**Type:** Immediate  
**Format:**  
```
aSLSTc jobname,spacename,addr,plen
```

-a A required action character in column 1:

- Changes to a comment character (>) after command executes.
- < Does not change to a comment character after command executes. Use this action character to repeat the command.

-.c Specifies the format of the output:

- B or b dump format (default)
- C character only
- X hex only

**jobname** The jobname or ASID in decimal of the owner of the data-only space.

**spacename** The name of the data-only space.

**addr** The starting address of the data.

**plen** The number (1 to 8 hex digits) of bytes to print. The default is 16 (X'10') bytes or 1 line.
SLST will list memory from data-only spaces. When necessary, an SRB will be scheduled into the address space of the TCB owning the data-only space to be listed.

Here is an example of using SLST.

```
<SLST MYJOB,MYSPACE,1000,20
+Storage at 00001000 in dataspace MYSPACE, job MYJOB ASID=12
+ 0000 E3C5E2E3 40C4C1E3 C140E2D7 C1C3C540 *TEST DATA SPACE *
+ 0010 F0F1F2F3 F4F5F6F7 F8F9C1C2 C3C4C5C6 *0123456789ABCDEF*
```

<table>
<thead>
<tr>
<th>SSCN</th>
<th>Scans data-only space storage for the occurrence of a specific string of data.</th>
</tr>
</thead>
</table>

**Type:** Immediate

**Format:** `aSSCNc jobname,spacename,addr,string,len1,len2`

- `a` A required action character in column 1:
  - Changes to a comment character (>) after command executes.
  - `<` Does not change to a comment character after command executes. Use this action character to repeat the command.

- `c` Specifies the format of the output:
  - `B` or `b` dump format (default)
  - `C` character only
  - `X` hex only

- `jobname` The jobname or ASID in decimal of the owner of the data-only space.

- `spacename` The name of the data-only space.

- `addr` The starting address of the scan. See “Address Specification for Storage Commands” for additional options on specifying `addr`.

- `string` The comparison string for the scan. Either a hexadecimal string or a character string. Use single quotes around a character string. Do not use quotes around a hexadecimal string.
SSCN will scan data-only space storage until a match to the string is found or the length of storage specified is exhausted. When necessary, an SRB will be scheduled into the address space of the TCB owning the data-only space to be scanned.

Following is a sample SSCN display.

```
>SSCN MYJOB,MYHIPER,1000,'TEST',200,20
+Storage at 00001100 in hiperspace MYHIPER, job MYJOB ASID=12
+ 0000 B3C5E2E3 40C8C9D7 C5D9E2D7 C103C540 *TEST HIPERSPACE *
+ 0010 F0F1F2F3 F4F5F6F7 F8F9C1C2 C3C4C5C6 *0123456789ABCDEF*
```

### SCHN

<table>
<thead>
<tr>
<th>SCHN</th>
<th>Scans data-only space control blocks for a string of data and displays the location.</th>
</tr>
</thead>
</table>

**Type:** Immediate

This command is used to search chained control blocks located in a data-only space for the occurrence of a specific string of data.

**Format:** `aSCHNc jobname,spacename,addr,string,off1,off2,plen`

- **a** A required action character in column 1:
  - `>` Changes to a comment character (>) after command executes.
  - `<` Does not change to a comment character after command executes. Use this action character after command executes.

- **c** Specifies the format of the output:
  - `B` or `b` dump format (default)
  - `C` character only
  - `X` hex only

- **jobname** The jobname or ASID in decimal of the owner of the data-only space.

- **spacename** The name of the data-only space.
SCHN scans data-only space storage until either a match to the string is found, the chain loops, or the address of the next control block is zero. When necessary, an SRB will be scheduled into the address space of the TCB owning the data-only space to be scanned.

This example shows the SCHN display for a data space.

```plaintext
>SCHN MYJOB,MYSPACE,1000,'TEST',0,30,20
+Storage at 00001100 in dataspace MYSPACE, job MYJOB ASID=12
+ 0000 E3C5E2E3 40C4C1E3 C140E2D7 C1C3C540 *TEST DATA SPACE *
+ 0010 F0F1F2F3 F4F5F6F7 F8F9C1C2 C3C4C5C6 *0123456789ABCDEF*
```

### SCHN

**addr** The starting address of the scan. See “Address Specification for Storage Commands” for additional options on specifying addr.

**string** The comparison string for the scan. Either a hexadecimal string or a character string surrounded by single quotes.

**off1** The offset from the beginning of the control block to the location of the comparison string. This value may be preceded by a + or - sign.

**off2** The offset from the beginning of the control block to the fullword address of the next control block. This value may be preceded with a + or - sign.

**plen** The length of print display. Default is one line or 16 (X'10') bytes.

SCHN scans data-only space storage until either a match to the string is found, the chain loops, or the address of the next control block is zero. When necessary, an SRB will be scheduled into the address space of the TCB owning the data-only space to be scanned.

This example shows the SCHN display for a data space.

### SZAP

**Type:** Immediate

**Format:** `-SZAP jobname,spacename,addr,vercode,repcode`

- Action character required for execution.

  **jobname** The jobname or ASID in decimal of the owner of the data-only space.

  **spacename** The name of the data-only space.

There is a potential integrity exposure when using SZAP on hiperspaces. SZAP will use HSPSERV to read in a page of data from the target hiperspace, check the data, alter the data, and finally use HSPSERV to write the page back to the hiperspace. If someone else is writing to the same page of the hiperspace while this process is occurring, the newly entered data could be lost. There is no available enqueue mechanism to guard against this exposure.
The lengths of vercode and repcode must match.

When necessary, an SRB will be scheduled into the address space of the TCB owning the data-only space to be zapped. Here is an example of using SZAP.

```
> SZAP MYJOB,MYSpace,1000,00000005,00000008
>             >> OB7110: Memory Zap Successful <<
```
# Automating and Logging Features

## Appendix overview

OMEGAMON can automatically invoke displays, initiate action, and log information in response to exception conditions or at specific times. When an automated sequence finishes, OMEGAMON resumes normal operation without manual intervention. There are three event- and time-driven features:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>XLF (exception logging facility)</strong></td>
<td>This feature automatically time-stamps and logs exception messages for your review. It enables you to correct intermittent performance problems by documenting the frequency and severity of systemwide exceptions.</td>
</tr>
<tr>
<td><strong>ASF (automatic screen facility)</strong></td>
<td>This feature automatically invokes a predefined screen space when a given exception occurs for more than a specified number of successive cycles. The predefined screen space can contain commands to turn on the log, further evaluate the exception condition, and perform other options.</td>
</tr>
<tr>
<td><strong>TSF (timed screen facility)</strong></td>
<td>This feature automatically invokes screen spaces at times or time intervals you specify. Many sites use TSF to spin off copies of the REPORT and/or XLFLG files to the printer. In general, you can use the TSF facility to automate many day-to-day housekeeping routines.</td>
</tr>
</tbody>
</table>

These three event- and time-driven features require OMEGAMON to be running in dedicated or VTAM automatic update mode.

## Appendix contents

- Controlling Automating Features ........................................... 332
- Setting Parameters for XLF and ASF Operation ..................... 334
- Setting Parameters for TSF Operation ............................... 336
- Logging Parameters for XLF, ASF, and TSF .......................... 339
- Using Screen Spaces in ASF and TSF ................................ 340
Controlling Automating Features

Introduction

It is important for you to be familiar with how exception analysis works and how to create screen spaces before you try to use the event- and time-driven features. This section explains how to access the commands that control these features.

If you are operating OMEGAMON with the menu interface, choose the PROFILE option from the Main Menu. From there, you can choose the following options:

- options listed under EXCEPTIONS, to define XLF and ASF parameters for each individual exception
- BACKGROUND to control the operation of the three background processing features, as well as define entries for TSF
- LOGGING to turn the log on and off, and to spin the log out to the printer

The following commands control the XLF, ASF, and TSF features.

XLF commands

The commands that you need to set up and operate XLF are:

**XACB**
Sets parameters and controls the operation of XLF for each individual exception. The XLF keywords are explained in “Setting Parameters for XLF and ASF Operation” on page 334.

**OPTN**
Turns the XLF feature ON and OFF (with the XLF keyword).

**/XLFOUt**
Sends the XLF data to the JES output queue and reallocates the output file. If you want to save the command in a screen space, use the .XLFOUt immediate command instead of the /XLFOUt INFO-line command.

ASF commands

The commands that you need to set up and operate ASF are:

**XACB**
Sets parameters and controls the operation of ASF for each individual exception. The ASF keywords are the same as the XLF keywords and are explained in “Setting Parameters for XLF and ASF Operation” on page 334.
TSF commands

The commands that you need to set up and operate TSF are:

OPTN          Turns the ASF feature ON and OFF (with the ASF keyword).
/LOGOUT       Sends the REPORT data to the JES output queue and reallocates the output file. If you want to save the command in a screen space, use the .LOGOUT immediate command instead of the /LOGOUT INFO-line command.

.TSF           Defines entries for TSF. This command is described in “Setting Parameters for TSF Operation” on page 336.
OPTN          Turns the TSF feature ON and OFF (with the TSF keyword).
/LOGOUT       Sends the REPORT data to the JES output queue and reallocates the output file. If you want to save the command in a screen space, use the .LOGOUT immediate command instead of the /LOGOUT INFO-line command.
Setting Parameters for XLF and ASF Operation

Introduction

The parameters for XLF and ASF operation are set with the XACB command. XACB allows you to activate the XLF and ASF features for all occurrences of any given exception, or only if the exception persists for a specified number of cycles. You can also set a limit on the number of times a given exception invokes the XLF or ASF feature.

The following figure shows the format of the XACB command. The variable cccc is the exception name. To dynamically set parameters for an exception, type over the current value displayed with the XACB command.

<table>
<thead>
<tr>
<th>XACB LIST=cccc</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ DISPLAY Parameters:</td>
</tr>
<tr>
<td>: cccc</td>
</tr>
<tr>
<td>: State=</td>
</tr>
<tr>
<td>: Group=</td>
</tr>
<tr>
<td>: Bell=</td>
</tr>
<tr>
<td>+ BOX Parameters:</td>
</tr>
<tr>
<td>: Boxchar=</td>
</tr>
<tr>
<td>: Boxclr=</td>
</tr>
<tr>
<td>: Boxattr=</td>
</tr>
<tr>
<td>+ THRESHOLD Parameters:</td>
</tr>
<tr>
<td>: Threshold=</td>
</tr>
<tr>
<td>: Attribute=</td>
</tr>
<tr>
<td>+ CYCLE Parameters:</td>
</tr>
<tr>
<td>: ExNcyc=n</td>
</tr>
<tr>
<td>: Stop=n (m)</td>
</tr>
<tr>
<td>: Cumulative=n</td>
</tr>
<tr>
<td>+ XLF Parameters:</td>
</tr>
<tr>
<td>: Auto=</td>
</tr>
<tr>
<td>: Log=</td>
</tr>
<tr>
<td>: Limit=nn</td>
</tr>
<tr>
<td>+ BOX Parameters:</td>
</tr>
<tr>
<td>: Repeat=YES</td>
</tr>
<tr>
<td>: Persist=nn</td>
</tr>
<tr>
<td>: SS=</td>
</tr>
</tbody>
</table>

The XLF and ASF parameters are:

**AUTO** Controls the status of ASF for this exception (ON/OFF).

**LOG** Controls the status of XLF for this exception (ON/OFF). It does not affect logging for ASF.

**LIMIT** Limits the number of times \( mn \) XLF and/or ASF is invoked if the exception occurs. If you specify Limit=00, no events are logged. If you specify Limit=NONE, XLF and/or ASF are invoked each time the exception occurs. The parenthetical number to the right of this parameter indicates the remaining number of times that the exception will be logged. You can reset the LIMIT parameter to continue logging the exception.

**REPEAT** Used with the PERSIST threshold, Repeat=YES specifies that XLF logging or ASF action occurs each time the PERSIST threshold is reached. For example, if Persist=5, the exception condition persists for 15 cycles, and logging is in effect, then the message would be logged three times. If Repeat=NO, the message would be logged only once as specified with the PERSIST parameter.
**PERSIST**

Logs the exception message and/or invokes the ASF screen spaces when the condition persists for \( nn \) consecutive OMEGAMON cycles. After it has reached the threshold, the message is logged only once (or ASF is invoked only once), unless the condition stops for at least one cycle and then trips again. If you specify \texttt{Persist=00}, no events are logged. The default is 0 cycles.

**SC**

You can dynamically alter the second character of this keyword to either an S or an L. You must specify this parameter for ASF to work.

- **SL**
  
  Specifies the screen space to invoke if ASF is in effect (\texttt{Auto=ON}), and specifies that the output of the ASF screen spaces is to be logged. The OMEGAMON REPORT log automatically turns on when the exception trips, and screen space logging starts.

- **SS**
  
  Specifies the screen space to invoke if ASF is in effect. It does not turn on the log.
Setting Parameters for TSF Operation

Introduction

The Timed Screen Facility (TSF) schedules certain screen spaces not on an exception basis, but rather at specified times of day or at specified intervals. For example, you can turn on bottleneck analysis automatically at 2:00 PM every weekday, and then turn it off again at 2:15. You can also invoke a screen space at regular intervals, such as every hour.

To use TSF, follow this procedure:

1. Create any screen spaces you want TSF to invoke.
   You can use the .SGO or .FGO command to chain screens together, and .RTN to end the cycle. Special considerations for creating screen spaces in TSF are discussed later in this chapter.

2. Turn on TSF with the TSF keyword of the OPTN command.

3. Set your terminal in automatic update mode using the .AUPON command.
   Do not enter any more commands.

Caution

Entering commands interrupts a TSF cycle, so that screen spaces that should execute on that cycle won’t execute.
To view and change TSF definitions:

1. Use the .TSF00 command to display the names of screen spaces and the times or time intervals when you want them invoked.

The .TSF00 command displays the current status of the TSF feature (ON/OFF), and lists all entries. The TSF table is shipped with 99 blank entries. For example:

<table>
<thead>
<tr>
<th>.TSF00</th>
<th>ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Time=0000  SS=<em>NONE</em>  DAY=DAILY</td>
</tr>
<tr>
<td>2</td>
<td>Time=0000  SS=<em>NONE</em>  DAY=DAILY</td>
</tr>
<tr>
<td>3</td>
<td>Time=0000  SS=<em>NONE</em>  DAY=DAILY</td>
</tr>
<tr>
<td>4</td>
<td>Time=0000  SS=<em>NONE</em>  DAY=DAILY</td>
</tr>
</tbody>
</table>

Enter .TSFnn to display entry nn in the TSF table. To define an entry, type TSF followed by the number of the entry.

For example, type .TSF01 to produce:

```plaintext
.TSF01 Time=0000  SS=*NONE*  DAY=DAILY
```

You can type the new entry over the current entry, press Enter, and the value is reset.

If you want to change an entry that doesn’t appear on the physical screen, you can specify an argument to skip nn entries. Type .TSF00 20 to display entries 21 through 99 in the TSF table, skipping the first 20.

Following are the keywords and valid entries:

**Time**

Specifies the time of day (from 0000 to 2400) to invoke the screen. TIME=+nn invokes the screen every nn minutes.

*Note: The screen space will not execute while the TIME=+nn entry remains on your current screen.*

**SL or SS**

Specifies the screen space to invoke if TSF is in effect. SS specifies the screen space to invoke, but does not turn on the log. SL automatically turns on the REPORT log when the exception trips, and screen space logging starts. You must specify either SS or SL for TSF to work.

**DAY**

The valid entries for day of week are MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY, WEEKDAY, WEEKEND, and DAILY. The days of the month are also valid entries (numerals 1 through 31). The default value is DAILY. You can abbreviate the input as long as it is unique, and as long as the day of the week is recognized.

You can specify day combinations by enclosing the names of the days within parentheses, and by separating each day with either a comma or a blank.
This example shows five TSF entries:

<table>
<thead>
<tr>
<th>TSF01</th>
<th>1</th>
<th>TIME=1800 SL=WENDSHFT DAY=DAILY</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSF02</td>
<td>2</td>
<td>TIME=0600 SS=WEEKSTRT DAY=MONDAY</td>
</tr>
<tr>
<td>TSF03</td>
<td>3</td>
<td>TIME=1200 SL=NOONTIME DAY=(TH,F)</td>
</tr>
<tr>
<td>TSF04</td>
<td>4</td>
<td>TIME=0800 SS=MONTHEND DAY=30</td>
</tr>
<tr>
<td>TSF05</td>
<td>5</td>
<td>TIME=+30 SL=STATUS DAY=DAILY</td>
</tr>
</tbody>
</table>

These entries perform the following functions:

**TSF01**
Executes and logs screen space WENDSHFT, at 6:00 PM daily.

**TSF02**
Executes screen space WEEKSTRT, at 6:00 AM every MONDAY.

**TSF03**
Executes and logs screen space NOONTIME, at 12:00 PM every Thursday and Friday.

**TSF04**
Executes screen space MONTHEND, on the 30th of each month at 8:00 AM.

**TSF05**
Executes and logs screen space STATUS, every 30 minutes, every day.
Logging Parameters for XLF, ASF, and TSF

Introduction

Exception messages that trip when XLF is in effect are routed to the OIXLFLOG file. Screen space output from ASF and TSF is routed to the OIREPORT file. You can display or change the logging parameters for XLF, ASF, and TSF with the OUTP major command and its minors. For XLF, specify OUTP XLFLOG. For ASF or TSF, specify OUTP REPORT.

XLFLOG and REPORT messages

In XLF and ASF processing, only one record is written to the log while a given exception condition persists. However, new records will be written to the XLFLOG or the REPORT log if any of the following situations occur:

- If an exception disappears for even one cycle, and then reappears, it is considered a new event.
- If you turn a given exception off and then on again, you clear the event. If the exception condition still exists, another record goes to the XLF log. OMEGAMON does not check for the condition unless the exception is turned on.
- If you turn off XLF or ASF and then turn it on again, you clear all events and new records go to the log.
- For a given exception, if you set the XACB REPEAT parameter to YES and specify a threshold for the PERSIST parameter, a new record is written each time the PERSIST threshold is reached.
Using Screen Spaces in ASF and TSF

Introduction

The power of ASF and TSF lies in their ability to branch to, execute, and log an analysis screen (or series of screens). The screen space you specify to be called when ASF is activated can contain whatever information-gathering and/or action-taking commands you specify. You might include commands to turn on the log, change OMEGAMON defaults, further analyze the exception condition, or even call other screen spaces (with .SGO or .FGO).

This section contains information on how to use and execute screen spaces, and how to use the .RTN command.

How ASF and TSF screen spaces execute

If two exceptions occur at the same time, screen spaces and logging execute for the first exception until the final screen space issues the .RTN command. If the second exception persists, the automatic mode reactivates on the next cycle.

If XIMS appears on any of your target screen spaces, any triggered exception messages appear as usual, but ASF ignores these new exceptions until the sequence is ended with the .RTN command. If the exception condition still exists, ASF proceeds to the next exception in sequence.

If you want to cancel an ASF or TSF sequence while it is running, press any key other than a cursor key. If, for example, you press Enter, the sequence will immediately terminate. However, when it terminates without a .RTN command, OMEGAMON does not return to the original calling screen, or to another screen defined with .RTN.

You can invoke either named or numbered screen spaces with ASF and TSF. By assigning a screen to a PF key in virtual storage, you assure availability even when the I/O subsystem is not functioning due to a problem.

Note that TSF sequences always function at their scheduled times, except when an ASF sequence is also tripped. ASF sequences take precedence over TSF. That means that if an ASF sequence trips while a TSF sequence is in progress, the ASF sequence cancels the TSF sequence. When the ASF sequence is complete, OMEGAMON returns to the screen displayed at the time that ASF tripped, but the TSF sequence does not continue. Similarly, if a TSF sequence is scheduled while an ASF sequence is already in progress, the TSF sequence is ignored.

Using the .RTN command

You can branch to as many screen spaces as you want in ASF and TSF by using the .FGO or .SGO command to chain screens together. The .RTN immediate command is required at the bottom of the last screen space in an ASF or TSF sequence in order to terminate the sequence, return to the
original calling screen or branch to the next, and re-enable exception analysis for further automatic calls. We recommend that you use definition mode (/DEF ON) when creating screen spaces that contain the .RTN command.

While the .RTN command is normally used to return to the calling screen space, it also accepts an argument, and can force the return to a screen space that is not the calling screen. For example, to return to a screen called SCREEN2, enter the .RTN immediate command followed by SCREEN2.

You can delay the return for up to 35 cycles by placing the number of cycles to be delayed in the label field of .RTN (1-9 for numbers 1-9 and A-Z for numbers 10-35). The following command causes a return to SCREEN2 after 6 cycles:

6.RTN SCREEN2

On each cycle, OMEGAMON replaces the number in the label field with the next lower number. When the count reaches zero, OMEGAMON fetches SCREEN2.

ASF example

Remember that you must be running OMEGAMON in dedicated mode or VTAM mode with automatic update in effect. You must set the ASF keyword of the OPTN command to ON to enable the automatic screen feature.

Here are some parameters set with the XACB command for the DRDY exception.

<table>
<thead>
<tr>
<th>XLF Parameters:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto=ON</td>
</tr>
<tr>
<td>Log=OFF</td>
</tr>
<tr>
<td>Limit=3</td>
</tr>
<tr>
<td>Repeat=NO</td>
</tr>
<tr>
<td>Persist=5</td>
</tr>
<tr>
<td>SL=DEX01</td>
</tr>
</tbody>
</table>

In this example, when the exception condition exists for 5 cycles in a row, ASF invokes screen space DEX01. From this point on, an A appears in the far right portion of the INFO-line, which indicates that the current screen is part of an ASF sequence.
DEX01 might consist of the following sequence of commands:

```
> After 8 cycles, branch to screen space DEX02
8.SGO DEX02
======
> Reset the OMEGAMON cycle time to 15 seconds
.SET INTERVAL=15
======
> Start degradation analysis
IDEG
BEGN
======
> Display degradation analysis for performance group 2
PDEX02
```

In the previous example, the OMEGAMON cycle time is set to 15 seconds. Because the 8.SGO entry waits 8 cycles before it jumps to DEX02, bottleneck analysis runs on this screen for 2 minutes and logs the results. At this point, screen space DEX02, shown in Figure 40 on page 342, is invoked.

**FIGURE 40. DEX02 screen space**

```
> Suspend degradation analysis
IDEG
SUSP
======
> Return to the calling screen
.RTN
```

This screen space suspends degradation analysis, and returns to the calling screen with the .RTN command. Upon return, the ASF sequence terminates (the A disappears from the INFO-line), enabling XIMS for further automatic calls. The .RTN command also automatically resets the interval to the one in effect when the ASF sequence started. You could, however, add an NR argument to .RTN (.RTNNR) to direct OMEGAMON to keep the new interval in effect.

Note that because the SL= parameter was used, ASF automatically turned on the REPORT log when the exception occurred (if it was not already on). When you leave automatic mode, ASF also turns off the log (if it was in the off state before the exception tripped).

When you use ASF to turn on the log automatically with SL=, OMEGAMON first logs the screen in use, and then branches to the scheduled screen space. This is done so that any exceptions can be logged before the ASF sequence begins.
The Candle Subsystem

Appendix overview

The Candle Subsystem is an MVS subsystem that enables OMEGAMON to monitor dynamic device activity in MVS/ESA SP4.2. The Candle Subsystem runs in its own address space, providing dynamic I/O device information to OMEGAMONs running in other address spaces.

The Candle Subsystem must be installed before OMEGAMON can report on the dynamic devices available to MVS/ESA SP4.2 users.

Appendix contents

Candle Subsystem Operation .............................................. 344
Starting and Stopping the Candle Subsystem ....................... 345
Candle Subsystem Operation

Introduction

When you install the Candle Subsystem, you define it to MVS as an MVS subsystem. This allows all Candle products that define themselves as subsystems to share a single subsystem address space.

To make Candle Subsystem functions available to Candle products as early as possible, you can start the Candle Subsystem automatically at IPL. Instructions for specifying automatic startup are contained in the Configuration and Customization Guide.

Subsystem interface module

Candle products use a Subsystem interface module, KCNDLI, to obtain information from the Subsystem’s address space. KCNDLI must be available to each Candle product that uses Subsystem functions. Instructions for installing the Subsystem interface module are contained in Installing Candle Products on MVS.
Starting and Stopping the Candle Subsystem

Introduction

The Candle Subsystem can be started automatically at IPL. Instructions for specifying automatic startup are contained in the Configuration and Customization Guide.

Starting the Candle Subsystem

To start the Candle Subsystem any time after IPL, use the START command from the operator console:

```
START KCNDL
```

where KCNDL is the name of the Subsystem startup procedure.

The following options are available when you start the Subsystem address space automatically or by operator command.

- **CNDPROC START Parameter**
  
  The optional CNDPROC parameter determines which Subsystem procedure file member is used during Subsystem initialization. The value of CNDPROC is a 2-character suffix which is used to form a complete start member name. The suffix is appended to KCNDL. Its default value is 00, so the default start member name is CNSTRT00.

- **RESTART START Parameter**
  
  The optional RESTART parameter forces the Subsystem to complete initialization, bypassing checks designed to prevent the start of a second address space.

Important

RESTART should only be used if the Subsystem address space terminates abnormally, and message **CNDL018I** appears. This message indicates that the Subsystem is already active. If RESTART is used when the Subsystem is already active, results are unpredictable.

The FORCE operand is required, as in the following example:

```
RESTART=FORCE
```

Stopping the Candle Subsystem

To stop the Candle Subsystem, use the STOP command from the operator console:

```
STOP KCNDL
```

where KCNDL is the name of the Subsystem startup procedure.
Starting and Stopping the Candle Subsystem
Introduction

Candle Corporation is committed to producing top-quality software products and services. To assist you with making effective use of our products in your business environment, Candle is also committed to providing easy-to-use, responsive customer support.

Precision, speed, availability, predictability—these terms describe our products and Customer Support services.

Included in this Guide to Candle Customer Support is information about the following:

Base Maintenance Plan .................................................. 348
  – Telephone Support
  – eSupport
  – Description of Severity Levels
  – Service-level objectives
  – Recording and monitoring calls for quality purposes
  – Customer Support Escalations
  – Above and Beyond

Enhanced Support Services ............................................. 352
  – Assigned Support Center Representative (ASCR)
  – Maintenance Assessment Services (MAS)
  – Multi-Services Manager (MSM)

Customer Support Contact Information .............................. 354
  – Link to Worldwide Support Telephone and E-mail information
Base Maintenance Plan

Overview

Candle offers a comprehensive Base Maintenance Plan to ensure that you realize the greatest value possible from your Candle software investments. We have more than 200 technicians providing support worldwide, committed to being responsive and to providing expedient resolutions to support requests. Technicians are available worldwide at all times during the local business day. In the event of an after-hours or weekend emergency, our computerized call management and forwarding system will ensure that a technician responds to Severity One situations within one hour. For customers outside of North America, after-hours and weekend support is provided in English language only by Candle Customer Support technicians located in the United States.

Telephone support

Candle provides consistently reliable levels of service—thanks to our worldwide support network of dedicated experts trained for specific products and operating systems. You will always work with a professional who truly understands your problem.

We use an online interactive problem management system to log and track all customer-reported support requests. We give your support request immediate attention by routing the issue to the appropriate technical resource, regardless of geographic location.

- **Level 0 Support** is where your call to Candle Customer Support is first handled. Your support request is recorded in our problem management system, then transferred to the appropriate Level 1 support team. We provide Level 0 manual interaction with our customers because we support more than 170 products. We feel our customers would prefer personal interaction to a complex VRU or IVR selection menu.

- **Level 1 Support** is the service provided for initial support requests. Our Level 1 team offers problem determination assistance, problem analysis, problem resolutions, installation assistance, and preventative and corrective service information. They also provide product usage assistance.

- **Level 2 Support** is engaged if Level 1 cannot provide a resolution to your problem. Our Level 2 technicians are equipped to analyze and reproduce errors or to determine that an error is not reproducible. Problems that cannot be resolved by Level 2 are escalated to Candle’s Level 3 R&D support team.

- **Level 3 Support** is engaged if a problem is identified in Candle product code. At Level 3, efforts are made to provide error correction, circumvention or notification that a correction or circumvention is not available. Level 3 support provides available maintenance modifications
and maintenance delivery to correct appropriate documentation or product code errors.

**eSupport**

In order to facilitate the support process, Candle also provides **eSupport**, an electronic full-service information and customer support facility, using the World Wide Web at [www.candle.com/support/](http://www.candle.com/support/). **eSupport** allows you to open a new service request and update existing service requests, as well as update information in your customer profile. New and updated service requests are queued to a support technician for immediate action. And we can respond to your request electronically or by telephone—it is your choice.

**eSupport** also contains a continually expanding knowledge base that customers can tap into at any time for self-service access to product and maintenance information.

The Candle Web Site and **eSupport** can be accessed 24 hours a day, 7 days a week by using your authorized Candle user ID and password.

**Description of Candle severity levels**

Responses to customer-reported product issues and usage questions are prioritized within Candle according to Severity Code assignment. Customers set their own Severity Levels when contacting a support center. This ensures that we respond according to your individual business requirements.

<table>
<thead>
<tr>
<th>Severity 1</th>
<th>Crisis</th>
<th>A crisis affects your ability to conduct business, and no procedural workaround exists. The system or application may be down.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity 2</td>
<td>High</td>
<td>A high-impact problem indicates significant business effect to you. The program is usable but severely limited.</td>
</tr>
<tr>
<td>Severity 3</td>
<td>Moderate</td>
<td>A moderate-impact problem involves partial, non-critical functionality loss or a reasonable workaround to the problem. A “fix” may be provided in a future release.</td>
</tr>
<tr>
<td>Severity 4</td>
<td>Low</td>
<td>A low-impact problem is a “how-to” or an advisory question.</td>
</tr>
<tr>
<td>Severity 5</td>
<td>Enhancement Request</td>
<td>This is a request for software or documentation enhancement. Our business units review all requests for possible incorporation into a future release of the product.</td>
</tr>
</tbody>
</table>
Candle has established the following service-level objectives:

<table>
<thead>
<tr>
<th>Call Status</th>
<th>Severity 1 Goal</th>
<th>Severity 2 Goal</th>
<th>Severity 3 Goal</th>
<th>Severity 4 Goal</th>
<th>Severity 5 Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Call Time to Answer</td>
<td>90% within one minute</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 1 Response (Normal Business Hours)</td>
<td>90% within 5 minutes</td>
<td>90% within one hour</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level 2 Response (Normal Business Hours)</td>
<td>Warm Transfer</td>
<td>90% within two hours</td>
<td>90% within eight hours</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scheduled follow-up (status update)</td>
<td>Hourly or as agreed</td>
<td>Daily or as agreed</td>
<td>Weekly or as agreed</td>
<td>Notification is made when an enhancement is incorporated into a generally available product.</td>
<td></td>
</tr>
</tbody>
</table>

The above information is for guideline purposes only. Candle does not guarantee or warrant the above service levels. This information is valid as of October 1999 and is subject to change without prior notice.

Recording and Monitoring Calls for Quality Purposes

Candle is committed to customer satisfaction. To ensure that our customers receive high levels of service, quality and professionalism, we'll monitor and possibly record incoming and outgoing Customer Support calls. The information gleaned from these calls will help us serve you better. If you prefer that your telephone call with Candle Customer Support in North America not be monitored or recorded, please advise the representative when you call us at (800) 328-1811 or (310) 535-3636.

Customer Support Escalations

Candle Customer Support is committed to achieving high satisfaction ratings from our customers. However, we realize that you may occasionally have support issues that need to be escalated to Candle management. In those instances, we offer the following simple escalation procedure:

If you experience dissatisfaction with Candle Customer Support at any time, please escalate your concern by calling the Candle support location closest to you. Ask to speak to a Customer Support manager. During standard business hours, a Customer Support manager will be available to talk with you or will return your call. If you elect to hold for a manager, you will be connected with someone as soon as possible. If you wish a return call, please tell the Candle representative coordinating your call when you will be available. After contacting you, the Customer Support manager will develop an action plan to
resolve your issue. All escalations or complaints received about support issues are logged and tracked to ensure responsiveness and closure.

**Above and Beyond**

What differentiates Candle’s support services from our competitors? We go the extra mile by offering the following as part of our Base Maintenance Plan:

- Unlimited multi-language defect, installation and operations support
- eSupport using the World Wide Web
- Regularly scheduled product updates and maintenance provided at no additional charge
- Over 200 specialized technicians providing expert support for your Candle products
Enhanced Support Services

Overview

Our Base Maintenance Plan provides a high level of software support in a packaged offering. However, in addition to this plan, we have additional fee-based support services to meet unique customer needs.

The following are some examples of our added-value support services:

- **Assigned Support Center Representative Services (ASCR)**
  - An assigned focal point for managing support escalation needs
  - Proactive notification of available software fixes
  - Proactive notification of product version updates
  - Weekly conference calls with your ASCR to review active problem records
  - Monthly performance reviews of Candle Customer Support service levels
  - Optional on-site visits (extra charges may apply)

- **Maintenance Assessment Service (MAS)**
  - On-site assessment services
  - Advice about product maintenance and implementation
  - Training your staff to develop efficient and focused procedures to reduce overall cost of ownership of your Candle software products
  - Analysis of your Candle product environment: versions, updates, code correction history, incident history and product configurations
  - Reviews to ensure that purchased Candle products and solutions are used effectively

- **Multi-Services Manager (MSM)**
  Multi-Services Manager provides highly valued services to customers requiring on-site full time expertise to complement their technical resources.
  - Dedicated on-site Candle resource (6 months or one year) at your site to help ensure maximum use and effectiveness of your Candle products
  - Liaison for all Candle product support activities, coordination and assistance with implementation of all product updates and maintenance releases
  - Works with your staff to understand business needs and systems requirements
- Possesses technical and systems management skills to enhance your staff’s knowledge and expertise
- Other projects as defined in Statement of Work for MSM services
Customer Support Contact Information

Link to Worldwide Support Telephone and E-mail information

To contact Customer Support, the current list of telephone numbers and e-mail addresses can be found on the Candle Web site, www.candle.com/support/.

Select Support Contacts from the list on the left of the page.
### Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td># (pound sign)</td>
<td>with major commands</td>
<td>38</td>
</tr>
<tr>
<td>-</td>
<td>as action character</td>
<td>275, 284, 297</td>
</tr>
<tr>
<td>( )</td>
<td>waiting</td>
<td>174</td>
</tr>
<tr>
<td>:</td>
<td>in column 1</td>
<td>33</td>
</tr>
<tr>
<td>* (as wildcard)</td>
<td></td>
<td>68</td>
</tr>
<tr>
<td>* (operator action required)</td>
<td></td>
<td>306</td>
</tr>
<tr>
<td>.ASF</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.AUP</td>
<td>immediate</td>
<td>42</td>
</tr>
<tr>
<td>.BEL</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.CLR</td>
<td>immediate</td>
<td>95</td>
</tr>
<tr>
<td>.CN</td>
<td>immediate</td>
<td>42</td>
</tr>
<tr>
<td>.DCL</td>
<td>immediate</td>
<td>43</td>
</tr>
<tr>
<td>.DD</td>
<td>immediate</td>
<td>43</td>
</tr>
<tr>
<td>.DEF</td>
<td>immediate</td>
<td>44</td>
</tr>
<tr>
<td>.DIR</td>
<td>immediate</td>
<td>45</td>
</tr>
<tr>
<td>.DSA</td>
<td>immediate</td>
<td>325</td>
</tr>
<tr>
<td>.DSE</td>
<td>immediate</td>
<td>46, 69</td>
</tr>
<tr>
<td>.EXM</td>
<td>immediate</td>
<td>47</td>
</tr>
<tr>
<td>.EXP</td>
<td>immediate</td>
<td>47</td>
</tr>
<tr>
<td>.FGO</td>
<td>immediate</td>
<td>48, 101, 336, 340</td>
</tr>
<tr>
<td>.I</td>
<td>immediate</td>
<td>51</td>
</tr>
<tr>
<td>.ILC</td>
<td>immediate</td>
<td>35, 51</td>
</tr>
<tr>
<td>.INT</td>
<td>immediate</td>
<td>101</td>
</tr>
<tr>
<td>.IOD</td>
<td>immediate</td>
<td>101</td>
</tr>
<tr>
<td>.LOG</td>
<td>immediate</td>
<td>52</td>
</tr>
<tr>
<td>.LOG OFF</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.LOG ON</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.LOGOFF</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.LOGON</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.LOGOUT</td>
<td>52, 89</td>
<td></td>
</tr>
<tr>
<td>.LOGPOP</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.LOGPUSH</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.MIN</td>
<td>immediate</td>
<td>53</td>
</tr>
<tr>
<td>.MJ</td>
<td>immediate</td>
<td>54</td>
</tr>
<tr>
<td>.MJC</td>
<td>immediate</td>
<td>54</td>
</tr>
<tr>
<td>.MJI</td>
<td>immediate</td>
<td>55</td>
</tr>
<tr>
<td>.MMA</td>
<td>immediate</td>
<td>55</td>
</tr>
<tr>
<td>.MOD</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td>.NEX</td>
<td>immediate</td>
<td>55</td>
</tr>
<tr>
<td>.PCS</td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>.PFK</td>
<td>immediate</td>
<td>56</td>
</tr>
<tr>
<td>.PLM</td>
<td>immediate</td>
<td>101</td>
</tr>
<tr>
<td>.PRM</td>
<td>immediate</td>
<td>58</td>
</tr>
<tr>
<td>.PRT</td>
<td>immediate</td>
<td>58</td>
</tr>
<tr>
<td>.R</td>
<td>immediate</td>
<td>59</td>
</tr>
<tr>
<td>.RC</td>
<td>immediate</td>
<td>59</td>
</tr>
<tr>
<td>.REP</td>
<td>immediate</td>
<td>61</td>
</tr>
<tr>
<td>.RMF</td>
<td>immediate</td>
<td>63, 274</td>
</tr>
<tr>
<td>.RMFS</td>
<td>immediate</td>
<td>63</td>
</tr>
<tr>
<td>.RTN</td>
<td>immediate</td>
<td>63, 340, 342</td>
</tr>
<tr>
<td>.SCC</td>
<td>immediate</td>
<td>69, 95, 119</td>
</tr>
<tr>
<td>.SET</td>
<td>immediate</td>
<td>101</td>
</tr>
<tr>
<td>.SGO</td>
<td>immediate</td>
<td>65, 101, 336, 340</td>
</tr>
<tr>
<td>.SLD</td>
<td>immediate</td>
<td>101</td>
</tr>
<tr>
<td>.SMT</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.SPT</td>
<td>immediate</td>
<td>67, 169</td>
</tr>
<tr>
<td>.TSF</td>
<td>immediate</td>
<td>87, 333</td>
</tr>
<tr>
<td>.VAR</td>
<td>immediate</td>
<td>71</td>
</tr>
<tr>
<td>.VAR POP</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.VAR PUSH</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>.VAR</td>
<td>immediate</td>
<td>73</td>
</tr>
<tr>
<td>.VTM</td>
<td>immediate</td>
<td>74</td>
</tr>
<tr>
<td>.WAT</td>
<td>immediate</td>
<td>75</td>
</tr>
<tr>
<td>.XLF</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>.XLFOUT</td>
<td>immediate</td>
<td>89</td>
</tr>
<tr>
<td>.ZER</td>
<td>immediate</td>
<td>87</td>
</tr>
<tr>
<td>/A</td>
<td>INFO-line</td>
<td>41</td>
</tr>
<tr>
<td>/ABORT</td>
<td>INFO-line</td>
<td>40</td>
</tr>
<tr>
<td>/ASF</td>
<td>INFO-line</td>
<td>87</td>
</tr>
<tr>
<td>/ATTACH</td>
<td>INFO-line</td>
<td>41</td>
</tr>
<tr>
<td>/ATTN</td>
<td>INFO-line</td>
<td></td>
</tr>
<tr>
<td>/AUP</td>
<td>OFF INFO-line</td>
<td>41</td>
</tr>
<tr>
<td>/AUPON</td>
<td>INFO-line</td>
<td>41</td>
</tr>
<tr>
<td>/BELL</td>
<td>INFO-line</td>
<td>87</td>
</tr>
<tr>
<td>/D</td>
<td>INFO-line</td>
<td>46</td>
</tr>
<tr>
<td>/DEF</td>
<td>INFO-line</td>
<td>44</td>
</tr>
<tr>
<td>/DOWN</td>
<td>INFO-line</td>
<td>46</td>
</tr>
<tr>
<td>/G</td>
<td>INFO-line</td>
<td>50</td>
</tr>
<tr>
<td>/GIVE</td>
<td>INFO-line</td>
<td>50</td>
</tr>
<tr>
<td>/H</td>
<td>INFO-line</td>
<td>50</td>
</tr>
<tr>
<td>/HELP</td>
<td>INFO-line</td>
<td>50</td>
</tr>
</tbody>
</table>
/HOLD INFO-line 51
/INT INFO-line 101
/LOG INFO-line 52, 56
/LOG OFF INFO-line 87
/LOG ON INFO-line 87
/LOGOUT INFO-line 89, 333
/MODIFY command 153
/O INFO-line 52, 56
/OPTN immediate 332
/P INFO-line 57
/PRINT INFO-line 57
/PWD INFO-line 58
/RENM immediate 60
/arguments 60
/REP INFO-line 60
/RESHOW INFO-line 63
/S INFO-line 64
/SAVE INFO-line 61, 64
/STK INFO-line
screen stacking feature 68
/STOP INFO-line 70
/T INFO-line 70
/TAKE INFO-line 70
/TOP INFO-line 70
/TSF immediate 87
/U INFO-line 70
/UP INFO-line 70
/WAIT INFO-line 75
/XACB immediate 332
/XLF INFO-line 75, 87
/XLFOUT INFO-line 89, 332
/XRFS immediate 266
/XRFT immediate 267
/XRSV immediate 267
/ZER INFO-line 87
/ZOOM INFO-line
   cursor pointing feature 79
   zooming feature 77
; (semicolon) for command help 38
< as action character 287, 290, 294, 295, 297
<(less-than) with major commands 38
== == == immediate 40
> (greater-than) for comment lines 38
> new input 174
? (question mark) for command help 38
? in column 1 33
| CPU dispatchable 174
/(slash) for command help 38

Numerics
2305 major 171
24-bit address 310
31-bit address 310
3330 major 171
3330-11 disks 171
3340 major 171
3350 major 171
3375 major 171
3380 major 171
3390 major 172
3330-11 disks 171
3340 major 171
3350 major 171
3375 major 171
3380 major 171
3390 major 172

A
A (in label field of TCBS) 319
ABWP minor 243
ACB minor 155
ACBLIB
dataset 155
ACCS minor 222
action 275
caracter 282, 287, 295, 297, 301
action character 307, 328
action messages 306
active CPUs, display number of 152
ACTN minor 306
LINE minor 275
address
   pre-defined 278
   specifying 278
address names 279
   specifying (MDEF) 279
address space
collecting data about 307
ID 208
   peeking at OMEGAMON 307
plotting status and activity 173
ADMB
   minor 222
Adobe portable document format 14
AENV minor 207
AINF minor 262
ALL 109, 118
allocated pool space 236, 237, 243, 249, 250
allocating datasets 323
allocations to disk 172
ALOC minor 235, 236, 237, 243, 249, 250
AMAP minor 276, 308
anchor 278
AOIP 233
APF authorization 319
APIE exception 253
application
   program scheduling 205
APSB
   minor 229
area information 262
argument
  command 32
difference 39
  hour 39
  minute 39
  operand 32
  rate 39
  second 39
ASCR
  assigned support center representative 352
ASF OPTN keyword 87
ASID
  minor 208
ASID number
  in PEEK command 307
assigned support center representative
ASCR 352
ATTRIBUTE 119
authorization
  for data-only spaces listing 325
authorized commands 274
  requiring action character 275
  supplying the password 274
  using action character ( - ) 275
AUTO keyword 334
automated features
  updating 41, 42
automatic features 331, 342
automatic screen facility
  screen space 340
automatic screen facility (ASF) 29, 331, 334, 339
  .DEF immediate 44
  /DEF INFO-line 44
activating 82, 87
cancel while running 340
commands 332
  example 341
  screen space 340, 342
terminating 63
  using .FGO 48
  using .SGO 65
automatic update 26, 83, 102
AVQ minor 178
AWAT minor 243

BELL 119
bell
  activating 82, 87
  interval 82, 87
  setting terminal 45
  status on INFO-line 29
BFSP minor 132
BFUS minor 132
BLKL major 139
BLKN minor 139
BLKU minor 139
BMP
  major 194
  bottleneck analysis component (DEXAN) 24
box 120
BOXATTR 120
BOXCHAR 120
BSIZ minor 240
BSPH minor 133, 134, 135, 136, 137
BSTS minor 247
BSZE minor 244
buffer
  subpool statistics 239
  buffer pool
    allocation 247
    statistics 217, 247
    usage sizes 247
  buffer subpool information 242
BUFS 234, 235

C
CA splits 222
CALL minor 198
Candle subsystem 343
  operation 344
  starting 345
  stopping 345
CANSOI 65
CASP minor 222
CBTA major 138
CBTL major 138
CBTP major 138
CDMB minor 198, 199
CERR minor 240
CESS major 234
chaining screens 340
character 275
checkpoint ID 141
  table 132
checkpoint/restart log record buffer 136
CI
  free 262

Index 357
bottleneck analysis component (DEXAN) 24
historical component (EPILOG) 24
realtime performance component
(OMEGAMON) 23
CON minor 178
CONF immediate 123
CONS major 275, 304
console
  issue commands 303
continuation character 33
control
  area splits (VSAM only) 222
interval splits (VSAM only) 222
control region
  private area 286
storage, modifying 294
CONU 305
COPY minor 90
CPCB minor 199
CPU
dispatchable 174
  minor 208
number of active 152
percent used 214
time figures about initiator 212
usage 151, 152
cross memory
collector segment 28
commands 314
director code on INFO-line (DIR) 28
director code on INFO-line (VTD) 28
executing commands 45
cross system
collector 50
collector code on INFO-line (DSK) 28
director code on INFO-line (DIR) 28
director code on INFO-line (VTD) 28
executing commands 45
CSA 297
  immediate 129
real memory 128
storage isolation 131
usage 129
CSAR immediate 128
CSTI immediate 131
CTL major 195
CTRM minor 199
CTRN minor 199
CTSK minor 199
CUMULATIVE 121
CUOW
  minor 263
CURR minor 139

number of 262
request 263
size 263
splits 222
total 263
CIAB minor 262
CIDO minor 262
CIIF minor 262
CIIO minor 262
CIRA minor 263
CISO minor 263
CISP minor 222
CISZ minor 263
CIUW minor 263
CKPT major 141
clear screen 41
CNDPROC START parameter 345
code on INFO-line 28
collector 45
  synchronize 75
collector segment
  DSK on INFO-line 28
  XMM on INFO-line 28
color
  .SCC command 95
    basic 27, 97
commands 95
defining 100
extended 27
options 83
profile definition mode 96
command 300
  argument 32
  entering parameters 32
  help 33, 34, 35, 38
  IMS, issuing 300
  interface 23
  name 31
  output, setting case 82, 87
command format 31
command type
  immediate 30, 31
INFO-line 30
major 30
minor 30
commands
  issuing console 303
comment lines 38
common storage area 297
Common User Access model 23
communications external subsystem pool
  commands 234
components of OMEGAMON II 22, 23
<table>
<thead>
<tr>
<th>Term</th>
<th>Page(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>current user of device</td>
<td>175</td>
</tr>
<tr>
<td>cursor pointing feature</td>
<td>/ZOOM INFO-line 77</td>
</tr>
<tr>
<td>CUSF minor</td>
<td>263</td>
</tr>
<tr>
<td>customer support</td>
<td></td>
</tr>
<tr>
<td>base maintenance plan</td>
<td>348</td>
</tr>
<tr>
<td>contact information</td>
<td>354</td>
</tr>
<tr>
<td>enhanced support services</td>
<td>352</td>
</tr>
<tr>
<td>eSupport</td>
<td>349</td>
</tr>
<tr>
<td>severity levels</td>
<td>349</td>
</tr>
<tr>
<td>telephone support</td>
<td>348</td>
</tr>
<tr>
<td>customizing</td>
<td></td>
</tr>
<tr>
<td>exceptions</td>
<td>106</td>
</tr>
<tr>
<td>cycle delay</td>
<td></td>
</tr>
<tr>
<td>.RTN immediate</td>
<td>64</td>
</tr>
<tr>
<td>cycles</td>
<td>173</td>
</tr>
<tr>
<td>D immediate</td>
<td>43</td>
</tr>
<tr>
<td>DADR minor</td>
<td>172</td>
</tr>
<tr>
<td>DALT minor</td>
<td>172</td>
</tr>
<tr>
<td>DASD</td>
<td>254</td>
</tr>
<tr>
<td>DASD logging exceptions</td>
<td>109</td>
</tr>
<tr>
<td>DATA minor</td>
<td>276</td>
</tr>
<tr>
<td>data management blocks (DMBs)</td>
<td>217</td>
</tr>
<tr>
<td>DATA minor</td>
<td>311</td>
</tr>
<tr>
<td>data space</td>
<td></td>
</tr>
<tr>
<td>display memory</td>
<td>326</td>
</tr>
<tr>
<td>modifying storage</td>
<td>329</td>
</tr>
<tr>
<td>owner</td>
<td>325</td>
</tr>
<tr>
<td>scan control blocks</td>
<td>328</td>
</tr>
<tr>
<td>scan storage</td>
<td>327</td>
</tr>
<tr>
<td>type</td>
<td>326</td>
</tr>
<tr>
<td>data space/HiPerspace</td>
<td></td>
</tr>
<tr>
<td>utilization (DATA minor)</td>
<td>311</td>
</tr>
<tr>
<td>data spaces</td>
<td>325</td>
</tr>
<tr>
<td>database</td>
<td>218, 237</td>
</tr>
<tr>
<td>buffer pool statistics (DBVS)</td>
<td>239</td>
</tr>
<tr>
<td>information, non-recoverable</td>
<td>221, 225</td>
</tr>
<tr>
<td>pool statistics</td>
<td>217</td>
</tr>
<tr>
<td>split and extent information</td>
<td>227</td>
</tr>
<tr>
<td>database exceptions</td>
<td>109</td>
</tr>
<tr>
<td>data-only spaces</td>
<td>325</td>
</tr>
<tr>
<td>authorization (.DSA)</td>
<td>325</td>
</tr>
<tr>
<td>dataset</td>
<td></td>
</tr>
<tr>
<td>allocation</td>
<td>323</td>
</tr>
<tr>
<td>information</td>
<td>155</td>
</tr>
<tr>
<td>OLDS</td>
<td>254, 256</td>
</tr>
<tr>
<td>primary</td>
<td>254, 255</td>
</tr>
<tr>
<td>secondary</td>
<td>254, 255</td>
</tr>
<tr>
<td>sequential for logging</td>
<td>89</td>
</tr>
<tr>
<td>WADS</td>
<td>255, 258</td>
</tr>
<tr>
<td>dataset allocation</td>
<td></td>
</tr>
<tr>
<td>displaying</td>
<td>312</td>
</tr>
<tr>
<td>date display</td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>82, 87</td>
</tr>
<tr>
<td>on INFO-line</td>
<td>29</td>
</tr>
<tr>
<td>DAY keyword</td>
<td>337</td>
</tr>
<tr>
<td>DB#</td>
<td>109</td>
</tr>
<tr>
<td>DB2 attach information</td>
<td>269</td>
</tr>
<tr>
<td>DBCTL control region</td>
<td>194</td>
</tr>
<tr>
<td>dependent regions</td>
<td>194</td>
</tr>
<tr>
<td>region minor commands</td>
<td>198</td>
</tr>
<tr>
<td>regions</td>
<td>189, 194</td>
</tr>
<tr>
<td>DBIO</td>
<td>218</td>
</tr>
<tr>
<td>DBOS major</td>
<td>242</td>
</tr>
<tr>
<td>DBRC</td>
<td>286, 295</td>
</tr>
<tr>
<td>major</td>
<td>195</td>
</tr>
<tr>
<td>RECON dataset information</td>
<td>157</td>
</tr>
<tr>
<td>region</td>
<td>281</td>
</tr>
<tr>
<td>DBT minor</td>
<td>199</td>
</tr>
<tr>
<td>DBVS immediate</td>
<td>239</td>
</tr>
<tr>
<td>DBWP major</td>
<td>236</td>
</tr>
<tr>
<td>DC monitor log control directory</td>
<td>135</td>
</tr>
<tr>
<td>DCAT minor, issued to disk</td>
<td>172</td>
</tr>
<tr>
<td>DCB</td>
<td></td>
</tr>
<tr>
<td>attributes</td>
<td>89</td>
</tr>
<tr>
<td>open on disk</td>
<td>172</td>
</tr>
<tr>
<td>DD statement for logging</td>
<td>89</td>
</tr>
<tr>
<td>DDIR minor</td>
<td>133</td>
</tr>
<tr>
<td>DDNM Minor</td>
<td></td>
</tr>
<tr>
<td>dataset</td>
<td>156</td>
</tr>
<tr>
<td>DDNM minor</td>
<td>91, 156</td>
</tr>
<tr>
<td>DDNS minor</td>
<td>276, 312</td>
</tr>
<tr>
<td>extended information</td>
<td>313</td>
</tr>
<tr>
<td>debugging</td>
<td>297</td>
</tr>
<tr>
<td>DED</td>
<td></td>
</tr>
<tr>
<td>code on INFO-line</td>
<td>28</td>
</tr>
<tr>
<td>DEDA major</td>
<td>260</td>
</tr>
<tr>
<td>DEDB</td>
<td></td>
</tr>
<tr>
<td>master control</td>
<td>133</td>
</tr>
<tr>
<td>resources accessed and/or request by area</td>
<td>264</td>
</tr>
<tr>
<td>DEDB area</td>
<td>260</td>
</tr>
<tr>
<td>by list</td>
<td>261</td>
</tr>
<tr>
<td>control</td>
<td>133</td>
</tr>
<tr>
<td>DEDH major</td>
<td>261</td>
</tr>
<tr>
<td>dedicated mode</td>
<td>27</td>
</tr>
<tr>
<td>DED on INFO-line</td>
<td>28</td>
</tr>
<tr>
<td>DEDL major</td>
<td>261</td>
</tr>
<tr>
<td>DEDU major</td>
<td>261</td>
</tr>
<tr>
<td>DEFER</td>
<td>41</td>
</tr>
<tr>
<td>definition mode</td>
<td></td>
</tr>
</tbody>
</table>
delete 120
both main storage and RKOIPCSV (default) 45
main storage only 45
RKOIPCSV only 45
screen space 45
delimiter(’) 120
delimiter \( \) 120
developer 45
dictator minor 91
device 168, 173
device
3270-type 95
activity (IMS system control address spaces) 177
address 312
color 95, 96
DCAT minor of disk commands 172
DPIN minor of disk commands 172
EDS or non-EDS 96
EXCP count 172
four-base 96
generic 166, 170
monochrome, EDS or non-EDS 96
name table entries 83, 101
reserve 174
developer major 168
developer major 169
DF996I message suppressed 302
dGN minor 199
dGU minor 199
difference argument 39
dING immediate 45
dIO minor 172
dIOQ minor 172
dIR
code on INFO-line 28
director 45
director segment
DIR on INFO-line 28
VTD on INFO-line 28
disk
activity, plotting 173
allocations, current 172
busy 169
control unit, selecting by 170
I/O queue length 170, 172
information 168
mass storage virtual 169
permanently resident 169
reserved from CPU 170
suspended channel programs 169
types 174
unit address 172
waiting for mounts 169
disk dataset
DMHR minor 133
DMPL major 237
documentation set information 14
DOM# minor 209
domain number 209
DOPN minor 172
DPCB minor 200
DPDKS minor 177
DPIN minor 172
DPRT minor 209
DRCS minor 200
DRDY exception 341
DRES minor 174, 223
DSC minor 178
DSETG99 immediate 186
DSIZ minor 250, 251
DSKB major 169
DSKC major 169
DSKE major 169
DSKG major 169
DSKM major 169
DSKN major 169
DSKP major 170
DSKQ major 170, 173
DSKR major 170
DSKS major 170
DSKU major 170
DSKV major 170
DSME 261
DSML 261
DSORG 313
DSPS 261
DSPX 262
DSTA minor 174
DSTU minor 91
DSZE minor 223
DTPY minor 174
dump IMS control blocks 132
DUMP major 132
DUSR minor 175
DUT minor 179
DVMP minor 175
DVOL minor 175
DYNA immediate 323
dynamic
   I/O device activity 343
      storage area control block tables 138
DYNU immediate 324
ECRC minor 271
ECSA immediate 129
EDS 96
EIO minor 244
ELSQA,SWA
   available storage 310
      current bottom 310
      unallocated 310
EPCB major 249
EPST minor 134
ESCD minor 134
ESQA immediate 129
eSupport
   customer support 349
ETIC minor 200
ETIO minor 200
ETLK minor 200
ETPL minor 200
ETSP minor 200
exception
   batch message processing 194
      defining characteristics 108, 112
         group 109
      summarizing activity 108
      summary of activity 110
      thresholds 117
exception analysis 332
   exception groups 112
   immediate commands 109
EXCP 313
LEXC immediate 114
starting 108, 109
exception group
   controlling 122
   defining 112
exception logging
   .LOG immediate 52
   /LOG INFO-line 52
   /XLF INFO-line 75
      print options 89
exception logging facility (XLF) 26, 331, 334, 339
   activating 82, 88
   commands 332
exception output
   reorder execution of exceptions 114
exceptions
   customizing 106
      grouping 112
EXCP 313
   issued to disk 172
execute minor commands 47
EXNCYC 121
expiration date 47
EXTD minor 235

extended
  attributes  96
  data stream  96
  highlighting  98
PST  134
SCD  134

FA  109
Fast GO command  48
Fast Path  217, 247, 260
  buffer headers  133
  data entry database areas  260
  exceptions  109
  latch conflicts  125
  output threads  265
  resource request conflicts  124
  summary information  260
  utility active on DEDB area  263
FCDE immediate  142
FCPB immediate  143
FGOLIMIT .SET keyword  101
FGOLOOP .SET keyword  101
field
  argument  32
  command name  31
  label  31
  parameter  32
fixed frames  209
FIXF minor  209
FMCT minor  209
FOLD minor  92
FORCE  118
FORM minor  92
FPDB major  247
FPDSK minor  177
FPUT minor  263
FR  109
frames  209
free
  Cls  262, 263
    pool space  236, 237, 243, 250, 251
free block  310, 311
FREE minor  235, 236, 237, 243, 250, 251
free space
  fragmented  310, 311
free-allocated queue elements (FAQEs)  236, 237,
  243, 249, 250, 251
FRES minor  139
FSYS immediate  260
FWRT minor  240
FXFR minor  209

G
GDEV major  170
GDEVUCBS .SET option  101
GDFN immediate  112, 119
generic
  devices  166, 170
  selection  67
GETS minor  140, 240
GHN minor  200
GHNP minor  200
GHI minor  201
GLBL immediate  188
GLST immediate  166
GNP minor  201
GROUP  119

H
H.MJ  54
H.MJC  55
H.MJL  55
help  33
  added information  33
  extended  33
  immediate commands  33
  INFO-line commands  35, 51
  major commands  33
  minor commands  33
  highlighting  95, 119, 120
Hiperspace
  hit radio  242
hiperspace
  display memory  326
  modifying storage  329
  owner  325
  scan control blocks  328
  scan storage  327
  type  326
hiperspaces  325
historical component (EPILOG)  23, 24
hit radio  242
HOLD minor  92
hour argument  39
HSPB minor  201
HSSP
  DEDH major  261
  options  205
HUIC minor  210

I
I/O
  queue length  172, 176
IBM releases  25
IBPL minor 134
IBPR minor 134
ICHP minor 175
ICMD immediate 300
ICNS 302
   immediate 301
ID# minor 223
ID1 minor 93
ID2 minor 93
ID3 minor 93
ID4 minor 94
IDDN major 155
IDEV major 177
IEALIMIT 149, 315
IHLD minor 201
IKT00405I screen error 98
ILST 282
ILST immediate 281
IM 109
   immediate commands 31
   DBCTL regions 190
   help 33
   listing 54, 55
IMS
   1.3, IMS DASD logging 254
   attach information 269
   commands, entering 300
   control blocks 132
   CPU utilization 151
   DASD logging analysis 254
   displaying 301
   internal exceptions 109
   pool statistics 217, 233
   READY message suppressed 302
   resources 217
   virtual storage exceptions 109
IMS ID
   on INFO-line 29
IMS READY message suppressed 302
INFO-line
   automatic screen facility indicator 29
   bell status 29
   commands 51
   date display 29
   format 28
   help 51
   logging status 28
   screen stacking feature 69
   scroll amount 29
   system ID 29
   time of last refresh 29
INFO-line commands 30
   help 35
   input area 28
   initialization cycle 39
   initialization parameter block 134
   inserting blank lines 51
   installation 320
   installation profile, IOPT command 83
   installed channel path for the disk 175
   in-storage screen facility 45, 60, 61, 64, 65
   intensity, changing with .SCC 97
   interval
      setting 83, 101
      value 173
   INTERVAL .SET keyword 101
   INUM minor 201
   IODELAY .SET keyword 101, 102
   IODP minor 210
   IOJ minor 210
   IOPT immediate 103
   IOS minor 179
   IPB minor 134
   IPDSK minor 177
   IPFR minor 140
   IPGA major 138
   IPGL major 138
   IPGP major 138
   IPLN minor 140
   IPRF immediate 86
   IR# minor 223
   IRES minor 230
   IRLM 146, 286, 295
      CONF immediate 123
      lock conflicts 123
      locks 202
      major 195
      region 281
      startup parameters 145
ISAM/OSAM 217
   buffer pool prefix 134
   buffer subpool header 135
   buffer subpool prefix 134
ISBP minor 135
ISCN immediate 285
ISES minor 240
ISKS minor 240
ISPFD mode 26
   under VTAM (VTS) 28
ISRT minor 201
ISYS immediate 150
IT# minor 223
IV 109
IWRT minor 240
IZAP immediate 294, 297, 298
IZIB minor 135

J
JES2
operator command 303
job status 210
jobpack
queue 142
jobpack area 314
JOBS minor 276, 313
example 314
JPCI minor 210
JPUI minor 210
JPUO minor 210
JSTA minor 210

K
KCNDLI subsystem interface module 344
KOBVTAM
monitor 74
KOIGBL module 188

L
label field 31
LANG minor 230
largest free block (fragmentation exceptions) 109
LBFR minor 244
LBUF minor 254
len argument 282, 284
LENV minor 254
LEVL immediate 95
LEXC immediate
exception analysis 114
LFND minor 244
LIMIT keyword 334
LINE minor 306
LIST 118
list minor commands 53
list screen space
both main storage and disk datasets (default) 65
CANSOI 65
default 65
disk datasets 65
main storage only 65
RKOIPCSV 65
LNCT minor 94
load screen space
from CANSOI to main storage 53
locate IMS modules in virtual memory 142
lock
conflicts 123
trace table entries 160
LOCK minor 202
LOCOP minor 140
LOG
keyword 334
on INFO-line 28
log
activating 82, 87
print 52, 56
turn off 52
turn on 52
LOGGED message 57
logging 331, 342
analysis 254
DASD 254
IMS messages 321, 323
LOGGING Main Menu option 332
long-term utilization displays 177
LOOPCOUNT .SET keyword 101, 102
LOOPCOUNT keyword
control block testing 319
using 308
looping
detection 308
setting parameters 83, 102
LOOPTIME .SET keyword 101, 102
LPDSK minor 177
LRECL 313
LROWS
screen stacking feature 68
LSCR immediate 53
LSETG immediate 185, 186
LSO=S option 230
LSQA,SWA
unallocated 310
LSQA/SWA
bottom 311
storage space 311
LSTA minor 254
LSYS major 254
main storage 45, 60, 61, 64, 65
maintenance assessment service
MAS 352
major commands 30, 228
controlling output 38
DBCTL regions, selecting 194
displaying complete list 38
DMBs, selecting 220
help 33
listing 54
PSBs, selecting 228

364 OMEGAMON II for DBCTL Realtime Commands Reference Manual Version 510
management block (DMB)
  commands 218
  pool 237
MAS
  maintenance assessment service 352
master console 304
Master Terminal
  Input 300
  Output 301
MAXG immediate 187
MAXS minor 140
MCHN immediate 275, 288
MCS messages, logging 320
MDBL minor 156
MDEF immediate 275, 279
MDST minor 157
memory
  display from data-only spaces 326
  scanning data-only spaces 327
menu interface 23
message 301
  boxing 120
  forcing 119
  logging facility 320
  No Act 275
  processing regions 194
  queued to WTOR LTERM 302
  setting color in 119
messages
  in-line 306
  mount 306
  out-of-line 306
  reply 306
minor commands 30, 222
  case, setting 82, 87
  DMBs, selecting 222
  execute all 47
  executing 47
  help 33
  listing 53
  PSBSs, selecting 229
minute argument 39
MLCD minor 135
MLOG
  CLASS suboperand 321
  immediate 275, 321, 323
  START operand 321
  STATUS operand 322
  STOP operand 322
  MLST immediate 275, 283
  MNT minor 306
  CONU major 275
  MODBLK dataset 156
  modes of operation 26
  shown on INFO-line 28
MODS minor 276, 314
MODSTAT dataset 157
MODSTAT2 dataset 153
mount
  messages 306
  status 174
MSCN immediate 275, 290
MSGD immediate 115, 116
MSM
  multi-services manager 352
MSYS immediate 151
MTO 300, 301
  command 300
  console support 300
  messages, logging 320
  multi-line input facility 85
  multi-processors and CPU utilization 214
  multi-services manager
    MSM 352
MVS
  region minor commands 207
  resource exceptions 109
  versions SP1.3 169
MVS command
  issuing 303
MVS console authority 304
MZAP immediate 275
name
  DEDB 264
  pre-defined 280
NAME minor 264
NBFR minor 241
NBUF minor 244
NDSP 119
non-recoverable database 221, 225
non- VSAM database 224
NPGS minor 140
number
  CI request by area 263
numeric argument
  controlling major command output 38
NVSC minor 211
O
  OBLB minor 256
  OBLK minor 256
  OBLW minor 256
  OBLZ minor 256
OBST minor 257
OCHG immediate 153
OCHP minor 175
OCIO minor 257
OCMD immediate 275, 303
ODDN minor 224
ODDS major 256
OFF 119
OIREPORT file 339
OIXFLOG file 339
OKN minor 207
OLDS
dataset 254, 256
minor 254
OLDSnn minor 254
OLRL minor 257
OMEGAMON
address space storage display 283
OMEGAMON address space (MLST) 283
OMEGAMON program check statistics 56
ON 119
online
change analysis 153
channel paths for the disk 175
disk 169
OPDN minor 257
operation
mode of 28
operational parameters
.SET immediate 101
setting 83
operator 300
console 303
operator console
display commands 304
OPST minor 257
OPTN immediate 87, 333
ASF keyword 332, 341
keyword 336
TSF keyword 333
XLF keyword 332
OPUA minor 257
OPVL minor 257
ORDR minor 257
OS 109
OSDN
minor 257
OSST minor 257
OSUA minor 258
OSUB major 243
OSVL minor 258
OTHR immediate 265
out-of-line messages 306
OUTP
minors, parentheses 90
REPORT 339
OUTP major 75, 89, 339
output 301
output buffer
location 304
owner
of data-only spaces (OSPC) 325
P
PA1 key
/ATTN INFO-line 41
PAB minor 135
pageable link pack area (PLPA) page-fix 295
PAGELIMIT .SET keyword 101, 102
page-stolen 215
paging fence limits 131
parameter anchor block 135
parameters
displaying startup command 58
logging 339
setting 334, 336
settings 334
PARM minor 271
partition
PSB pool 252
specification table 136
password 58
protection 313
pattern
clearing 68
selection 67
setting 67
PB# minor 224
PD# minor 225
PDIR minor 135
PDLS minor 230
PDRA minor 230
PDSK minor 177, 180, 181
PEEK major 275, 307
action characters with 307
text 307
example 307
LOOPCOUNT keyword 308
PEEKSIZE keyword 308
work area size 308
PEEKSIZE
.SET option 101, 102
PEEKSIZE keyword
using 308
PERF minor 211
PERSIST keyword 335
PF keys
assigning to 56
cancel definition 57
delete definition 57
screen save 56

PI
  lock 123, 202
PIEP major 252
PLAN minor 203
plot device activity 173
PLOT immediate 161
PLPA directory 142
PLSY minor 203
PMSG immediate 117
PMSGnn immediate 117
PND minor 179
PNR major 195
POLDS minor 254
pool
  requests 251
  space 236, 238, 243, 249, 250, 251
  statistics 233
  storage 250, 251
pool utilization exceptions 109
PPRF immediate 86
PPUS minor 231
pre-defined names 280
PRES minor 231
primary dataset 254, 255
printing
  .PRT immediate 58
  .REP 61
  /O 56
  /p 57
  /PRINT INFO-line 57
  output options 83, 102
printing problems 14
private area 307
private area 314
  bottom 310, 311
  top 310
  top of extended 310
PROC minor 212
profile 29, 96
PROFILE Main Menu option 332
ProfileDefinitionMode keyword 96
program
  name 315
  structure 318
program check statistics
  OMEGAMON 56
program isolation
enqueue pool 252
PROGxxx program check 98
protect key 129
PSB 217, 228
  directory entry 135
  groups 183, 187
  name 203
PSBA major 228
PSBC minor 225, 231
PSBI major 228
PSBL major 228
PSBN minor 203
PSBP major 228
PSBS major 228
PSBU major 229
PSBW major 249
PSET major 163
PSPL major 250, 252
PST
  address 203
  minor 136
PSTA minor 203
PSTI minor 212
PSW (Program Status Word) 277
PSZE minor 231, 244
PT# minor 225
PTYP minor 231
PURG minor 244
PWADS minor 255
PWRT minor 244

Q
question mark
  in column 1 33

R
rate argument 39
ratio of real memory to virtual 152
RCMD major 304
RCNS minor 157
RCOV minor 225
RDB2 major 195
RDS minor 158
real memory 152
realtime performance component
  (OMEGAMON) 23
RECFM 313
refresh current screen
  /RESHOW INFO -line 63
region
  ID number 204
  IEALIMIT value 315
  immediate commands 190
  in use 315
limit 315
private area 286
requested 315
selection commands 194
SRM information 211
status 205
steps 212, 213
storage, modifying 295
type 204
relational operators (.FGO) 49
relational operators (.SGO) 67
relational operators (.VAR) 73
releases supported 25
remote terminals 26
rename screen space 60
reorder exception analysis chain linkage dynamically 114
REPEAT keyword 334
REPL minor 203
replace screen
  in both main storage and RKOIPCSV 61
  in main storage only 61
  in memory 61
  in RKOIPCSV 61
  in-storage screen facility 61
REPORT file 89
  defining size 83
  LOG on INFO-line 29
REPORT log 342
REQU minor 250, 251
reset
  colors 41
  passwords 41
resource
  commands 217, 260
resource measurement facility 172, 173, 274
restart dataset 158
RESTART START Parameter 345
return to calling screen 63
reverse video 119, 120
RGID minor 204
RGN2 major 197
RGNA major 195
RGNB major 196
RGNC major 196
RGND major 196
RGNH major 196
RGNI major 196
RGNL major 196
RGNN major 196
RGNP major 197
RGNN major 197
RGOR major 197
RGNS major 197
RGNT major 197
RIO minor 244
RKEY minor 241
RKOIPCSV 45, 53, 60, 64, 65
  DELT immediate 45
  immediate 45
RLMO immediate 145
RLMX 146
RMF 172, 173, 274
RRBA minor 241
RSIN minor 136
RSP minor 179
RTYP minor 204
S
S on INFO-line 69
SACT minor 204
SALL minor 204
SAP minor 136
SAPP major 253
save area
  prefix set (SAP) pool 253
  set prefix 136
save screen
  to both main storage and RKOIPCSV 64
  to main storage only 64
  to memory 64
  to RKOIPCSV only (default) 64
saving 279
SBPL minor 140
SBUF immediate 246
SC keyword 335
SCBF minor 241
SCD minor 136
SCHD minor 226, 232
scheduler trace
  table entries 160
scheduling, application program 205
SCHN immediate 276, 328
screen
  display options 82, 87
  erasure error 98
  print partial 58
  printing of 57
Screen GO command 65
screen space 300
/S INFO-line 64
/SAVE INFO-line 64
conditional fetch 66
creating 65
deleting 45
DELT immediate 45
description 28
display libraries 65
fetch (.FGO) 48, 83, 101
fetch (.SGO) 65
first 82, 87
input area 28
listing 65
loading 53
LSCR immediate 53
MTO 300
renaming 60
RENM immediate 60
returning to 63
saving 64
saving address names in a 279
SCRN immediate 65
stacking shown on INFO-line 29
test 49
screen spaces
deleting 45
screen stacking display
.DSE immediate 46
screen stacking feature
/STK INFO --line 68
INFO-line 69
SCRN immediate 65
arguments 65
scrolling
/U 70
/UP 70
amount 29
setting 82, 87
SDSK minor 177
SEAR minor 244
second argument 39
secondary dataset 254, 255
secondary terminals 42
security
external 274
features 58, 59
internal 274
selection
generic 67
pattern 67
semicolon (33, 34
separator line 40
SEQN minor 212
sequence on dispatching queue 212
service units 211
sessions
codes shown on INFO-line 28
options command 82, 87
set and display PF keys 56
SETG immediate 184
SETO minor 205
SETR minor 205
severity levels
customer support 349
SFND minor 241
single quote (') delimiter 120
SIO rate 152
SIZE 234
SIZE minor 238, 248, 249, 250, 251
size of CI 263
SL keyword 335, 337
SLCD minor 136
slots 209
SLST immediate 276, 326
SMF ID 152
SOLDS minor 254
SOUT minor 94
SPAL immediate 242
SPD# minor 226
SPID minor 137
split-screen 26
SQA
immediate 130
modifying 297
SRB (Service Request Block)
program check 277
scheduling 276
time-outs 276
SRBT minor 213
SRM information, regions 211
SS keyword 335, 337
SSCN immediate 276, 327
ST 109
start repeating commands 59
startup
JCL 143
parameter overrides 143
STAT minor 205, 226, 232, 272
STATE 119
static exceptions 109
status
information on INFO-line 28
STEP minor 213, 276, 314
STOP immediate 69, 70
stop processing for a collector immediately 40
STOP= 121
storage 282
addresses, naming (MDEF) 279
common area (MLST) 283
displaying 281
for ELSQA/SWA/USER 310
isolation parameters 131, 212
listing from data-only spaces 326
modifying 294
modifying data-only space 329
private area (MLST) 283
protect key 128, 319
scan 285
scanning data-only space control blocks 328
scanning data-only spaces 327
subpool allocations 316
storage area
 prefixed 311
storage display
 common area storage display 283
storage management information
 PEEM major 307
storage, modifying 294
STYP minor 232
SUAL minor 213
SUBA major 270
SUBL major 270
SUBP minor 276, 316
SUBP/n major 270
subpool
 ID 226
 number 129
subpool allocations 316
subsystem interface module 344
SUCP minor 213
SUIO minor 213
SUMM 158
SUMM Dataset information 158
SUMS minor 214
support 119
SVC vector table directory 137
SVTD minor 137
SWADS minor 255
SWRT minor 245
SYS/USER
 storage space 311
SYS1.PARMLIB
 SMFPRMnn parameter 152
system
 CPU usage 151
 log control directory 136
system ID 152
Systems Application Architecture 23
SZAP immediate 276, 329
dispatchability 319
scanning with PEEK command 308
structure 317
TCBS minor 276, 317
TCPU minor 214
telephone support
customer support 348
terminal bell 82, 87
setting 45
terminals
 color, EDS or non-EDS 96
monochrome, EDS or non-EDS 96
TERR minor 241
TERSE 118
TEST 119
THIN
 Immediate 190
THIRD major 191
THRESHOLD parameter
 in XACB 119
TIME keyword 337
timed screen facility (TSF) 331, 336, 339
 activating 82, 87
cancel while running 340
commands 333
defining entries 84
screen space 340
table 337
terminating 63
using .FGO 48
using .SGO 65
TIO minor 241
TIOT address 313
TLS minor 245
TLV minor 245
TMTR minor 214
total CIs in
 independent overflow 262
root addressable portion of the area 263
sequential dependent overflow 263
unit of work (UOW) 263
TRAC immediate 160
trace table entries 160
TRCE 232
TSF OPTN keyword 87
TSO mode 27
 under VTAM (VTT) 28
TTIM
 Immediate 191
TWRT minor 241
TWSS minor 214
TYPE minor 227
tape mount messages 306
TCB
### Index

#### U
- UCB hex dump **175**
- unallocating datasets **323**
- underscore (_) for comment lines **38**
- underscoring exception message **119, 120**
- unit address **172**
- user area
  - limit **310, 311**
  - top **311**
  - unallocated **311**
- user area extended
  - top **310**
  - unallocated **310**
- user interfaces **22, 23**
  - CUA interface **23**
  - historical information (EPILOG) interfaces **23**
  - menu interface **23**
- user profile **96**
- user profile facility **82**

#### V
- variable
  - OMEGAMON-defined **72**
  - setting **71**
- variable names
  - .VAR **72**
  - in .FGO **49**
  - in .SGO **66**
- VDDN minor **227**
- VERBOSE **118**
- virtual memory **152**
- VMEM immediate **148**
- volser **168**
- volume serial, selecting disks by **168, 169**
- VRDS minor **241**
- VS **109**
- VSAM **217, 227, 239**
  - buffer pool prefix **132**
  - buffer subpool header **133**
  - exceptions **109**
- VSUB major **240**
- VTAM
  - display users (.VTM) **74**
- VTAM director **28**
- VTAM mode **28**
  - code on INFO-line **28**
- VTAM mode VTM **28**
- VTD
  - code on INFO-line **28**
- VTS
  - code on INFO-line **28**

#### W
- WADS dataset **254, 255, 258**
- WADSn **255**
- WAIT minor **215**
- WBLK minor **258**
- WCIO minor **258**
- WDDS major **258**
- WKAP major **235**
- WKST minor **215**
- WLCT minor **246**
- WLRL minor **258**
- work area
  - for PEEK command **307**
  - size of PEEK command **308**
- work area for PEEK command **83, 102**
- workarea pool statistics **235**
- working set
  - fence limits **131**
  - size **215**
- WOWN minor **246**
- wraparound I/O# **174**
- WSIZ minor of PEEK **101**
- WSPL minor **246**
- WSTA minor **258**
- WSTR minor **246**
- WSWT minor **246**
- WTEE minor **207**
- WTO messages, displaying IMS **301**
- WTOR LTERM **302**
- WTOT minor **246**
- WTTE minor **207**
- WTUE minor **207**
- WUCB minor **258**
- WUSE minor **259**
- WVOL minor **259**

#### X
- X (in label field of TCBS) **319**
- XACB immediate **99, 117, 332, 334, 341**
- XCRB minor **264**
- XDSK minor **177**
- XFPQ immediate **124**
- XGRP immediate **109**
- XGSW immediate **122**
- XIMS immediate **110, 340**
- XLF **75**
- XLF OPTN keyword **87**
- XLFSORT **75**
- XLTQ **125**
- XLVL immediate **95**
- XMCH immediate **291**
  - example **292**
XMCH minor 276
XMLS immediate 276, 284, 314
XMM
code on INFO-line 28
XMSC immediate 276, 293, 294
element of hex and character 294
XMZP immediate 276, 298, 314
XR 109
XRF
exceptions 109
XSETG immediate 186
XSUM immediate 110
XTRP immediate 112

Z
zapping
data-only space storage 329
zone initial block 135