

Trademarks and registered trademarks of other companies: AIX, DB2, MQSeries and WebSphere are registered trademarks of International Business Machines Corporation. Citrix, WinFrame, and ICA are registered trademarks of Citrix Systems, Inc. Multi-Win and MetaFrame are trademarks of Citrix Systems, Inc. SAP is a registered trademark and RS is a trademark of SAP AG. UNIX is a registered trademark in the U.S. and other countries, licensed exclusively through X/Open Company Ltd. HP/UX is a trademark of Hewlett-Packard Company. SunOS is a trademark of Sun Microsystems, Inc. All other company and product names used herein may be trademarks or registered trademarks of their respective owners.

Copyright © August 2004, Candle Corporation, a California corporation. All rights reserved. International rights secured.


NOTICE: This documentation is provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions set forth in the applicable license agreement and/or the applicable government rights clause. This documentation contains confidential, proprietary information of Candle Corporation that is licensed for your internal use only. Any unauthorized use, duplication, or disclosure is unlawful.
# Table of Contents

**List of Figures** ........................................... 5
**List of Tables** ........................................... 7
**Preface** .................................................. 9
- About This Book ........................................... 10
- Adobe Portable Document Format ......................... 11
- Documentation Conventions ................................. 12

**What's New In Version 630** ................................ 15

**Chapter 1. Product Overview** ................................. 17
- Product Introduction ..................................... 18
- The Logical Tuning Approach .............................. 19
- Modes of Operation ...................................... 20
- Product Maintenance .................................... 20

**Chapter 2. Using OMEGAMON for VM with Menus** ............... 21
- Starting OMEGAMON ..................................... 22
- Operating OMEGAMON .................................... 22
- Help Screens ............................................. 24
- Exception Recommendations .............................. 25
- Menu PF Key Definitions ................................ 25
- Stopping OMEGAMON ..................................... 26

**Chapter 3. Using OMEGAMON for VM with Commands** ............ 27
- Screen Format ........................................... 29
- Command Format ........................................ 32
- Types of Commands ...................................... 33
- Command Help ........................................... 34
- Command Processing ..................................... 36
- Command Arguments ..................................... 38
- Screen Control ......................................... 39
- Using Program Function (PF) Keys ....................... 44
- Logging OMEGAMON Screens ............................ 46

**Chapter 4. Screen Spaces** .................................. 49
- Invoking Screen Spaces ................................ 50
- Creating Screen Spaces ................................ 50
- Designing Screen Spaces ................................ 51
- Modifying a Screen Space ............................... 52
<table>
<thead>
<tr>
<th>Chapter 5. Exception Analysis</th>
<th>59</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Exception Analysis</td>
<td>60</td>
</tr>
<tr>
<td>Exception Analysis Operation</td>
<td>61</td>
</tr>
<tr>
<td>System Exceptions and VM Exceptions</td>
<td>67</td>
</tr>
<tr>
<td>Exception Analysis Summary Commands</td>
<td>69</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 6. Impact And Bottleneck Analysis</th>
<th>71</th>
</tr>
</thead>
<tbody>
<tr>
<td>An Introduction to Impact And Bottleneck Analysis</td>
<td>72</td>
</tr>
<tr>
<td>Data Collection</td>
<td>72</td>
</tr>
<tr>
<td>Data Reporting for Impact Analysis</td>
<td>73</td>
</tr>
<tr>
<td>Data Collecting and Reporting for Bottleneck Analysis</td>
<td>74</td>
</tr>
<tr>
<td>Usage Examples</td>
<td>76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 7. Performance Analysis</th>
<th>81</th>
</tr>
</thead>
<tbody>
<tr>
<td>The First Step—Exception Analysis</td>
<td>82</td>
</tr>
<tr>
<td>CPU Analysis.</td>
<td>82</td>
</tr>
<tr>
<td>Real Storage Analysis</td>
<td>87</td>
</tr>
<tr>
<td>Paging Analysis</td>
<td>92</td>
</tr>
<tr>
<td>I/O Analysis</td>
<td>94</td>
</tr>
<tr>
<td>Analysis of the DOS/VSE Virtual Machine</td>
<td>97</td>
</tr>
</tbody>
</table>

| Index | 99 |
List of Figures

FIGURE 1. .ZAP Command Display .................................................. 20
FIGURE 2. Initial OMEGAMON Menu Screen .................................... 22
FIGURE 3. Typical Exception Analysis Display ................................. 23
FIGURE 4. Typical Help Screen ....................................................... 24
FIGURE 5. General Help Screen ..................................................... 24
FIGURE 6. Exception Recommendation ........................................... 25
FIGURE 7. PF Key Definitions .......................................................... 25
FIGURE 8. OMEGAMON INFO-line .................................................. 30
FIGURE 9. Format of OMEGAMON Commands ................................. 32
FIGURE 10. OMEGAMON Command Types ..................................... 33
FIGURE 11. Listing Screen Spaces .................................................... 53
FIGURE 12. Exception Analysis ........................................................ 60
FIGURE 13. Exception Recommendation Screen ............................... 65
FIGURE 14. DPLT Minor Command .................................................. 95
List of Tables

Table 1. Symbols in Command Syntax .........................................................12
Table 2. VM Systems Supported by OMEGAMON and EPILOG for VM Version 630 ........15
Table 3. OMEGAMON Variable Names .......................................................51
This document is a general introduction to the OMEGAMON® for VM performance monitor. It is intended for users who have no previous experience with the OMEGAMON product. The document describes how to

- define performance standards
- monitor the performance according to those standards
- identify the cause of performance problems
- initiate corrective action.

You may want to read this document before you begin to use the OMEGAMON for VM product.
About This Book

Who should read this book
This manual is intended for data center personnel who are responsible for monitoring VM performance, and for systems programmers and performance analysts who are responsible for one or more VM systems.

Documentation set information
The documentation listed below is available for OMEGAMON and EPILOG for VM. Each manual in this documentation set contains a specific type of information to help you use the product.

- OMEGAMON for VM User’s Guide
- OMEGAMON for VM Reference Manual
- OMEGAMON and EPILOG for VM Command Summary
- EPILOG for VM Product Manual
- OMEGAMON and EPILOG for VM Installation and Customization Guide
- Candle Products Messages Manual

Notes:
- Refer to the OMEGAVIEW customization information regarding connecting to OMEGAMON, if you are planning to install OMEGAMON for VM with OMEGAVIEW.
- EPILOG and OMEGAMON for VM are compatible with all levels of OMEGAVIEW.
Adobe Portable Document Format

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

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

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

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

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

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

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

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

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

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

Required blanks
The slashed-b (\) character in examples represents a required blank. The following example illustrates the location of two required blanks.

\beBA*ServiceMonitor\0990221161551000

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

Variables and literals in command syntax examples
In examples of command syntax for the OS/390, VM, OS/400, and NonStop Kernel platforms, uppercase letters indicate actual values (literals) that the user should type; lowercase letters indicate variables that represent data supplied by the user:

LOGON APPLID (ccccc)

However, for the Windows and UNIX platforms, variables are shown in italics:

-candle.kzy.instrument.control.file=instrumentation_control_file_name
-candle.kzy.agent.parms=agent_control_file_name

Note: In ordinary text, variable names appear in italics, regardless of platform.

Symbols
The following symbols may appear in command syntax:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The “or” symbol is used to denote a choice. Either the argument on the left or the argument on the right may be used. Example: YES I NO In this example, YES or NO may be specified.</td>
</tr>
</tbody>
</table>
Table 1. Symbols in Command Syntax

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Usage</th>
</tr>
</thead>
</table>
| [ ]    | Denotes optional arguments. Those arguments not enclosed in square brackets are required. Example:  
**APPLDEST DEST [ALTDEST]**  
In this example, DEST is a required argument and ALTDEST is optional. |
| { }    | Some documents use braces to denote required arguments, or to group arguments for clarity. Example:  
**COMPARE {workload} - REPORT={SUMMARY | HISTOGRAM}**  
The *workload* variable is required. The REPORT keyword must be specified with a value of SUMMARY or HISTOGRAM. |
| _      | Default values are underscored. Example:  
**COPY infile outfile - [COMPRESS={YES | NO}]**  
In this example, the COMPRESS keyword is optional. If specified, the only valid values are YES or NO. If omitted, the default is YES. |
What’s New In Version 630

Support for z/VM™

Candle is committed to supplying its customers with a performance monitoring package to support the most recent VM release.

- Version 630 installation drops support of z/VM releases 4.1.0 and 4.2.0 and provides support for z/VM releases 3.1.0, 4.3.0, and 4.4.0 in both 32-bit and 64-bit images. Version 630 also provides support for z/VM release 5.1.0.

VM Systems Supported by Version 630

OMEGAMON and EPILOG for VM Version 630 supports the following VM systems:

Table 2. VM Systems Supported by OMEGAMON and EPILOG for VM Version 630

<table>
<thead>
<tr>
<th>Candle System Code</th>
<th>IBM VM System</th>
<th>IBM Release Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z310</td>
<td>z/VM</td>
<td>3.1.0</td>
</tr>
<tr>
<td>Z430</td>
<td>z/VM</td>
<td>4.3.0</td>
</tr>
<tr>
<td>Z440</td>
<td>z/VM</td>
<td>4.4.0</td>
</tr>
<tr>
<td>Z510</td>
<td>z/VM</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>

System Requirement

OMEGAMON for VM Version 630 requires that the immediate-and-relative instruction facility be present in all z/VM images. If the relative branch instructions are not present, the product will fail with a PRG001 abend.
Corrective Enhancements

Corrective enhancements have been made in Version 630 for changes required by product maintenance through September 2004. The documentation has been updated where appropriate to reflect these enhancements.
Chapter Contents

Product Introduction ................................................................. 18
The Logical Tuning Approach ..................................................... 19
  Defining Performance Standards ............................................. 19
  Monitoring Performance ......................................................... 19
  Identifying the Cause of Performance Problems and Initiating Action ............................................. 19
Modes of Operation .................................................................. 20
Product Maintenance ................................................................. 20
OMEGAMON is a real-time software performance monitor for the VM (Virtual Machine) operating system and runs under the Conversational Monitor System (CMS) operating system. OMEGAMON warns you of exceptional conditions automatically, and also displays the status of VM internal operations and resources in real time. OMEGAMON provides this information through:

**Menus**

To help you access the OMEGAMON features as quickly and easily as possible, OMEGAMON has a full set of menus which automatically execute commands to initiate the type of performance analysis you select. “Using OMEGAMON for VM with Menus” on page 21 provides instructions on using OMEGAMON through the menu system.

**Exception Analysis**

The OMEGAMON exception analysis feature displays messages that warn of existing or impending hardware and software problems from both system wide and individual VM user perspectives. OMEGAMON triggers exception messages when system conditions do not comply with the service levels your installation has set as exception thresholds.

**Impact and Bottleneck Analysis**

The OMEGAMON impact analysis is a unique feature that uses degradation data to identify which virtual machines are competing for the same resources. It allows you to quickly analyze workload contention so that you can take immediate action to solve performance problems. The Bottleneck Analysis feature uses the same data to analyze response time by pinpointing the causes of delays, such as CPU waits, I/O waits, and paging delays.

**Diagnostic Commands**

OMEGAMON can display information about:

- CP control blocks
- DOS/VSE partitions in VM
- VM user’s CPU utilization, storage activity, and I/O
- main storage and DPA utilization
- paging and SPOOLing
- VM trace table
- critical system resource consumption
- minidisk and T-disk utilization
- vector processors

**Versatility**

OMEGAMON commands display many different categories of information, but you can customize the program. The OMEGAMON default settings and optional parameters are supplied from a DATA file called OVUSERcc. You can use the Candle default file, OVUSER99, or customize the settings for your installation. In addition, interfaces allow you to invoke CP commands, CMS commands, and CMS EXECs from within your session.
The Logical Tuning Approach

All of the OMEGAMON features and facilities are designed around the concept of a logical tuning approach for improving the performance of your system. The logical tuning approach consists of these steps:

- Defining standards for VM performance at your installation.
- Monitoring your system to measure actual performance against these standards.
- Identifying the cause of performance problems.
- Initiating action to correct performance problems.

Defining Performance Standards

You define standards for system performance within OMEGAMON by setting exception analysis thresholds. “Exception Analysis” on page 59 explains the procedure.

Monitoring Performance

Monitoring of actual VM performance consists of asking two basic questions:

- Is response time adequate? OMEGAMON commands help you answer this question by monitoring CPU usage, paging, and I/O activity.
- What are the problems in the system? Through the exception analysis feature, automatic warning messages alert you to system problems and OMEGAMON offers recommendations for resolving them.

Identifying the Cause of Performance Problems and Initiating Action

When response time falls below your installation’s performance standards, bottleneck analysis and impact analysis can help you identify the cause of the problem. Bottleneck analysis tells you what bottlenecks within VM may be causing the poor performance, while impact analysis identifies the other virtual machines that are affecting VM. With the information you receive from these two types of analysis, you can initiate action to correct the problem.

By using OMEGAMON in conjunction with its historical companion product, EPILOG™, you can analyze events from different perspectives. EPILOG collects information over a period of time, stores it, and produces reports and graphs that reflect the trends in your installation’s performance. Since OMEGAMON and EPILOG share data collection, you are assured of consistency while monitoring at a lower overhead than would otherwise be possible.
Modes of Operation

OMEGAMON communicates with one or more terminals in any of four modes:

**CMS Mode**
In the CMS interactive mode, OMEGAMON refreshes the screen each time you press Enter. OMEGAMON resides in a virtual machine and runs as a normal CMS transaction.

**Dedicated Mode**
Dedicated mode offers the highest OMEGAMON availability by automatically updating the screen every few seconds. It can be used to run OMEGAMON on a dedicated terminal or as a disconnected virtual machine.

**Low-Speed Mode**
This mode runs OMEGAMON on terminals that do not support 3270 full screen operation. OMEGAMON communicates with the virtual console in line mode (used by TWX and ASCII terminals).

**Cross System Mode**
Use this mode to monitor (on the same screen) two or more Candle OMEGAMON products running on the same or different CPUs.

Product Maintenance

OMEGAMON uses the standard IBM ZAP service aid for all maintenance. See the appropriate IBM Operator’s Guide for details on the use of the IBM ZAP command.

To find out what maintenance has been applied to OMEGAMON, enter the .ZAP immediate command. If maintenance has been applied, the number and version appear. The number to the left of the equal sign (=) is the ZAP number and the number to the right is the update version number.

**FIGURE 1. .ZAP Command Display**

```
.ZAP
+  Zaps Applied (REGULAR)
+  05=1  07=2  08=1 12=1
+  Zaps Applied (SPECIAL)
+  01=1  02=2
+  VM: z/VM 4.2.0  CPU: 2064 #09147E-19147E
+  Product: OM/VM  V610.99
+  Product Version Date: 09/28/03
+  Product Code: 01-048699-144403-09
```

Regular ZAPS are applicable to all installations whereas special ZAPS apply only to specific configurations.

For further information on the .ZAP command, see the OMEGAMON and EPILOG for VM Installation and Customization Guide.
This chapter contains information to help you use OMEGAMON for VM. It describes how to start, use, and stop OMEGAMON using the menus, help screens, and exception recommendations.

Chapter Contents

Starting OMEGAMON .......................................................... 22
Operating OMEGAMON ......................................................... 22
Help Screens ..................................................................... 24
Exception Recommendations ............................................... 25
Menu PF Key Definitions ..................................................... 25
Stopping OMEGAMON .......................................................... 26
Starting OMEGAMON

Note: You will learn the most from this chapter if you perform the instructions at a terminal as you read. The examples that follow assume that your site did not override the initial menu system or startup parameters supplied with OMEGAMON.

OMEGAMON runs as a virtual machine under the Conversational Monitor System (CMS). Invoke OMEGAMON as a CMS application by following this procedure:

1. Log onto a VM terminal.
2. Type in the following EXEC name: OM
   After you press Enter, you will be in OMEGAMON. The first screen that appears is the copyright screen, which declares the Candle proprietary rights to OMEGAMON.
3. When you see the copyright screen, press Enter to continue.
   This starts OMEGAMON in the default CMS mode.

Operating OMEGAMON

FIGURE 2. Initial OMEGAMON Menu Screen

Candle sometimes enhances these menus between releases, so your screen may differ from this example. This menu contains several performance monitoring and tuning options. At the top of the menu are the available PF key definitions. To select a menu option, type the option letter on the INFO-line (top line) starting at the first underscore (_), and press Enter.

For example, to access the EXCEPTIONS option, type E on the INFO-line and press Enter. You will see the following screen.
FIGURE 3. Typical Exception Analysis Display

<table>
<thead>
<tr>
<th>LEXSY</th>
<th>OMEGAMON/VM Exception Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ INQW Warning: In-Queue Wait percentage = 91% (Q1=94%, Q2=74%)</td>
<td></td>
</tr>
<tr>
<td>+ IOWT Warning: System I/O Wait percentage = 56.28%</td>
<td></td>
</tr>
<tr>
<td>+ VMVS VSE215 D Virtual Storage Size = 16384K</td>
<td></td>
</tr>
<tr>
<td>+ VMVS VSP213 D Virtual Storage Size = 16384K</td>
<td></td>
</tr>
<tr>
<td>+ VMVS VTAM DF$RM Working Set Size = 341 PAGES</td>
<td></td>
</tr>
<tr>
<td>+ VMVS VTAMTEST D Virtual Storage Size = 10240K</td>
<td></td>
</tr>
</tbody>
</table>

By default, OMEGAMON selects VIEW EXCEPTIONS from the horizontal menu in the upper portion of the screen, and displays all current exceptions in your system in the lower portion of the screen.

To make another selection, type the appropriate letter (B, C, or D) on the INFO-line and press Enter. The menu in the upper portion of the screen will remain the same, except for the asterisk (*) which will appear in the letter position of the screen you selected. The display in the lower portion of the screen will change according to the selection you make.
Help Screens

OMEGAMON has three types of help screens that provide you with general information, information for a particular display or menu, or exception recommendations for a particular exception. For example, from the VIEW option of the Exception Analysis screen, you can press PF1 to see an explanation of exception analysis.

FIGURE 4. Typical Help Screen

Press PF1 again to display the general Help Menu, which offers selections for help on the various OMEGAMON facilities.

FIGURE 5. General Help Screen

From the general Help Menu, press PF3 to return to the previous help screen, (or press PF4 to return to the Main Menu.)
Exception Recommendations

Another type of help that OMEGAMON gives is recommending a course of action to help you alleviate exception conditions or investigate them further. To get an exception recommendation from the Exception Analysis screen shown in Figure 3 on page 23, place your cursor under an exception name (such as VMVS) and press PF11. The exception recommendation for VMVS follows:

FIGURE 6. Exception Recommendation

> General Help PF1  Back PF3  Up PF7  Down PF8
> VMVS Exception Recommendation
> Explanation:
> This exception message displays the current VM storage size for the user.
> This size can be a factor for determining real storage utilization.
> Some systems and programs (i.e., some compilers and assemblers) use as much virtual storage as is available and cause a large working set size (see VMWS). A large working set size could lead to performance problems such as paging (see PGRT) or swapping (see SWSR).
> Recommendation:
> Reduce the user's virtual storage size by changing the maximum virtual storage size on the user directory statement for that user.

Menu PF Key Definitions

FIGURE 7. PF Key Definitions

> General Help PF1  Back PF3  Up PF7  Down PF8
> PF KEY LIST
> PF1  Help (Repeat PF1=General Help)  PF13  Help (Repeat PF1=General Help)
> PF2  
> PF3  Back to previous screen  PF15  Back to previous screen
> PF4  Main Menu  PF16  Main Menu
> PF5  
> PF6  Print the screen  PF18  
> PF7  Scroll up  PF19  Scroll up
> PF8  Scroll down  PF20  Scroll down
> PF9  
> PF10  
> PF11  Zoom for detail  PF23  Zoom for detail
> PF12  PF24
Note: Do not enter screen names (like ZEXCP) on the INFO-line for quick access, or these key settings may be invalid. If you do so by mistake, you may press PF4 from any point within the menu system to return to the Main Menu and reset the settings shown above.

Stopping OMEGAMON

To stop OMEGAMON at any time, type X on the INFO-line and press Enter. The following confirmation screen will appear.

```
> Enter your selection on the top line.
> EXIT CONFIRMATION
_ M MENU ............ Return to main menu
_ X EXIT ............ End OMEGAMON session
```

To confirm your choice and stop OMEGAMON, type X on the INFO-line and press Enter.
This chapter provides information to help you use OMEGAMON in command mode.

Chapter Contents

Screen Format .......................................................... 29
INFO-line .............................................................. 29
  Screen Space Name ................................................. 30
  Mode of Operation .................................................. 30
  Logging Status ....................................................... 30
  Product ID .......................................................... 30
  Product Version and Parmfile ID ............................... 30
  System ID ........................................................... 30
  Date ................................................................. 30
  Time ................................................................. 30
  Scroll Amount ...................................................... 30
  Automatic Screen Facility ..................................... 30
  Bell Status ........................................................ 30
Command Format ..................................................... 31
  Main Body of Screen ............................................. 31
    label field ....................................................... 31
    command name ................................................. 31
    argument field ................................................. 31
    output display area .......................................... 31
Types of Commands .................................................. 33
  INFO-line ........................................................ 33
  Major ............................................................... 33
  Immediate ......................................................... 33
  Minor ............................................................... 33
Command Help ....................................................... 34
  Major, Minor, and Immediate Command Help .................. 34
  INFO-line Command Help ....................................... 34
  Listing Commands ................................................ 34
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Processing</td>
<td>36</td>
</tr>
<tr>
<td>OMEGAMON Cycles</td>
<td>36</td>
</tr>
<tr>
<td>Automatic Update Interval</td>
<td>36</td>
</tr>
<tr>
<td>Defer Feature</td>
<td>36</td>
</tr>
<tr>
<td>Hold Feature</td>
<td>37</td>
</tr>
<tr>
<td>Command Synonym Feature</td>
<td>37</td>
</tr>
<tr>
<td>Command Arguments</td>
<td>38</td>
</tr>
<tr>
<td>Comment Lines</td>
<td>38</td>
</tr>
<tr>
<td>Rate and Difference Arguments</td>
<td>38</td>
</tr>
<tr>
<td>Screen Control</td>
<td>39</td>
</tr>
<tr>
<td>Up/Down Scrolling</td>
<td>39</td>
</tr>
<tr>
<td>Clear Screen Command</td>
<td>39</td>
</tr>
<tr>
<td>Command Continuation</td>
<td>39</td>
</tr>
<tr>
<td>Continuing Major Command Output</td>
<td>40</td>
</tr>
<tr>
<td>Continuing Major Command Output with Minors</td>
<td>40</td>
</tr>
<tr>
<td>Inserting and Deleting Lines</td>
<td>40</td>
</tr>
<tr>
<td>Color Support or Highlighting</td>
<td>41</td>
</tr>
<tr>
<td>Setting the Color Mode</td>
<td>42</td>
</tr>
<tr>
<td>Setting the Color of Commands and Displays</td>
<td>43</td>
</tr>
<tr>
<td>Secondary Console Control</td>
<td>43</td>
</tr>
<tr>
<td>Using Program Function (PF) Keys</td>
<td>44</td>
</tr>
<tr>
<td>Logging Commands</td>
<td>46</td>
</tr>
<tr>
<td>Printing a Single Screen</td>
<td>48</td>
</tr>
<tr>
<td>Printing Selected Lines</td>
<td>48</td>
</tr>
<tr>
<td>Log Control</td>
<td>48</td>
</tr>
</tbody>
</table>
The following is an illustration of the OMEGAMON screen format:

```
#01 CMS LOG OM/VM   V630.99 CMS   October 2003 10:32:22 5 AB
                  Input area                           Display area
```

The OMEGAMON screen consists of two general areas:

- an input area on the left side of the screen which you use to enter commands.
- a display area to the right of the input area which OMEGAMON uses for display purposes. On the first line of the screen, called the INFO-line, OMEGAMON displays general product and status information. Below that, on the rest of the screen, OMEGAMON displays information in response to the commands you enter in the input area.

**Note:** The last 8 columns on the right side of the screen are not available for command input. The maximum number of characters available for command input on various terminal types is as follows.

<table>
<thead>
<tr>
<th>Terminal Type</th>
<th>Maximum Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>3278 models 2, 3, 4</td>
<td>72</td>
</tr>
<tr>
<td>3278 model 5</td>
<td>124</td>
</tr>
<tr>
<td>3290</td>
<td>152</td>
</tr>
</tbody>
</table>

**INFO-line**

Use the INFO-line command input area to enter OMEGAMON control commands and screen space names. A screen space is a set of commands stored as an image, saved in a file, and executed all at once by typing the screen space name on the INFO-line. After entering an INFO-line command, press the Enter key to execute the command.

If a command is entered on the INFO-line that is not an INFO-line command, OMEGAMON looks for a screen space of that name. If none is found, the message **SCREEN SPACE NOT FOUND** appears and you must correct or erase the input before you can proceed.
The information displayed on the INFO-line follows.

**Screen Space Name**
The 1–8 character name of a file whose filetype is PROCFILE. #01, in this example, is the name for the set of commands contained in #01 PROCFILE. This file is called a screen space.

**Mode of Operation**
The mode in which the 3270 is being accessed. The possible values are: CMS for CMS mode, DED for dedicated mode, DSK for cross system collector, or DIR for cross system director.

**Logging Status**
Indicates if each screen is being copied to the REPORT file. LOG indicates that logging is on; blanks indicate that logging is off.

**Product ID**
Indicates that OMEGAMON for VM is installed.

**Product Version and Parmfile ID**
Indicates the product level. In the example, the product level is Version 610. The suffix indicates the version of the OVUSER parameter file in use. Here, the default parameter file OVUSER99 is in use (filename OVUSER99, filetype DATA).

**System ID**
Indicates the system ID value from the parm field. If omitted, CMS is displayed. This normally identifies which system is being monitored.

**Date**
Indicates the date the screen was last refreshed.
**Time**
Indicates the time the screen was last refreshed.

**Scroll Amount**
Indicates that the top line of the command input area is the fifth logical row of the display. That is, there are four lines above the first line displayed. This field is blank if the physical screen is positioned at the top of the logical screen.

**Automatic Screen Facility**
Indicates that the current display (screen space) was invoked via the automatic screen facility.

**Bell Status**
Indicates whether OMEGAMON is to use the bell on the 3270 to warn when exceptions occur. If this field is blank, the bell is not enabled.

**Main Body of Screen**
The remainder of the screen below the INFO-line is available for entering commands and displaying the output. The following sections explain OMEGAMON commands.
Command Format

FIGURE 9. Format of OMEGAMON Commands

| lcccn_ _ _ _ | output display area (and extended argument field) |
| _ _ _ _ _ _ _ | argument field (usually 1-2 characters) |
| _ _ _ _ _ _ _ | command name (4 characters) |
| _ _ _ _ _ _ _ | label field (1 character) |

**label field**

(column 1) For many commands, this field accepts a character that alters the type of output displayed. It also accepts special characters to request online help text for commands.

**command name**

(columns 2-5) This field contains the command name and is four characters in length. Command names (without a label) start in column 2; however, OMEGAMON adjusts and executes commands you begin in column 1 of the main body of the screen (as long as the characters starting in column 2 do not resolve to a valid command name as well).

**argument field**

(columns 6-7) Many commands accept arguments that modify their function or specify output options. For example, many commands accept an argument of .D or .R, which converts the value to a difference or rate.

*Note: Do not include a blank space between a command and an argument unless the format of the command specifically directs you to do so.*

**output display area**

(column 8 up to the last column) Commands use this area to display the requested information. Some commands, however, require additional input (such as a device address) that extends into this area.
Types of Commands

OMEGAMON uses the following four types of commands:

INFO-line
Always enter INFO-line commands on the top line (INFO-line) and always begin them with a slash (/) in column 2 over the first underline. These commands perform control functions such as saving a screen space (/S), turning on the hardcopy log (/L), 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. Thus you cannot save them in a screen space.

Major
Enter major commands on any line below the INFO-line in columns 2 to 5 of the main body of the screen. These commands select general categories for display, such as system information, resource utilization, or storage utilization.

Immediate
Enter immediate commands on any line below the INFO-line in columns 2 to 5 of the main body of the screen. These commands can perform control functions or monitoring functions. Many control function immediate commands begin with a period and duplicate the functions of INFO-line commands of the same name. Thus you can save them in screen spaces.

Minor
Enter minor commands on any line below the INFO-line in columns 2 to 5 of the main body of the screen. You must precede a minor command with a major command. The minor displays detailed information about the category that the major selects.

Note: You can enter immediate commands between a major command and one of its minors.

The figure that follows shows examples of each type of command.

FIGURE 10. OMEGAMON Command Types

/PRINT________ #01   CMS   OM/VM   V630.99   October 2003   17:03:37   5 B
< /PRINT is an INFO-line command >
SYS       < major command >
,MIN      < immediate command >
fmap      < minor command >
Command Help

This section discusses how to obtain online command help.

Major, Minor, and Immediate Command Help
To request a brief, one-line help for any major, minor, or immediate command, place a question mark (?) in column 1 to the left of the command name.

Notes:
- Minor command names must be preceded by the major command name.
- If the one-line help has a plus sign (+) after it, there is an extended help available which you can request by replacing the question mark in column 1 with a slash (/).

INFO-line Command Help
To request help for an INFO-line command, use the immediate command .ILC. The .ILC command, used by itself, lists all INFO-line commands. If you enter an H in column 1 to the left of .ILC, you get a list of all INFO-line commands along with brief one-line helps. To request help for an individual INFO-line command, enter .ILC cccc, where cccc is the command name.

Listing Commands
The following commands display lists of available OMEGAMON commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.MJ</td>
<td>Lists all major and immediate commands</td>
</tr>
<tr>
<td>.MJC</td>
<td>Lists all major commands</td>
</tr>
<tr>
<td>.MJI</td>
<td>Lists all immediate commands</td>
</tr>
<tr>
<td>.MIN</td>
<td>Lists all minor commands under a major command (the major must precede .MIN)</td>
</tr>
<tr>
<td>.EXM</td>
<td>Executes all minors under major command</td>
</tr>
<tr>
<td>.ILC</td>
<td>Lists all INFO-line commands and aliases or describes one of them</td>
</tr>
</tbody>
</table>

If you enter an H in the label field (column 1) of any of these commands, the online help text for each command is displayed.
.MJC lists all major commands. To list a particular group of major commands, enter .MJCc, where cc is any of the following mnemonic characters:

<table>
<thead>
<tr>
<th>cc</th>
<th>Command Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS</td>
<td>Disks</td>
</tr>
<tr>
<td>EX</td>
<td>Exception analysis commands</td>
</tr>
<tr>
<td>PG</td>
<td>Paging commands</td>
</tr>
<tr>
<td>SY</td>
<td>System information</td>
</tr>
<tr>
<td>TA</td>
<td>Tapes</td>
</tr>
<tr>
<td>VD</td>
<td>Virtual DASD</td>
</tr>
<tr>
<td>VM</td>
<td>Virtual machines (users)</td>
</tr>
<tr>
<td>MS</td>
<td>Miscellaneous commands</td>
</tr>
</tbody>
</table>

For example, display the help text for all OMEGAMON device commands by entering this information:

```
H.MJCDS
```
Command Processing

There are two ways to operate OMEGAMON outside of the menu system. You can enter individual command names on the INFO-line or in the main body of the screen following the format discussed previously. You can also execute screen spaces, which are pre-defined sets of commands that have been saved in a file and given a name. When you type the screen space name on the INFO-line, all of the commands are executed.

You can assign a screen space name to a PF key and execute it with one stroke. Candle Corporation supplies a group of pre-defined screen spaces designed to perform specific types of analyses. You can list them with the SCRN command.

Creating your own screen spaces is discussed in “Creating Screen Spaces” on page 50.

OMEGAMON processes the INFO-line first. If there is an entry there that begins with a slash, OMEGAMON treats it as an INFO-line command. Otherwise, it attempts to find, execute, and display a screen space by that name. If the entry is not valid, OMEGAMON displays an error message.

OMEGAMON then looks at the input area of the main body of the screen. It executes the commands as it finds them, starting at the top of the screen and working down.

OMEGAMON Cycles

An OMEGAMON cycle is the interval between refreshes of the screen data. At each cycle, OMEGAMON reads and processes all commands currently entered on the screen. In any of the interactive modes, the screen is refreshed each time you press the Enter key. In dedicated mode or automatic update mode, you set the interval or accept the default interval and OMEGAMON automatically refreshes the screen.

Automatic Update Interval

In the CMS interactive mode of operation, a cycle is normally the interval between two depressions of the Enter key, however, you can use the /INTn command with a time interval specified to enable automatic updating.

The default cycle time in dedicated mode is specified in the OVUSER file as five seconds. You can change this time in OVUSER, or you can adjust it for the current session with the .INTn immediate command or the /INTn INFO-line command (where nnn can be any number of seconds between 0 and 999).

Defer Feature

When you are in automatic update mode and you attempt to make changes on the OMEGAMON screen, OMEGAMON defers processing of the line in order to avoid executing half-entered input. The message DEFER appears at the right of the top line and commands do not execute until you move the cursor off the line.
Hold Feature
To freeze the screen image when OMEGAMON is automatically updating, position the cursor at row 1, column 1—the blank space in the first column of the INFO-line. The information on the screen does not change until you move the cursor. You can use this hold mode feature while you are creating screen spaces or when you want to print the current screen output with the /PRINT command. The screen remains in hold mode until you press the Enter key.

Command Synonym Feature
With OMEGAMON for VM you can build synonyms for commands with the SYNONYM parameter in the OVUSER DATA file or dynamically with the .SYN immediate command. You can define a character string to be equated to an OMEGAMON command, along with operands that provide a certain selection or display criteria. This feature enables you to operate OMEGAMON using command names that are meaningful to you, and saves you the trouble of entering keywords and/or parameters each time.

Here is an example defining a synonym called SYSDASD in the OVUSER file.

    SYNONYM SYSDASD 'DASD VOL=(VM*,SYS*)'

When you enter SYSDASD on the screen, OMEGAMON processes the DASD command for all volume serials beginning with VM or SYS.
Command Arguments

OMEGAMON has a number of standard labels which you can enter in column 1 before a command to modify its meaning. You already saw in “Command Help” on page 34 how to use a ?, or / in column 1 to request help for a command. There are also standard arguments that you can add to the end of a number of commands to modify the form of the output.

Comment Lines

You can designate comment lines by placing a greater-than symbol (>) in column 1 of the command input and display area. Any command entered on that line will not execute. OMEGAMON places a greater than symbol in column 1 when some commands are executed to prevent automatic re-execution. The greater-than symbol also appears in column 1 of all help texts.

Rate and Difference Arguments

For many minor commands, you can display the output as a rate rather than as a single value. If you enter .R in columns 6 and 7 after the command, OMEGAMON displays the output as a rate per second. For certain commands where a rate is not appropriate, a percentage displays. The following example shows the .R argument used to display a percentage.

```
VALL   USER1   USER2
vtim.R 12.7    6.1
```

Here, VALL, a major command used to display all user virtual machines, selects USER1 and USER2. VTIM.R, a minor command which displays the amount of virtual CPU utilization for the user since logon, displays CPU utilization as a percentage of the total virtual CPU time during the previous OMEGAMON interval. Without the .R argument, the display would be shown as the virtual CPU time used by the virtual machine in hundredths of a second.

Placing an argument of .D in columns 6—7 of many commands causes OMEGAMON to display the output as a difference between the two most recent values of the parameter. For example:

```
SYS
frea.D  2 <available free storage frames have increased by 2>
```

OMEGAMON needs data from two cycles to calculate a rate or a difference. During the initialization cycle, eight periods (........) appear as the output. If a major command selects different items in a command group from cycle to cycle, the output continually indicates that it is initializing. An example of this is the DSKB major command, which selects busy disks. At each screen update, this command may select different disks. In that event, eight periods appear each time, indicating that rates cannot be calculated.
Screen Control

This section describes how to control various aspects of your screen’s appearance, such as scrolling, clearing the screen, and displaying multiple lines of output data for a command.

Up/Down Scrolling

During startup, you can define a logical screen size for OMEGAMON that is larger than the physical screen size. You do this with the LROWS parameter. You can then scroll up and down to display the full logical screen.

To scroll the window up and down the logical screen, use the /UP (/U) and /DOWN (/D) INFO-line commands. They both accept a numerical argument specifying the number of lines you want to scroll.

As an alternative to /UP and /DOWN, you can enter /TOP to scroll to the top of the logical screen, and /BOTTOM (or /BOT) to scroll to the bottom of the physical screen.

PF keys 19 and 20 are set by default to /UP and /DOWN to scroll a physical screen at a time.

If you are using /UP or /DOWN with the \texttt{nn} argument, enter the argument on the INFO-line before you press the PF key to change the scroll amount.

\textbf{Note:} While you are scrolling, OMEGAMON does not update command output.

Clear Screen Command

If you have a Candle menu, screen space, or any commands displayed on your screen, and you want to clear the screen to make space to enter commands, you can do so with the clear screen immediate command. The clear screen command consists of two periods (..) followed by two or more blanks. You can enter the clear screen command on any line of the screen below the INFO-line to have it clear all the lines below it.

1. Move the cursor to row 2, column 2 of your screen, just below the INFO-line.

2. Type two periods, clear any text in the next two columns to the right, and press Enter, like this:

```
..bb
```

The entire screen below the INFO-line will now be blank.

Command Continuation

Some major commands select a series of items. In many of these cases, the display of the output will not fit onto one line. When this is the case, entering the major command by itself displays only the first line of output, and a plus sign appears at the right of the line to indicate that there is more data available.

If you want to see a count of the number of items that a major command will select, enter a pound sign (\#) in column 1 before the command. The following example shows that there are 20 online disks for the DISK command to display.

```
#DISK 20
```
Continuing Major Command Output

There are different continuation characters and commands that you can use to control the output display. The simplest method is to generate all major command output at once by entering a less than symbol in column 1 in front of the major command when you execute it. For example:

<table>
<thead>
<tr>
<th>DISK</th>
<th>VMXA04</th>
<th>VMXA05</th>
<th>VMSP50</th>
<th>VMHP02</th>
<th>OMONVM</th>
<th>DOSTST</th>
<th>DP215R</th>
<th>DOSRES +</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPM023</td>
<td>DBRCRPROD</td>
<td>DLSPROD</td>
<td>IRLMPROD</td>
<td>MPP01</td>
<td>MPP02</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

There are two alternate methods of viewing the additional lines of major command output available after you have issued the major command the first time. They are: putting a number in column 1 to specify which row of output you want to see, or repeating the major command. Repeating the major command displays only the next line of output.

Continuing Major Command Output with Minors

When a major command lists a series of items that continues for more than one line, any minor command you enter after the major applies only to the last line of output. Thus, if you enter a minor command after the third line of major command output, that minor command only generates information about the third line of items listed by the major. Normally, the major and minor must be repeated once for each line of available output.

The .RC command is a shortcut to this process. .RC automatically repeats the major and the minor until all available lines of data are displayed. Enter the major command once followed by the minor command, and then enter RC below the minor. The output will look like this:

<table>
<thead>
<tr>
<th>DISK</th>
<th>VMXA04</th>
<th>VMXA05</th>
<th>VMSP50</th>
<th>VMHP02</th>
<th>OMONVM</th>
<th>DOSTST</th>
<th>DP215R</th>
<th>DOSRES +</th>
</tr>
</thead>
<tbody>
<tr>
<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>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Inserting and Deleting Lines

To insert blank lines on the OMEGAMON screen, use the following immediate command:

```
.i nn
```

indicates the number of blank lines to insert (default is one)

indicates line insertion

OMEGAMON inserts the new line(s) above the line where you typed the insert command. Therefore, all screen space lines below the inserted line shift downward.
When the command executes, the line you typed over with the insert command is restored with its original data. For example:

<table>
<thead>
<tr>
<th>Original Entry</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALL</td>
</tr>
<tr>
<td>CLSS</td>
</tr>
<tr>
<td>CPID</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALL</td>
</tr>
<tr>
<td>.i</td>
</tr>
<tr>
<td>CPID</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALL</td>
</tr>
<tr>
<td>CLSS</td>
</tr>
<tr>
<td>CPID</td>
</tr>
</tbody>
</table>

Note that OMEGAMON cannot insert blank lines between a major command’s multiple lines of output.

To delete lines from a screen space, use the .D nn immediate command. This command works the same way as the .i nn command.

You can also delete a block of data from the physical screen by entering .DD on the first line and .DD on the last line of the block to be deleted.

**Color Support or Highlighting**

OMEGAMON lets you highlight certain fields on the screen for 327x non-color terminals, and basic and extended color support for the 3279 color terminal (includes all terminals in the 3279 family of color terminals). OMEGAMON also supports 3290 non-color terminals. The highlighting and color features distinguish between the main body of the screen, the INFO-line input area, and the INFO-line display area. These features help exception messages and separation lines stand out on the screen as well as differentiating between major, minor, and immediate command displays.

- Highlighting means that certain fields appear more intense than others. On non-color terminals, text is normally not highlighted, however, you can issue commands that cause your basic input text, one or more types of command displays, separation lines, or exception messages to be highlighted.

- Basic color support allows four colors to appear on your color terminal—red, blue, green, and white. The INFO-line input area appears in green. The INFO-line output is blue. The underline cursor is white. General text is green, but there are commands which can cause your basic input text, separation lines, or any exception messages to appear in red.

- Extended color support allows seven colors to appear on your color terminal—red, blue, green, white, turquoise, yellow, and pink. The INFO-line always appears as green, but you can control the color of each command type (major, minor, immediate), each separation line, each exception message, and basic text.
Note that the 3290 also has extended color support, but the display results are underlining, or other forms of emphasis determined by the 3290 hardware options. In addition, exception messages can appear in reverse video, with underscores or blinking.

**Setting the Color Mode**

The immediate command .CLR accepts these arguments which select the kind of color support or highlighting you want for your OMEGAMON session:

- **ON**
  
  Turns basic color support or highlighting on. When you turn on basic color support for a non-color terminal, you make it possible to highlight basic text, each command type, each exception message, and each separation line. On a color terminal, you can control whether each of these elements is red or green.

  The .SCC command has four options to control the color of basic text (the default color), major commands, minor commands, and immediate commands. (See the next section for more information.) You can also set highlighting or color for each exception message and each separation line individually.

- **OFF**
  
  Turns off highlighting differentiation, and basic and extended color support. For the 327x terminal, all fields appear in high intensity. For the 3279 terminal, all fields appear in red except the INFO-line, which is green (input area) and blue (output area).

- **XON**
  
  Turns on extended color support and highlighting, allowing you to use seven different colors on the screen. This capability is only available with 3279 color terminals. With this on, you can use the .SCC command to select the default color and the color of major commands, minor commands, and immediate commands.

  You can set the color of each separation line and each exception message individually. You can also specify reverse video, underscored, and/or blinking for each exception message.

  **Caution**

  Do not turn on extended color unless you have a terminal with extended color support. If you do this accidentally you may get a screen error. You can press the CLEAR key to resume the session, but this action also clears the current security authorization and the current screen space.

- **XOFF**
  
  Turns off extended color support, returning the terminal to basic color support.

- **LO**
  
  Turns off high intensity. Use this feature when many lines of high intensity, such as are produced by the EXSY command, are unacceptable. On non-color terminals, no highlighting appears. On 3290 terminals, which use underscoring to show intensity, underscores are removed. If basic color is in effect, all text in the command area appears green, regardless of how you have individual items set. If extended color is on, the request to turn on low intensity is ignored.

- **HI**
  
  Turns on high intensity capability (the default). This argument reverses a LO entry, restoring highlighting capabilities on non-color terminals, and the red option for items you specify on terminals with basic color in effect.

The default color mode is set in the OVUSER module. To change the default, modify the COLOR keyword of the OPTIONS parameter.

To check the current setting, enter .CLR without an argument.
**Setting the Color of Commands and Displays**

The .SCC immediate command controls highlighting or color of major, minor, and immediate command displays, as well as the default color setting. The default color setting is used for comments, basic text in the command area, and for any command types and exception messages for which you do not specify a color.

To use the .SCC command, first enter .SCC without any arguments. The current color settings will be displayed. For example:

```
.SCC    Major RED    Minor BLUE    Immed GREEN    Default PINK
```

To change a color, type the first two letters of the color you want over the current color setting, and blank out any remaining characters that are still displayed in the field. To return to the default setting, type DE over the current setting.

On terminals with extended color in effect, you can request any of these seven colors:

- RE (RED)
- GR (GREEN)
- BL (BLUE)
- WH (WHITE)
- TU (TURQUOISE)
- YE (YELLOW)
- PI (PINK)
- DE (DEFAULT)

On terminals with basic color in effect, selecting any color other than the default for any of the categories causes displays of that type to appear in red rather than green. On non-color terminals, selecting any color other than the default for any of the categories causes displays of that type to appear highlighted.

**Secondary Console Control**

OMEGAMON provides secondary console support, allowing you to set up multiple 327x and 3290 terminals to be used either for output only, or for both input and output.

A secondary console can be any device that is at least as large as the primary console, providing it has the same number of columns (with the minimum being 80), and at least as many rows. For example, if the primary console is a 327x model 3, a secondary console could be a model 3, model 4, or 3290. It can be ATTACHed to the OMEGAMON session, DIALed, or dynamically allocated.

Secondary console support is available only in dedicated mode. Refer to the OMEGAMON for VM Reference Manual for more information.
Using Program Function (PF) Keys

You can use the PF keys to save keystrokes in order to invoke screen spaces or INFO-line commands quickly. OMEGAMON allows you to assign screen spaces or INFO-line commands for up to 99 PF keys with the .PFK immediate command. Invoke assigned PF key settings with the physical PF keys on your keyboard or by typing the PF key number on the INFO-line.

For example, you might want to assign the /LOGON INFO-line command to PF19 and the LOGOFF command to PF20 to turn the printed log on and off. Type comment text following a slash and asterisk (/*).

```
.PFK 19=/LOGON   /* Turns the printed log on
.PFK 20=/LOGOFF  /* Turns the printed log off
```

**Note:** Use the same format to assign screen space names to PF keys.

You can redefine several PF keys at once without having to reenter 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:

   ```
   +.PFK07=/U               /* Scroll Up
   ```

2. For each new assignment, blank out the plus sign (+) in front of .PFK and type the new assignment following the equal sign. When you press Enter, the assignments are in effect for the duration of the session.

   To cancel a PF key definition, enter:

   ```
   .PFK nn=_
   ```

   OMEGAMON requires the underscore following the = sign. If you enter .PFK without arguments, current values and short descriptions of all currently-defined PF keys appear. PF keys without assignments do not appear on the screen.

   **Note:** The screen spaces and INFO-line commands you assign with .PFK stay in effect only for the duration of the session. When using OVINITZZ as the first screen space (this is the default, which invokes the standard OMEGAMON menu system), the default PF keys for the standard menu system are set in the screen spaces called @ZPFKNEW and @ZPFK2. If you wish to change the PF key assignments used in the menu system, you can make the changes in these screen spaces.

   If you are not using OVINITZZ as the first screen space, PF key assignments can be made in the screen space called @ZPFKDEF, which is invoked just before your first screen space. This screen space has an assignment for PF4 to invoke the OVINITZZ screen space, which will, in turn, invoke the standard OMEGAMON menu system. If you wish to access the standard menu system after invoking your own first screen space, leave the PF4 assignment as defined, or set up another PF key to invoke OVINITZZ.
If you are specifying a first screen space, other than OVINITZZ, the following PF keys will also be set for your convenience:

- **PF3 / PF15**  
  Stop (exit Omegamon)
- **PF6 / PF18**  
  /P (print the current screen)
- **PF7 / PF19**  
  /U (scroll up)
- **PF8 / PF20**  
  /D (scroll down)

**Note:** Refer to the OMEGAMON for VM Reference Manual and OMEGAMON and EPILOG for VM Installation and Customization Guide for more information regarding first screen space (FIRSTSS).
Logging OMEGAMON Screens

OMEGAMON lets you print a copy of the logical screen via a log facility. OMEGAMON dynamically allocates a print file for this purpose unless an OVREPORT file has been defined with the CMS FILEDEF command.

The SPOOLing characteristics for the print file are derived from the current attributes of the user’s virtual printer. You can display or alter the characteristics using the .REP immediate command.

For example:

```
.REP CLASS=A COPY=1 NODE=local DEST=SYSTEM FOLD=YES FORM=STD
```

The keywords are described as follows:

- **CLASS=** The SPOOL class for the file (default is A).
- **COPY=** The number of copies to print (default is 1).
- **NODE=** The name of the network node (default is local node).
- **DEST=** The user ID, printer ID, or remote printer ID (default is system).
- **FOLD=** All lowercase characters are translated to uppercase before being written to the log file (default is yes).
- **FORM=** The form name.

You can modify these default characteristics by typing over the current setting and pressing Enter.

You can also change your defaults with the REPORT parameter in OVUSER. See the OMEGAMON and EPILOG for VM Installation and Customization Guide for details on the REPORT parameter.

If you modify any of the first three parameters, the SPOOL file closes. If the DEST field is blanked out, it appears as SYSTEM. OMEGAMON dynamically reopens REPORT the first time /LOGON is issued.

A page of output consists of 60 lines, which accommodates two screen images. To change the number of lines per page, adjust the LINECT parameter in OVUSER.

If you direct the log to a disk file, you must specify the DISP=MOD parameter on the OVREPORT FILEDEF statement to prevent CMS from repositioning the file to the first record. For example:

```
R;
FILEDEF OVREPORT DISK fn ft fm (DISP MOD
```

Logging Commands

To turn on the log, enter:

```
/LOGON or /L < in the INFO-line input area >
```

or

```
.LOGON < starting in column 2 anywhere below the INFO-line >
```
The word LOG appears on the INFO-line when the log is activated. OMEGAMON logs the screen output each cycle as long as the log is on.

When you want the log to print but to continue logging subsequent data, enter the /LOGOUT INFO-line command or the .LOGOUT immediate command. This action closes the current log file, then dynamically reallocates a new log.

To turn off the log, enter:

```
/LOGOFF or /F   < in INFO-line input area >
```

or

```
.LOGOFF   < starting in column 2 anywhere below INFO-line >
```

The .LOG immediate command also has PUSH and POP arguments (.LOGPUSH and .LOGPOP) that enable you to change the log’s status when you branch to another screen space with .SGO or .FGO and automatically return the status to its original state after these screen routines complete.

.LOGPUSH and .LOGPOP give you the capability of interrupting a regular OMEGAMON session, recording selected information from a screen space routine, and returning to your regular session with the original log status restored.

You might, for example, want to record the information on screen space SAMPLE whenever it is invoked. Enter .LOGPUSH at the beginning of the screen space to save the current status of the log. Enter the .LOGON command to insure that this screen will be logged. At the end of the screen space, enter .LOGPO to insure that the log will be restored to its original state. The following figure shows this example:

```
| SAMPLE     CMS PRT OM/VM    V630.99 CMS  October 2003 15:51:16 |
| .LOGPUSH   >> Log inactive. Status saved. << |
| .LOGON     >> Log started. <<             |
| LEXSY       OMEGAMON/VM Exception Analysis |
| + XACP XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX |
| + X Warning: Average CPU utilization = 99% X |
| + XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX |
| + XCPA Warning: CP Assist not activated on CPU 02 |
| + VMVS DOSPRD D     | Virtual Storage Size = 12288K |
| + VMCP MVSA DF     | using 92.5% of the CPU |
| + VMIO              | Executing 42 SIOs/second |
| + VMID OPERATOR     | Has been IDLE for 9:54 Hours |
| + VMWT              | Waiting 9:54 Hours |
| +                   | Disabled Wait State - PSW = 00020000 00000000 |
|================================================================================|
| .LOGPOP   >> Log status restored to inactive. << |
| .SGO START |
```
Logging OMEGAMON Screens

Printing a Single Screen

There may be times when you only want to print a single screen image and not turn the log on. The /PRINT INFO-line command lets you print a single screen image.

If you want to freeze the current screen image before you print, move the cursor to column 1, row 1 to place it in hold mode. OMEGAMON prints exactly what you see, rather than the output for the next OMEGAMON cycle. Press PF21, which has been assigned the /PRINT command, to print the screen.

Printing Selected Lines

To print selected lines from a screen space, use the .PRT command. For example:

```
VUSR
pwss
ress
.PRT  < Screen traced up to this line >
=====
VDSC
EXSY
```

Here, OMEGAMON logs one copy of the lines above the .PRT command and then changes the command to a comment (> .PRT). To log these lines continually, add the H (hold) argument to .PRT (.PRTH). This prevents OMEGAMON from changing the command to a comment.

Log Control

To set a limit to the number of pages of output printed to the log file, use the .PLM immediate command. You can set the default limit with the PAGELIM entry in OVUSER. If you do not supply an operand for the .PLM command, it displays the number of pages left; supplying an argument resets the pages left count. When the count reaches zero, no further logging is allowed until the limit is reset.

This next example sets the report file page limit to 50 pages and then turns the log off.

```
.PLM50
```

When the limit reaches zero, the screen space clears and the following message displays:

```
>> WARNING: LOG SHUT OFF - OUTPUT PAGE LIMIT EXCEEDED <<
```

Press Enter to restore the screen display.

Set the .PLM limit to zero to prevent any information from going to the log. The /P INFO-line command that prints a single screen space is not affected by the .PLM limit; it can cause logging even when the counter is at zero.
A powerful feature of OMEGAMON allows you to create, store, and recall custom screen spaces. A screen space is a set of OMEGAMON commands saved in a disk file. Candle ships many pre-defined screen spaces with OMEGAMON.

Chapter Contents

Invoking Screen Spaces .......................................................... 50
Creating Screen Spaces ......................................................... 50
Designing Screen Spaces ...................................................... 51
  Separator Lines ................................................................. 51
  Comment Lines ................................................................. 51
  Variable Symbols ............................................................. 51
Modifying a Screen Space ..................................................... 52
Loading Screen Spaces ......................................................... 52
Listing Screen Spaces .......................................................... 53
Renaming a Screen Space ...................................................... 53
Deleting a Screen Space ....................................................... 54
Invoking Screen Spaces Automatically ...................................... 54
Using the Zooming Feature ................................................... 56
  Zoom Examples ................................................................. 56
Invoking Screen Spaces

To invoke a screen space, type its name on the INFO-line. Since it is not an INFO-line command, do not precede it with a slash. The following example invokes screen space HELP:

HELP___________ #01 CMS OV V610.99 SYSA October 2003 10:55:12

When you invoke a screen space, its commands replace the commands currently on the screen. The commands then execute as they do when you enter them individually.

Creating Screen Spaces

The screen spaces that are shipped with OMEGAMON are located on the OMEGAMON 191 disk. Screen spaces always have a filetype of PROCFILE. You may create local screen spaces on any disk that is accessible by using an editor. You can also create or modify screen spaces from within OMEGAMON by following this procedure:

1. If you are creating a single screen space, enter the .DEFON immediate command or the /DEF ON INFO-line command to place OMEGAMON in definition mode until you are done. Definition mode inhibits updating and prevents OMEGAMON from executing the commands you have entered.

   Since saving the screen space automatically turns off definition mode, use the /DEF HOLD INFO-line or .DEFHO immediate command if you are creating several screen spaces at once. This holds definition mode until you issue a /DEF OFF or .DEFOFF command.

2. Enter the desired commands on the screen. You can do this at any time on any screen.

3. Enter /SAVE cccccccc,a in the INFO-line input area to save the screen. The variable cccccccc is a 1–8 character name. The variable a specifies whether you want to save the file to A-disk or to virtual storage. (Variable a is a code described in “Deleting a Screen Space” on page 54.)

   If you want to replace an existing screen space with a new screen space of the same name, use /REP cccccccc,a.

4. If you have turned on definition mode with the HOLD option, use .DEFOFF or /DEF OFF to restore automatic updating.
Designing Screen Spaces

Separator Lines

You can use the separator line immediate command =*=*= to form a horizontal line that separates data in a screen space. These lines can provide greater visual clarity in output presentation. The command uses one or two arguments (any keyboard characters) in columns 6 and 7 to form a line. The characters repeat across the screen space. For example, enter this:

====*-

OMEGAMON produces a line similar to this:

====*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

Comment Lines

You can use comment lines wherever appropriate, to explain the commands. Begin all lines of comment text with a greater than (>) sign in column 1.

If you include in the screen space any commands that comment themselves out after execution, make sure that the comment character (>) does not appear in column 1 when you save the screen. Using the definition mode while you design the screen prevents this.

Note: You cannot include INFO-line commands as part of a screen space, unless you are using an editor to create the screen space, because they will disappear as soon as they are executed. Instead, use the equivalent immediate command.

Variable Symbols

OMEGAMON accepts variable symbols as arguments for commands. The .VAR immediate command allows you to define variable symbols for use in designing screen spaces. You can define the variables at OMEGAMON startup or at any other time during your session.

Variable names beginning with the letter Z are reserved for OMEGAMON use. There are some reserved variables that are used by the standard OMEGAMON menu system. They are described in the table that follows. When setting up screen spaces for exception analysis, you can also use a special set of OMEGAMON-defined variables whose values are initialized when an exception invokes a screen space. For a list of the OMEGAMON-defined variables, see the discussion of automatic screen facility (ASF) in the OMEGAMON for VM Reference Manual.

Table 3. OMEGAMON Variable Names

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;ZSCRN</td>
<td>Current or target screen space name.</td>
</tr>
<tr>
<td>&amp;ZSEND</td>
<td>Return screen space name (for PF3).</td>
</tr>
<tr>
<td>&amp;ZSc</td>
<td>Screen space to go to when c is entered, where c represents a letter from A–Z.</td>
</tr>
<tr>
<td>&amp;ZSHELPSW</td>
<td>Used during online help facility.</td>
</tr>
</tbody>
</table>
Modifying a Screen Space

The techniques employed for creating screen spaces are also used for modifying existing screen spaces. You can use an editor to view or modify screen spaces (filetype must be PROCFILE). If you wish to modify an existing screen space from within OMEGAMON, and want to save it under the same name, use the /REP INFO-line command. To save the screen space with a new name, use the /SAVE INFO-line command.

Loading Screen Spaces

The LSCR immediate command loads screen spaces from disk to main storage. By placing screen spaces in storage with LSCR, you make them more available and more easily fetched. Thus, if a disk is not available, you can continue to invoke the screen spaces that you loaded into main storage with LSCR. The format is:

```
LSCR  cccccccc  cccccccc ... cccccccc
```

The variables are screen space names. You can specify screen space names starting in column 7. You can load as many screens as can fit on the input line, up to the last eight columns of the particular terminal. In the following example, OMEGAMON attempts to load screen spaces ZZ1, ZZ2, and ZZ3 from the disk to main storage.

```
LSCR  ZZ1 ZZ2 ZZ3
```

If, for example, OMEGAMON could not find screen space ZZ1, it would display the following messages:

```
+      OB1507 Member not found - ZZ1
+      OB1508 2 members loaded
```
Listing Screen Spaces

The SCRN immediate command lists screen spaces in main storage and on disk. You can limit the list with arguments and/or parameters. Entering the SCRN command followed by B produces a display consisting of all screen spaces on all accessed disks in EBCDIC order from $ through 99999999. Screen spaces in main storage are listed first. The following figure shows a sample display.

FIGURE 11. Listing Screen Spaces

<table>
<thead>
<tr>
<th>SCRN $</th>
<th>THRU 99999999</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-storage screen facility</td>
<td>14 members</td>
</tr>
<tr>
<td>+ #01  #02  #03  #04  #05  #06  #07  #08  #09  #13  #14  #15  #16  #17  #18</td>
<td></td>
</tr>
<tr>
<td>+ OV_191 193, Mode: B, Stat: R/O, Usage: 76.7%, Blks Free: 356</td>
<td></td>
</tr>
</tbody>
</table>

Renaming a Screen Space

Use the RENM immediate command to keep and rename a screen space. You can only rename screen spaces in main storage and on A-disk. The RENM command accepts an argument specifying where the screen space is to be found.

If you want to rename the SAMPLE screen space to EXAMPLE in both main storage and on the A-disk, enter:

```
RENM SAMPLE EXAMPLE
```

The following message appears:

```
> >> Member "SAMPLE" Renamed to "EXAMPLE" Both In-Storage and on A-disk <<
```

The screen space SAMPLE no longer exists.
Deleting a Screen Space

Use the DELT immediate command to delete screen spaces. OMEGAMON deletes screen spaces from main storage, from the A-disk, or from both. Candle-supplied screen spaces are stored on a read-only disk so that they remain intact.

The DELT command accepts an argument specifying where the screen space is to be found.

- **B or b**: Deletes screen space from both main storage and A-disk (default).
- **I**: Deletes screen space from main storage only.
- **D**: Deletes screen space from the A-disk only.

For example, if you want to delete the screen space SAMPLE from both main storage and A-disk, enter:

```
DELTB SAMPLE
```

The following message appears:

```
>       >> Member "SAMPLE" " Deleted Both In-Storage and from disk <<
```

Invoking Screen Spaces Automatically

To cause one screen to automatically invoke another screen, use the .SGO or .FGO immediate command. The format of .SGO (slow go) is:

```
n.SGO cccccccc
```

where:

- **n**: Optional number of cycles OMEGAMON delays before it invokes screen cccccccc. The variable n can be a number from 1—9 or a letter from A—Z (representing 10—35). The default is to immediately invoke the screen space on the next cycle.
- **ccccccc**: 1–8 character screen space name.

OMEGAMON does not invoke the specified screen space until one cycle goes by during which you do not type anything or scroll the logical screen. This does not include a cursor movement (except in automatic update mode). To delay invocation of the screen space, you must actually type characters or scroll.

To delay invocation of the screen n cycles, place a number (n) in the label field of .SGO. During each cycle, OMEGAMON replaces the label field with the next lower number. When the count reaches zero, OMEGAMON fetches the specified screen space.

The .FGO (fast go) command has the same syntax as .SGO, but it provides a high-speed mechanism to execute a series of screen spaces. If you use the .FGO command, OMEGAMON does not wait for at least one cycle to complete or for one screen to appear before it executes and logs the result of a series of commands. Therefore, you do not see any of the screens as they execute.

The .FGO command is disabled after 64 screen space cycles to protect against a looping condition. At that point, OMEGAMON issues a message that asks whether you want to proceed.
To test your .FGO screen spaces, you can disable .FGO (causing it to act like .SGO) by executing the following:

```
.FGO TEST=YES
```

When you are done testing and wish to enable .FGO, execute:

```
.FGO RESET=YES
```
Using the Zooming Feature

The /ZOOM INFO-line command invokes the navigational zoom feature using the cursor as a pointer. 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 assigns whatever value the cursor is on to a pre-defined OMEGAMON variable and loads a screen space @ZSMcccc, where cccc is the name of the current OMEGAMON command.

To understand the following discussion about how the zoom feature works, you need to be aware of the following points:

- The standard OMEGAMON menu system defines PF11 to issue the “/ZOOM @ZSM” INFO-line command. When you Enter this key, a screen space with the name @ZSMxxxx is invoked, where xxxx represents the OMEGAMON command name or the exception name being “zoomed” on. If you do not use the standard menu system, set your PF key for zooming and use any four character screen space name prefix you choose.

- If you are not using the standard menu system, you must set up your zooming screen spaces prior to using /ZOOM.

- When the /ZOOM INFO-line command is invoked, the following variables are created:

  - **&ZOOM**: Data found at the cursor location.
  - **&ZOOMS**: The name of the originating screen space.
  - **&ZOOMC**: The command or exception name field (cols. 2—5).
  - **&ZOOMROW**: A 2-byte hexadecimal value representing the row of the current cursor location.
  - **&ZOOMCOL**: A 2-byte hexadecimal value representing the column of the current cursor location.

- OMEGAMON provides sample zooming screen spaces that you can use to become familiar with both the setup of a zooming screen space and the types of commands for which /ZOOM is most appropriate. To see the names of those screen spaces, use the SCRN command and look for screen spaces beginning with @ZSM.

**Zoom Examples**

/ZOOM may be used for different types of applications. Below are two examples:

1. For the first example, let’s assume you have a screen that invokes the command “3DASD10” to display the 10 most active DASD devices sorted by I/O rate. If you position the cursor on one of the device addresses and Enter PF11, the following sequence of events occurs:
   a. The /ZOOM @ZSM INFO-line command is invoked.
   b. The &ZOOMS variable is set to the device address.
   c. The &ZOOM variable is set to the name of the screen space that contains the “3DASD10”.
   d. The &ZOOMC variable is set to the name of the command (“DASD”).
   e. The @ZSMDASD PROCFILE screen space is invoked.
The @ZSMDASD PROCFILE could contain the following OMEGAMON commands: DASD ALL, ADDR=\&ZOOM, DEV \&ZOOM, DPLT05, or .SGO \&ZOOMS.

This would cause the DASD command output for the device to be redisplayed, followed by a detailed display of the activity for that device (DPLT minor command). Pressing the Enter key would cause the .SGO command to take effect; OMEGAMON would then return to the original screen that contained the “3DASD10” command.

2. Secondly, suppose you have issued the exception analysis command (LEXSY), and an exception condition is displayed that you want to further investigate. Assuming that you have a predefined screen named @ZSMcccc, where cccc is the name of the exception, position your cursor under the exception name on the LEXSY screen and Enter PF11. OMEGAMON executes the commands on your @ZSMcccc PROCFILE to give a detailed analysis of the condition. The standard OMEGAMON menu system uses this approach to provide analysis of the exceptions. To view an example of an exception, you can choose the EXCEPTIONS option from the main menu and zoom on an exception.
Using the Zooming Feature
One of the key questions to ask in the logical tuning approach is “What are the problems in the system?” The OMEGAMON exception analysis is at the heart of the logical tuning approach for improving system performance.

Chapter Contents

Introduction to Exception Analysis .................................................. 60
Exception Analysis Operation ......................................................... 61
  Setting Exception Parameters ...................................................... 61
  Available Parameters ............................................................... 62
  Setting Severity Levels and Colors .............................................. 63
    Assigning Levels and Colors in OVUSER .................................. 63
    Assigning Levels and Colors Dynamically ................................ 64
Responding to Exception Messages ................................................ 64
  Set a New Threshold ............................................................... 64
  View Exception Recommendations ............................................. 65
  Investigating the Condition ..................................................... 66
System Exceptions and VM Exceptions .......................................... 67
  Exception Groups ................................................................. 67
  Creating Threshold Groups ...................................................... 68
Exception Analysis Summary Commands ........................................ 69
Introduction to Exception Analysis

The first step in using exception analysis is to set exception thresholds that reflect your installation’s performance standards. Whenever exception analysis is invoked, OMEGAMON compares your VM operating statistics against the exception thresholds that you have set. When these limits are exceeded, or when other problem conditions occur, an exception message appears. You can then investigate the problem further or respond to the condition. To familiarize yourself with the types of information exception analysis gives you, see the following display.

**FIGURE 12. Exception Analysis**

<table>
<thead>
<tr>
<th>LEXSY</th>
<th>OMEGAMON/VM Exception Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ XACP XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td></td>
</tr>
<tr>
<td>+ X Warning: Average CPU utilization = 99% X</td>
<td></td>
</tr>
<tr>
<td>+ XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td></td>
</tr>
<tr>
<td>+ VMVS DOSPRD  D</td>
<td>Virtual Storage Size = 12288K</td>
</tr>
<tr>
<td>+ VMCP MVSA  DF</td>
<td>Using 92.5% of the CPU</td>
</tr>
<tr>
<td>+ VMIO</td>
<td>Executing 42 SIOs/second</td>
</tr>
<tr>
<td>+ VMID OPERATOR</td>
<td>Has been IDLE for 9:54 Hours</td>
</tr>
<tr>
<td>+ VMWT</td>
<td>Waiting 9:54 Hours</td>
</tr>
<tr>
<td>+</td>
<td>Disabled Wait State - PSW = 00020000 00000000</td>
</tr>
</tbody>
</table>

An explanation of each of the exception messages follows.

**XACP**
The boxed message warns that CPU utilization during the last interval has exceeded the threshold.

**VMVS**
User DOSPRD’s virtual storage size is 12288K, beyond the established threshold. The virtual storage size can be an important factor for determining real storage utilization at installations which define a range of virtual storage sizes users may allocate (with the CP DEFINE STOR command).

The D notation following DOSPRD indicates that this user is disconnected. All VM exceptions, which analyze individual virtual machines, include additional notations indicating if a user is disconnected, favored, has reserved pages, and/or has a minimum working set assigned.

**VMCP**
User MVSA’s CPU utilization over the last OMEGAMON cycle was 92.5 percent. This figure includes both virtual and CP time attributed to the user.

The DF notation following MVSA indicates that this user is both disconnected and favored.

**VMIO**
User MVSA is executing 42 SIO instructions per second. This exception helps determine which user may be causing system I/O problems.

**VMID**
User OPERATOR has been idle for 9:54 hours. This exception also indicates if the user has any pending I/O, logoff, and/or force status.

**VMWT**
User OPERATOR has waited in the run list in a disabled wait state for 9:54 hours.
Exception Analysis Operation

The EXSY command starts exception analysis. Once it is started, an exception message will appear whenever a pre-defined exception threshold is exceeded. Exception messages appear only while the command EXSY is on the logical screen. By entering EXSY with the L label (that is, LEXSY), the exception name appears to the left of each exception message.

When a threshold is exceeded, exception analysis can cause your terminal's alarm to sound. To avoid becoming a nuisance, the alarm sounds once when a threshold is first exceeded, and then only once per minute as long as the threshold is still exceeded.

The alarm can be turned on or off for each exception. However, no alarm sounds unless the system bell is activated using the /BELLON INFO-line command or .BELON immediate command. You can also specify BELL=ON under the OPTIONS parameter in OVUSER. The letter B appears on the INFO-line to show that the system alarm is on.

Setting Exception Parameters

The OVUSER files that are shipped with OMEGAMON contain default threshold settings. You can check your current settings with one of these methods:

1. Examine the OVUSER file that you are using.
2. Enter the XSET major command and then the exception name that you want to check.
3. Enter the XSET major command and then the .EXM command, which invokes all of the exceptions.

Exception names function as minor commands of XSET for the purpose of setting parameters.

In addition to setting thresholds, you can also turn exceptions on or off and set parameters for how you want the exception to appear on your screen. Exception parameters can be changed or set in two ways:

- Permanent default threshold limits and display attributes are specified in a DATA file called OVUSERcc where cc is a two-character identifier. OVUSER99 is the default OVUSER DATA file shipped with the product.

  To change a threshold limit or attribute in the OVUSER file, edit the file by entering the exception name and the appropriate keyword. Type over the existing settings or type in new ones and save the file as OVUSERcc DATA.

- To change each exception threshold dynamically for the duration of a terminal session, use the XSET major command and the exception name. When you enter XSET followed by the exception name, OMEGAMON displays the current thresholds and settings for exceptions. You can change these defaults by typing over the displayed settings. You can add new attributes by typing the parameter into the command line.

The procedure for setting parameters in the OVUSER file is detailed in the OMEGAMON and EPILOG for VM Installation and Customization Guide, and the procedure for setting parameters dynamically is detailed in the OMEGAMON for VM Reference Manual.
Available Parameters

For each individual exception, OMEGAMON gives you the capability of controlling the following parameters:

1. The threshold, which is usually expressed as a percentage.
2. The state (on, off, test, no display). “Set a New Threshold” on page 64 describes testing an exception.
3. The bell (that is, whether or not you want an audible alarm for this exception).
4. The severity level, that is, the color (on color terminals) or intensity (on non-color terminals) of the exception message display. You may want to assign severity levels and colors to draw your attention to more important exception messages.
5. Highlight attributes, such as blinking, reverse video, or underscored messages.
6. Box attributes. You can decide that this exception message should appear in a box when it trips and you can specify severity level and highlight attributes for the box.

**Note:** The following parameters are additional options, but the procedure for changing the settings dynamically differs from the preceding. Enter a label in column 1 in front of the exception name to display keywords associated with each parameter. Type your option after the keyword.

7. The label N allows you to set:
   - The exception message routing. Messages can be routed to a VM user ID other than the OMEGAMON console. The keyword is NOTIFY.
   - The message routing type. You can specify that a message be routed with a header, without a header, or via the CP SMSG protocol. The keyword is XMSGTYPE.

8. The label E allows you to set the Automatic Screen Facility (ASF) and the Exception Logging Facility (XLF) parameters. These are special OMEGAMON features for logging exceptions explained in the OMEGAMON for VM Reference Manual.

9. The label C allows you to set the following parameters for CPU-intensive exceptions: PGFL, SPUS, TDSK, RSCA, and RSCQ. This option is intended to cut down on CPU overhead.
   - The frequency in OMEGAMON cycles for execution. The keyword is EXNCYC (EXecute every N CYCles).
   - A limit on the number of times the exception is allowed to trip. The keyword is STOP.

**Note:** If any of these five exceptions trips when you invoke exception analysis with the EXSY command, you can also control whether their messages continue to display on every OMEGAMON cycle by using the .NXE immediate command.
Setting Severity Levels and Colors

Assigning severity levels to exceptions is a feature that allows you to specify different levels of intensity (on non-color terminals) or different colors (on color terminals) for your exception messages so that your attention is drawn to the more important ones. The severity levels and colors can be coded in OVUSER or dynamically. You can first assign the desired color or intensity to the level and then assign the level to the individual exception, or you can assign the desired color or intensity directly to the exception.

Assigning Levels and Colors in OVUSER

This description applies to individual exceptions, but additionally, you have the capability of assigning levels and colors to VM user and DASD threshold groups. They are described in “System Exceptions and VM Exceptions” on page 67.

1. To first assign the severity level to a certain color or intensity in OVUSER, type in the SETLVLS parameter and the desired setting as shown in the following figure.

```
< Extended color terminals >
SETLVLS LVL1=RED,
        LVL2=YELLOW,
        LVL3=BLUE,
        LVL4=GREEN

< Non-extended color terminals >
SETLVLS LVL1=HI,
        LVL2=HI,
        LVL3=LO,
        LVL4=LO
```

Seven severity levels are available, but not all need to be assigned. Any level which is not assigned a color or intensity (HI or LO) uses the default background color (usually green when using extended color).

2. The COLOR keyword assigns the desired setting to the individual exception.
   a. If you have done step 1, you can now enter the exception name and after the COLOR= keyword, type in the level you have associated with the desired color or intensity. The CHNQ exception is shown here as an example:
      
      CHNQ COLOR=LEVEL1

      In this example, the CHNQ exception is assigned to severity LEVEL 1. The CHNQ exception will now pick up the color or intensity attribute that you assigned to LEVEL 1 in step 1. If you are not on a color terminal, the severity levels change to non-colors, appearing instead as HI and LO intensity.

   b. If you have not done step 1, you can skip it and just type in the exception name and the desired color (for extended color terminals), or HI or LO (for non-extended color terminals) after the COLOR= keyword.
Assigning Levels and Colors Dynamically

1. To first assign the severity level to a certain color or intensity, use the LEVLnn immediate command, where nn is a level number from 1—7. Then type in the desired setting as shown in the following figure:

   < Extended color terminals >
   LEVL01  RED
   LEVL04  BLUE

   < Non-extended color terminals >
   LEVL01  HI
   LEVL04  LO

2. Enter the XSET major command and then the individual exception name.
   a. If you have done step 1, you can now enter Ln after the exception name where n is the number of the level you want to assign. The CHNQ exception is again shown as an example in the following figure:

   XSET  OMEGAMON/VM Exception Analysis Thresholds
   CHNQ   10 On      L1                Box=('*',L2)
   
   In this example, L1 is the severity level for this exception. To change the level to L2, for example, simply type L2 over L1. The severity level for the box is L2, and you can change it in the same way.

   b. If you have not done step 1, you can skip it and just type in a two-letter color code instead of a severity level (for extended color terminals), or HI or LO (for non-extended color terminals) after the exception name. In the following example, the CHNQ exception appears in red, surrounded by a blue box of asterisks.

   XSET  OMEGAMON/VM Exception Analysis Thresholds
   CHNQ   10 ON      RE                Box=('*',BL)

Responding to Exception Messages

When an exception trips, you will want to decide whether you want to change the exception threshold, whether you need more information to help you interpret the exception message, or whether you are ready to take immediate action.

Set a New Threshold

Deciding on exception thresholds is an ongoing process. Ideally, you should set thresholds that cause exception messages to appear only when action is called for. When deciding on appropriate thresholds for your installation, you may want to check some current system values without incurring the CPU overhead of the EXSY command. You can test an individual exception at any time to determine what a normal value is for your system and then set the threshold slightly higher.
To test an exception, use XSET with the exception name on the next line to display the current state (ON or OFF). Type in **TEST** over ON or OFF; this will cause OMEGAMON to bypass the threshold check. Place an **X** in front of the exception name to execute the exception. An exception message will display showing the current system value for that exception.

**View Exception Recommendations**

When you are using the OMEGAMON exception analysis feature from the menu system, you can request an exception recommendation screen for any exception message that is displayed. These screens help you interpret exception messages and determine what follow-up analysis to perform.

To list current exception messages, select **A** from the exception analysis menu. You might see an exception analysis display similar to this one:

```
LEXSY   OMEGAMON/VM Exception Analysis
  + VMCP MVSA           | Using 64.2% of the CPU
  + VMID CMSBATCH D     | Has been IDLE for 43:31 Hours
```

In this example, two exception messages appear.

To request a recommendation screen for the VMCP exception, place your cursor under VMCP and press PF11.

OMEGAMON displays a screen which explains the message and makes some recommendations about what you can do to prevent the exception condition.

**FIGURE 13. Exception Recommendation Screen**

```
> Back PF3
> Explanation:
> This exception message displays the percent of CPU consumed by the VM user over the last interval. This calculation includes both virtual and CP time attributed to the user.

> Recommendations:
> 1) Determine if this user is continually consuming too much of the CPU.
> 2) If consumption is excessive, adjust the user’s relative priority by issuing the following command:
>    CP SET SHARE userid RELative
```

Once you have finished viewing the recommendation screen, press PF3 to return to a menu screen which has exception analysis on display.
Investigating the Condition

Your installation can set up screen spaces in advance to give more detailed information about potential problem conditions. In addition, Candle supplies a number of predefined screen spaces that analyze various areas, such as paging, CPU utilization, active devices, DASD, and main storage. These screens are accessed from PF keys if you are using the Candle-supplied OVUSER file. Enter the .PFK command to check your installation’s PF key definitions.

The following OMEGAMON features can be used to automate and speed up problem investigation.

- The zooming feature allows you to access predefined screen spaces with just the touch of a key. Once in the screen space, you can zoom in on device addresses, individual virtual machines, or other detailed information by pressing the zoom key again.

- The Automatic Screen Facility (ASF) automatically reacts when an exception trips by invoking user-specified screen spaces and logging the command output—all without operator intervention.

- The Exception Logging Facility (XLF) automatically turns on the log to capture detailed information about performance exceptions when they occur. You can use this feature to ensure that intermittent performance problems are documented and subsequently corrected. ASF and XLF are both explained in the OMEGAMON for VM Reference Manual.
System Exceptions and VM Exceptions

Exception analysis collects information on problems that affect the system in general and on problems that relate directly to individual virtual machines. Therefore, exceptions are known as either system exceptions or VM exceptions.

Some system exceptions do not have threshold limits because they check only for the existence of problem conditions rather than exceeded thresholds. These system exceptions generate warning messages whenever the particular problem appears.

Virtual machine exceptions are identified by beginning with the letters VM. You can limit the execution of exception analysis to VM exceptions by using the XVM immediate command in place of EXSY.

Exception Groups

System exceptions and VM exceptions are broken down into five exception groups. The exception group concept allows you to focus on a particular aspect of system performance and control the amount of overhead attributed to exception analysis. Each group has a two-character mnemonic that serves as an identifier. The identifiers can be used with group switches that control the operating state and with two immediate commands, XSUM and XTRP, which give summary displays of exception analysis.

Group switches can be used as minors of XSET to override the operating states set individually for the exceptions within that group. The possible operating states are:

- ON
- OFF
- TEST
- NDSP
- NULL

The following table shows the exception groups with their related group abbreviations and group switches.

<table>
<thead>
<tr>
<th>Exception Groups</th>
<th>Abbreviation</th>
<th>Switch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware Related Exceptions</td>
<td>HD</td>
<td>HDSW</td>
</tr>
<tr>
<td>I/O Related Exceptions</td>
<td>IO</td>
<td>IOSW</td>
</tr>
<tr>
<td>Operational Exceptions</td>
<td>OP</td>
<td>OPSW</td>
</tr>
<tr>
<td>Paging Exceptions</td>
<td>PG</td>
<td>PGSW</td>
</tr>
<tr>
<td>Real Storage Exceptions</td>
<td>RS</td>
<td>RSSW</td>
</tr>
</tbody>
</table>

The following example shows how you can set groups to these operating states.

<table>
<thead>
<tr>
<th>XSET</th>
<th>Exception Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDSW</td>
<td>ON &lt; sets all hardware exceptions on &gt;</td>
</tr>
<tr>
<td>PGSW</td>
<td>TEST &lt; sets all paging exceptions to Test state &gt;</td>
</tr>
<tr>
<td>IOSW</td>
<td>NULL &lt; relinquishes group control of all I/O exceptions &gt;</td>
</tr>
</tbody>
</table>
The NULL state is the default setting. It allows each exception in the group to have its own state. Unless NULL is specified, the group settings override the settings on each exception within that group.

The NEVER state, which can only be set in OVUSER, has ultimate control over an exception. If an exception is set to the NEVER state, it can never be invoked by any command.

**Creating Threshold Groups**

You can also customize exception analysis by creating groups of related items or users, and setting exception thresholds that are specialized for each group. For example, a given threshold may be appropriate for one set of users, but not appropriate for other types of users.

The following parameters in the OVUSER DATA file allow you to create threshold groups. For detailed information on defining threshold groups, see the OMEGAMON and EPILOG for VM Installation and Customization Guide.

**DEFVMTG**

The define VM threshold group parameter lets you define VM users into groups. Each group is a set of virtual machine user IDs or account codes, and each group threshold overrides individual thresholds for any of the VM exceptions. Here is an example of defining a group of users and setting thresholds for the VMID and VMPG exceptions specific to that group. The asterisk (*) is a wild card character.

```
DEFVMTG NAME=(ALLEN,PROFS*,CREDIT*,MIS,GEORG,STAR,
LINC,HOW*),
  VMID=((0,4,0,0),BL,ON),
  VMPG=(10,BL,ON),
```

You can display existing VM user threshold groups with the VMG minor of XSET.

**DASDTG**

The DASD threshold group parameter lets you create groups of DASD. Thus, you can define thresholds for DASD exceptions that are appropriate for each particular group or type of DASD. The DASD exceptions are DBSY, DEVQ, DNRS, DPTH, DRDY, DRPN, DSRV, DVRS, and DVRT.

You can specify DASD threshold groups by name, address, or volser. Here is an example of defining a group of DASD by name and setting thresholds for the DVRT and DVRS exceptions specific to that group.

```
DASDTG NAME=(3380,3350),
  DVRT=(25,RE,ON),
  DVRS=OFF
```

To display a DASD threshold group with its settings, enter the DATG minor of XSET.

**RSCSTG**

The RSCS threshold group parameter lets you define RSCS links into groups. Thus, you can define thresholds for the RSCS exceptions (RSCA and RSCQ) that are appropriate for each particular group or type of RSCS link.

Here is an example of defining a group of RSCS links and setting thresholds for the RSCA and RSCQ exceptions specific to that group. The asterisk (*) is a wild card character.

```
RSCSTG LINKID=(CACO*),
  RSCA=ON,
  RSCQ=20
```

To display a RSCS threshold group with its settings, enter the RSTG minor of XSET.
Exception Analysis Summary Commands

The XSUMcc immediate command shows a summary of exception analysis status for exception group cc. If you omit cc, XSUM shows the status of all exceptions.

In the color column, XSUM shows either the color, severity level, or intensity. Placing an A in the label field shows the value and time of the last and worst occurrences of an exception since OMEGAMON startup.

The next figure shows examples of the XSUM command for the hardware group. The lower half of the screen is an example that uses the A label.

XTRPcc shows tripped exception analyses for group cc. Tripped exceptions are those whose thresholds have been exceeded. If you omit cc, XTRP shows all tripped exceptions. Placing an A in the label field shows the value and time of the last and worst occurrences of a tripped exception since OMEGAMON startup. For example:

In the top half of the screen, the parameters State, Thresh, Gr, Bell, Color, and Notify are the OMEGAMON exception analysis parameters for each exception. The Lim, Pers, Log, Auto, and Scrn Spc parameters relate to the exception logging facility. They are explained in the OMEGAMON for VM Reference Manual.
Both the Bottleneck Analysis and impact analysis facilities are a significant extension to capabilities of OMEGAMON. They greatly enhance your ability to diagnose performance problems and tune your system to increase productivity.

Chapter Contents

An Introduction to Impact And Bottleneck Analysis .............................. 72
Data Collection .............................................................................. 72
Data Reporting for Impact Analysis .............................................. 73
Data Collecting and Reporting for Bottleneck Analysis ..................... 74
   How Bottleneck Analysis Is Used ................................................. 74
   Bottleneck Analysis Commands ................................................. 75
Usage Examples ........................................................................... 76
   Example 1—Bottleneck Analysis for Virtual Machines .................. 76
   Example 2—Bottleneck Analysis for a Performance Group ............ 77
   Example 3—Impact Analysis ...................................................... 78
An Introduction to Impact And Bottleneck Analysis

Bottleneck analysis shows where degradation is occurring and the degree to which each type of degradation is impacting productivity. Degradation is defined as unproductive time or time spent waiting for a system resource when the CP dispatcher has ready work to do. Impact analysis identifies the other virtual machines’ workloads running under VM that are impacting the virtual machine by competing with it for resources. They can be guest operating systems, CMS users, or any other VM application.

Bottleneck and impact analysis function as a two-step process: the first phase is data collection and the second phase is data reporting.

Data Collection

The collector samples the run state of each VM user in the system at regular intervals and provides statistics on whether the virtual machine is using or waiting for specific system resources. For impact analysis, the collector goes one step further and identifies which other virtual machine’s workload is currently using that resource.

The data collection service machine runs in a disconnected virtual machine and communicates with the reporting facility through an IUCV interface. The sample counts are collected every two seconds (by default) and are deposited in 15-minute interval buckets which are synchronized approximately on the quarter-hour. The impact and bottleneck analysis displays report statistics for wait states based on a short-term interval which represents the last two 15-minute interval buckets, and a long-term interval which represents the last four 15-minute interval buckets.

It is important that the collector run at the most favored priority possible for your installation to insure valid statistical sampling. The collector will not analyze system degradation accurately if it is dispatched after other virtual machines have completed their units of work.

The commands to initiate data collection manually are discussed in the next section. However, you can also start collection automatically when OMEGAMON is initialized by specifying the following options in the OVUSER DATA file.

- In the DEX parameter, specify SYSTEM=AUTO to start system-wide collection.
- In the PGNAME parameter, specify IA=AUTO and DEX=AUTO to start impact and bottleneck analysis collection for performance groups.

The collector operation and control parameters are described in the OMEGAMON and EPILOG for VM Installation and Customization Guide.
The reporting capability for impact analysis uses the statistics gathered to generate percentage figures on the degree of contention from workloads identified by the collector. You can obtain reports in two formats:

- A plot-type display.
- A high-level pictorial display, called a Workload Impact Profile, intended to facilitate quick interpretation.

All impact analysis commands are minors of the major command DEX and must be entered after the DEX command. To initiate impact analysis, enter DEX and then the IANL command, followed by the user ID or performance group name you want to specify for collection. Wait 10 to 15 minutes for sufficient collection to take place, then request a display as follows:

1. For a graphic plot display, there are three levels of detail you can choose from. The level of detail and the specification of user or user group for your display are both made with arguments to the IANL command.

   **Level of Detail**

   Enter one of the following in the label field in front of the IANL command:

   - **b** Requests an overview plot display showing the resource contenders.
   - **S** Requests a summary plot display showing the contenders, plus the execution states that are impacting the virtual machine.
   - **D** Requests a detailed plot display showing, in addition to the above, the specific device in contention.

   **User or User Group**

   Enter a space and one of the following after the IANL command.

<table>
<thead>
<tr>
<th>GROUP=groupname</th>
</tr>
</thead>
<tbody>
<tr>
<td>The name of a performance group after the keyword GROUP= to request a display for a particular performance group of users. (Performance groups are predefined in the OVUSER file with the PGNAMES parameter.)</td>
</tr>
<tr>
<td>userid</td>
</tr>
<tr>
<td>A user ID to request a display for a particular user.</td>
</tr>
</tbody>
</table>

2. To request a Workload Impact Profile for either an individual user or for a performance group, enter the command in this format:

   DEX
   IPRO IANL userid
   or
   IPRO IANL GROUP=groupname
Data Collecting and Reporting for Bottleneck Analysis

Data Collecting and Reporting for Bottleneck Analysis

For every sample gathered by the collector, Bottleneck Analysis analyzes every user and determines if and why it is waiting, in other words, its execution state. The reporter for Bottleneck Analysis generates reports that show the actual count of waits, the percentage of total waits attributable to each wait state, a graphic plot of the percentages calculated, or a pictorial display called a Resource Impact Profile.

How Bottleneck Analysis Is Used

Bottleneck analysis reports on the following wait reasons:

- **CFW**: Console function wait (voluntary wait).
- **CON**: Waiting for a console I/O.
- **CPW**: Waiting for CPU.
- **ELG**: Waiting in the eligible list.
- **ELP**: Waiting in eligible list for processor.
- **IDL**: Idle (voluntary wait).
- **IOA**: Waiting for I/O active on DASD device.
- **IOQ**: Waiting for I/O queued to DASD device.
- **IOW**: I/O waits.
- **PAG**: Waiting for Dynamic Paging Area page-in operation.
- **QDR**: Q drop delay (voluntary wait).
- **SIM**: Waiting for instruction simulation.
- **TIO**: Waiting in a test I/O busy condition.
- **TMR**: Idle with timer event pending (voluntary wait).

By sampling at regular intervals, Bottleneck Analysis can statistically estimate the relative impact each wait reason has on system performance. For example, assume that a CMS user's response time is 4 seconds and waits are distributed as follows:

- using CPU 10%
- I/O wait 29%
- waiting for DPA page-in 22%
- waiting for CPU 16%
- eligible list wait 15%
- test I/O busy condition 8%
Data Collecting and Reporting for Bottleneck Analysis

Using CPU is not degradation and accounts for 10% or .4 seconds of the user’s response time. Degradation accounts for the remaining 90%. Statistically, we can estimate the impact of the three types of degradation as follows:

- I/O wait = 29% of 4.0 seconds = 1.16 seconds
- waiting for DPA page-in = 22% of 4.0 seconds = .88 seconds
- waiting for CPU = 16% of 4.0 seconds = .64 seconds
- eligible list wait = 15% of 4.0 seconds = .60 seconds
- test I/O busy = 8% of 4.0 seconds = .32 seconds

By identifying the components of response time and their relative contributions, you can direct your tuning efforts to the most significant bottlenecks.

Bottleneck Analysis Commands

Bottleneck analysis commands are minor commands of the DEX major command. Therefore, DEX must be present on the screen for other commands to work. In using Bottleneck Analysis commands, first start collection on the virtual machines you want to monitor, wait 10 to 15 minutes, and request a display. (Later, to save overhead on the CPU, you may want to stop collection on the virtual machines you have sampled.)

There are three sets of commands that control the Bottleneck Analysis. The following questions will help you to determine your needs.

1. Which workload types would you like to collect data on? The options are:
   - **BEGIN/END** For the entire system.
   - **PON/POFF** For a specific performance group (groups are predefined in the OVUSER file). This command must be followed by the performance group name.
   - **SELV/DELV** For an individual user. This command must be followed by the user ID.

2. Would you like all wait states reported in your displays or would you like to exclude some? The BON and BOFF commands, followed by the three-letter code identifying each wait state, control which wait states are reported. Do you want to display only wait reasons which account for 5% or more of the total? The THRS command controls the threshold.
   If you want all wait reasons displayed and don’t want to change the 5% threshold, the Candle-supplied default settings can be used and these commands do not need to be set for the session.

3. What would you like your display to look like and which user(s) do you want information on? The variable c for each of the following commands can be S for system-wide, P for performance group, or V for individual virtual machine. In the case of performance group or virtual machine reporting, the command must be followed by the group name or user ID.
   - **CNTc** A report of wait state counts.
   - **PCTc** A report showing what percentage of the total wait is attributable to each user.
   - **PLTc** A graphic plot display of the percentages by short-term and long-term intervals.
   - **IPRO PLTc** A pictorial display (Resource Impact Profile) of the long-term percentages.
Usage Examples

The purpose of these sessions is to help you start using impact analysis and Bottleneck Analysis. The OMEGAMON for VM Reference Manual contains complete information.

Example 1—Bottleneck Analysis for Virtual Machines

This example shows how to use Bottleneck Analysis to analyze degradation for a workload.

Let’s say you are a systems programmer and an applications programmer or manager calls you to find out why the programmer compile turnaround time is poor. OMEGAMON is already running, so you only have to select the workload to monitor.

1. Enter the DEX and SELV cccccccc commands in one step.

```
DEX           CMS    OM/VM V610.99    CMS 09/29/03 04:01:03
selv CMSBATCH
```

- **DEX** Major command so you must enter it first.
- **SELV** Starts data collection for a virtual machine.
- **cccccccc** Name of the troubled virtual machine you want to select.

2. Press Enter and you will see the following message:

```
DEX   >> Bottleneck Analysis/VM  <<
selv +     CMSBATCH    - user ID collection started
```

3. Wait 10 to 15 minutes for sufficient data collection to take place. You may bring down OMEGAMON during this time to do other things without affecting your collection request.

4. Enter the display command, PLTV, followed by CMSBATCH and press Enter. The PLTV command displays a plot of bottleneck analysis for CMSBATCH.

```
DEX   >> Bottleneck Analysis/VM  <<
selv pltv CMSBATCH 14:32 MN/     436          14:32 MN/     436
+   ** Degradation **   Short|0.1.2.3.4.5.6.7.8.9.0| Long|0.1.2.3.4.5.6.7.8.9.0|
+   Using CPU           5.6 |> . . . . . . . . . .|  5.6|> . . . . . . . . . .|
+   Eligible list wait 31.0 |--===> . . . . . . . | 31.0|--===> . . . . . . . |
+   I/O wait            12.8 |---> . . . . . . . . | 12.8|---> . . . . . . . . |
+   Waiting for CPU     9.5 |-> . . . . . . . . .|  9.5|-> . . . . . . . . . |
```

Note that the total degradation does not add up to 100%. This is because the threshold (THRSnn command) is set at 5%, which means that wait reasons below 5% are not displayed. Also, the BOFF command has been issued to request that the wait reasons, IDL (idle) and TMR (idle with a timer event pending), not be displayed.
You can see that 31% of the CMSBATCH wait is attributable to waiting in the eligible list. Real storage problems, therefore, account for 49% of CMSBATCH’s time.

5. To remedy the performance problem for CMSBATCH, you can issue the CP SET FAVORED command to prevent CMSBATCH from waiting in the eligible list. Also, by using the CP SET RESERVE command, paging can be reduced or eliminated for this virtual machine.

6. You may want to keep an eye on this virtual machine until after tuning adjustments have been made. You can use a higher level, more graphic display, the Resource Impact Profile, because it displays degradation data in a very easy to read format. Use the IPRO command in conjunction with the PLTV command and press Enter.

Here, you can see that the eligible list wait is no longer a major source of degradation.

**Example 2—Bottleneck Analysis for a Performance Group**

As a performance analyst, you receive a call from the systems programming group advising you that they have been installing and testing the latest version of CP for several days and have been experiencing poor throughput. You have defined the systems programmers as performance group SYSPROG in your OVUSER DATA file. Bottleneck analysis easily lets you analyze the factors that cause degradation for a performance group.

1. Enter the DEX major command. Enter the PON minor command and the performance group name to start Bottleneck Analysis data collection for SYSPROG.
2. Wait 10 to 15 minutes for sufficient data collection to take place. Use the PLTP command along with the group name to display degradation data for performance group SYSPROG.

```
<table>
<thead>
<tr>
<th>DEX</th>
<th>&gt;&gt; Bottleneck Analysis/VM &lt;&lt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>pltp SYSPROG</td>
<td>14:03 MN/ 1686</td>
</tr>
<tr>
<td>+ ** Degradation **</td>
<td>Short 0.1.2.3.4.5.6.7.8.9.0</td>
</tr>
<tr>
<td>+ Using CPU</td>
<td>5.6</td>
</tr>
<tr>
<td>+ I/O wait</td>
<td>43.0</td>
</tr>
<tr>
<td>+ 123 PROD30 Active 16.4</td>
<td>16.4</td>
</tr>
<tr>
<td>+ 123 PROD30 Queued 15.0</td>
<td>15.0</td>
</tr>
<tr>
<td>+ 244 CMSOOA Active 7.3</td>
<td>7.3</td>
</tr>
<tr>
<td>+ Waiting for CPU 13.1</td>
<td>13.1</td>
</tr>
</tbody>
</table>
```

3. Notice that I/O waits account for 43% of wait time, primarily on DASD 123 (PROD30). Since Bottleneck Analysis shows you which resource is a problem and how much degradation is occurring, you decide to use OMEGAMON to find out why the resource is the problem. In this case, you use the OMEGAMON MDSK command and the XSEK command to delve deeper into the problem.

You discover that many virtual machines are contending for different mini disks on the I/O device, Disk PROD30. You suspect that this has been the cause all along. You call the DASD space manager and suggest that the mini disks be spread over more than one pack.

**Example 3—Impact Analysis**

As a systems programmer, you receive complaints from the Engineering Department that response time is poor. You already know from Bottleneck Analysis that the users in that department are waiting for CPU 27% of the time. As your first step, you need to know who is degrading CMS trivial response time. Impact analysis monitoring is exactly suited for this situation.

1. Start collection on the performance group by issuing the following command:

```
DEX
IANL GROUP=engdept
```

2. Wait 10 to 15 minutes and enter the following commands:

```
| IPRO IANL GROUP=engdept | CMS OM/VM V610.99 CMS 09/25/03 10:37:21 |
```
DEX is the major command, IPRO selects a Workload Impact Profile, IANL executes impact analysis, and GROUP= selects the user group for analysis. The resulting display follows.

3. The display shows that performance group QA is the biggest impactor of the Engineering Department response. This would indicate that adjusting the user priorities of the virtual machines within the QA group would alleviate the problem.
Usage Examples
This chapter shows you how to use particular OMEGAMON commands with other commands to analyze both system-wide and user performance. Examples are given to demonstrate how you can use OMEGAMON to analyze the most important aspects of a VM system.

**Chapter Contents**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>The First Step—Exception Analysis</td>
<td>82</td>
</tr>
<tr>
<td>CPU Analysis</td>
<td>82</td>
</tr>
<tr>
<td>TRND - System-wide CPU Utilization</td>
<td>83</td>
</tr>
<tr>
<td>MCPU - Graphic Display of CPU Utilization</td>
<td>83</td>
</tr>
<tr>
<td>VCPU - User CPU Utilization</td>
<td>84</td>
</tr>
<tr>
<td>VSEL - Selecting Users Based on Multiple Criteria</td>
<td>84</td>
</tr>
<tr>
<td>AENV - Graphic User Environmental Display</td>
<td>85</td>
</tr>
<tr>
<td>Real Storage Analysis</td>
<td>87</td>
</tr>
<tr>
<td>SMAP - Real Storage Map</td>
<td>88</td>
</tr>
<tr>
<td>FMAP - Free Storage Usage Map</td>
<td>89</td>
</tr>
<tr>
<td>MRES - Real Storage Allocated to Users</td>
<td>90</td>
</tr>
<tr>
<td>VCOR - Map of Virtual Storage</td>
<td>90</td>
</tr>
<tr>
<td>Paging Analysis</td>
<td>92</td>
</tr>
<tr>
<td>MPAG - System Paging Information</td>
<td>92</td>
</tr>
<tr>
<td>VPAG - User Paging Information</td>
<td>93</td>
</tr>
<tr>
<td>SOWN - Device Allocation Statistics</td>
<td>93</td>
</tr>
<tr>
<td>ALOC - Device Allocation Statistics</td>
<td>93</td>
</tr>
<tr>
<td>I/O Analysis</td>
<td>94</td>
</tr>
<tr>
<td>MIOR - Overall I/O Activity Analysis</td>
<td>94</td>
</tr>
<tr>
<td>DISK - Device Analysis</td>
<td>95</td>
</tr>
<tr>
<td>DPLT - Device Utilization</td>
<td>95</td>
</tr>
<tr>
<td>VIOS - User I/O</td>
<td>96</td>
</tr>
<tr>
<td>Analysis of the DOS/VSE Virtual Machine</td>
<td>97</td>
</tr>
</tbody>
</table>
The First Step—Exception Analysis

Exception analysis provides the first indication of problems within your system. It looks at over 50 problem areas, and although exception analysis does not give you an exact answer, it can show you where to look. After you have identified a problem area, you can then refer to one of the following sections in this guide for the four main analysis resources:

- “CPU Analysis” on page 82
- “Real Storage Analysis” on page 87
- “Paging Analysis” on page 92
- “I/O Analysis” on page 94

CPU Analysis

The following diagram shows the flow of commands you would use to perform CPU analysis.

The TRND major command shows system-wide CPU utilization. MCPU divides the output into utilization by user, type, and CPUs. You can use VCPU to select users with CPU usage over a certain percentage. VSEL and VUSR are other user commands that focus on specific criteria for CPU usage. The AENV minor command shows a graphic display of CPU usage for a specific user. A discussion of these commands follows.
TRND - System-wide CPU Utilization

Assume that the threshold for either the XACP or VMCP exception was exceeded, indicating an abnormally high CPU consumption. First, start with system-wide CPU utilization displays. Use the TRND command to get a quick picture of recent supervisor and problem state CPU utilization. The default is 5 intervals.

```
TRND
+         CPU  SUPR  PROB  STOR  PAGE  INQ  LOG
+         BUSY  STATE  STATE  UTIL   RATE  USER  USER
+current  11%  1%  10%  63%   13    33
+average  15%  2%  13%  62%   13    33
+peak     19%  2%  18%  63%   14    33
```

MCPU - Graphic Display of CPU Utilization

The MCPU command generates a graphic display of system-wide CPU utilization by user, type (queues and performance groups), and for all CPUs. MCPU displays CPU usage over a certain percent, which you must specify with a numeric argument nn. For example, MCPU30 displays CPU usage over 30 percent. Enter MCPU30 starting in column 2. MCPU00 displays all CPU usage, as in the example which follows.

```
MCPU00
+ |USER| CPU% 0_2_4_6_8_0 |TYPE| CPU% 0_2_4_6_8_0 |SYSTEM| CPU% 0_2_4_6_8_0 |
+ |SYSTEM| .3 > . . . . . |Q1| 2 > . . . . . |CPU2: 97 ---========>.|
+ |DOS001| 1.0 > . . . . . |Q2| 92 ---========>. |VIRT: 90 ---========>.|
+ |DOS002| .2 > . . . . . |DOS| 2 > . . . . . |CP: 7 > . . . . .|
+ |DOS003| .7 > . . . . . |MVS| 93 ---========>.|
+ |MVSA| 92.7 ---========>. |VM00001| 2 > . . . . . |
+ |RSCS| .1 > . . . . . |
+ |USER10| 2.1 > . . . . . |
```

The MCPU display takes one cycle to initialize and shows:

**USER**

CPU usage for all users over nn percent. Users are listed by threshold percentage for SYSTEM usage, which always appears first.

**TYPE**

CPU usage for interactive users in queue 1 (Q1), non-interactive users in queue 2 (Q2), and performance groups (DOS, MVS, and VM00001 in this example) over nn percent. After Q1 and Q2, performance groups appear in alphabetical order.

**SYSTEM**

CPU usage for all CPUs over nn percent. Total CPU time is broken down into virtual time and CP time.
To display users and performance groups in order by CPU utilization, precede MCPU\textit{nn} with a C in column 1, as in this example:

<table>
<thead>
<tr>
<th>CMCPU00</th>
<th>USER</th>
<th>CPU% 0.2.4.6.8.0</th>
<th>TYPE</th>
<th>CPU% 0.2.4.6.8.0</th>
<th>SYSTEM</th>
<th>CPU% 0.2.4.6.8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>MVS</td>
<td>87.8 &lt;=---&gt;---&gt;-</td>
<td>Q1</td>
<td>7 &gt; . . . .</td>
<td>CPU2: 99 --</td>
<td>---&gt;---&gt;</td>
</tr>
<tr>
<td>+</td>
<td>USER04</td>
<td>7.3 &gt; . . . .</td>
<td>Q2</td>
<td>93 --</td>
<td>---&gt;---&gt;</td>
<td>VIRT: 88 --</td>
</tr>
<tr>
<td>+</td>
<td>USER02</td>
<td>1.6 &gt; . . . .</td>
<td>MVS</td>
<td>88 --</td>
<td>---&gt;---&gt;</td>
<td>CP: 11 -&gt; . . . . .</td>
</tr>
<tr>
<td>+</td>
<td>DOS001</td>
<td>1.5 &gt; . . . .</td>
<td>VM00001</td>
<td>9 &gt; . . . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>SYSTEM</td>
<td>.4 &gt; . . . .</td>
<td>DOS</td>
<td>2 &gt; . . . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>DOS002</td>
<td>.3 &gt; . . . .</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>RSCS</td>
<td>.1 &gt; . . . .</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To display the top \textit{nn} users based on CPU utilization, precede MCPU\textit{nn} with a T in column 1. The variable \textit{nn} is the number of users, as in this example for the top three users:

<table>
<thead>
<tr>
<th>TMCPU03</th>
<th>USER</th>
<th>CPU% 0.2.4.6.8.0</th>
<th>TYPE</th>
<th>CPU% 0.2.4.6.8.0</th>
<th>SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>MVS</td>
<td>87.8 &lt;=---&gt;---&gt;-</td>
<td>Q1</td>
<td>7 &gt; . . . .</td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>USER04</td>
<td>7.3 &gt; . . . .</td>
<td>Q2</td>
<td>93 --</td>
<td>---&gt;---&gt;</td>
</tr>
<tr>
<td>+</td>
<td>USER02</td>
<td>1.6 &gt; . . . .</td>
<td>MVS</td>
<td>88 --</td>
<td>---&gt;---&gt;</td>
</tr>
</tbody>
</table>

**VCPU - User CPU Utilization**

The VCPU major command selects all users with CPU usage over \textit{nn} percent. The following VCPU minors display CPU and paging information for users selected by VCPU.

- **term**: Displays terminal address of user’s virtual console.
- **que**: Displays dispatching queue that user is on.
- **wait**: Displays the reason that a user is waiting.
- **page**: Displays a paging count (.R displays a rate per second).
- **vioc**: Displays virtual I/O count (.R displays a rate per second).
- **ttim**: Displays total CPU time (.R displays a rate per second).

Enter VCPU05 with the listed minor commands directly below it to see an example of user CPU information.

<table>
<thead>
<tr>
<th>VCPU05</th>
<th>PAYR01</th>
<th>PAYR02</th>
</tr>
</thead>
<tbody>
<tr>
<td>term</td>
<td>684</td>
<td>683</td>
</tr>
<tr>
<td>que</td>
<td>Q1</td>
<td>Q2</td>
</tr>
<tr>
<td>wait</td>
<td>EX: 8321 I/O: 191</td>
<td></td>
</tr>
<tr>
<td>page</td>
<td>R</td>
<td>2.1</td>
</tr>
<tr>
<td>vioc</td>
<td>R</td>
<td>62.1</td>
</tr>
<tr>
<td>ttim</td>
<td>R</td>
<td>7.3</td>
</tr>
</tbody>
</table>

**VSEL - Selecting Users Based on Multiple Criteria**

VSEL, one of the VM user analysis major commands, selects users to be displayed based on multiple criteria. You can use VSEL to invoke a string of user selection major commands with
conditions. Use the Boolean operators & for and, | for or, and ~ for not, to create logical statements for selections. For commands that require string input under VSEL, such as VUSR or VPAGN, enter the strings in parentheses like this: VUSR(PAYR*).

The following examples show how to specify multiple characteristics using VSEL with user selection major commands.

**VSEL ~ VCMS & VCPU05**

Selects all non-CMS users that are using over 5% CPU.

**VSEL VDSC & VPAG05**

Selects all users that are disconnected and are paging greater than 5 pages per second.

**VSEL VUSR(PAYR*) & VPAG05**

Selects all users that start with PAYR and are paging greater than 5 pages per second.

Refer to the OMEGAMON for VM Reference Manual for other user information commands to use with VSEL.

**AENV - Graphic User Environmental Display**

The AENV minor command graphically displays CPU usage information for a specific user. The user displayed is the first user on the last line of users selected by the preceding major command.

To select a specific user, enter the user’s name in the argument field of VUSR. Then enter the AENV minor command directly below VUSR in column 2, as in the following example. The AENV display takes one cycle to initialize.

For the user selected, AENV displays the following information:

- **Cycle** Number of cycles since command was first entered. If 20 or less, this figure also indicates the number of cycles in the long portion of the display. After 20 cycles, the long portion always includes the last 20 cycles.
- **Short** Short values represent figures for the last OMEGAMON cycle.
As you can see, the AENV command shows a summary of most user resources. A plus sign (+) or a minus sign (-) on the far right indicates whether the value is increasing or decreasing over cycles.

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Long</strong></td>
<td>Long values represent the average values over the last 20 OMEGAMON cycles.</td>
</tr>
<tr>
<td></td>
<td>(Until OMEGAMON executes the command for 20 cycles, the values represent</td>
</tr>
<tr>
<td></td>
<td>an average over the number of cycles since the command was first entered.)</td>
</tr>
<tr>
<td><strong>VIRT Time(%)</strong></td>
<td>User’s virtual processing time.</td>
</tr>
<tr>
<td><strong>CP Time(%)</strong></td>
<td>Supervisor processing time (main and attached).</td>
</tr>
<tr>
<td><strong>I/Os/sec</strong></td>
<td>Virtual I/O count for non-SPOOLed devices.</td>
</tr>
<tr>
<td><strong>Page-ins/s</strong></td>
<td>Number of page reads per second.</td>
</tr>
<tr>
<td><strong>Page-outs/s</strong></td>
<td>Number of page writes per second.</td>
</tr>
<tr>
<td><strong>Proj WSS</strong></td>
<td>User’s projected working set size.</td>
</tr>
<tr>
<td><strong>Res pages</strong></td>
<td>User’s number of resident pages.</td>
</tr>
<tr>
<td><strong>Pging/CPU-s</strong></td>
<td>Calculated ratio of paging requests to supervisor processing time for this user.</td>
</tr>
</tbody>
</table>

As you can see, the AENV command shows a summary of most user resources. A plus sign (+) or a minus sign (-) on the far right indicates whether the value is increasing or decreasing over cycles.
Real Storage Analysis

The following diagram shows the flow of commands you would use to perform real storage analysis.

For this real storage analysis example, assume that the threshold for either the EXRT or DPAA exception was exceeded, indicating a possible real storage problem. The SMAP minor of SYS shows how CP has allocated the real storage. The FMAP minor of SYS gives an indication of CP free storage utilization. The MRES command enables you to see which users occupy real storage. The VCOR command gives a picture of which virtual pages are being used by each virtual machine.

The next section starts a detailed discussion of these commands.
SMAP - Real Storage Map

The SMAP minor command of SYS displays a map of the physical layout of main storage. Enter the SYS major command with the SMAP minor command immediately below in column 2.

<table>
<thead>
<tr>
<th>Major Area</th>
<th>Size</th>
<th>Minor Area</th>
<th>Size</th>
<th>Address Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Area</td>
<td>7168K</td>
<td>Dynam Paging</td>
<td>7168K</td>
<td>0000000140000000-0000001406FFFFFFF</td>
</tr>
<tr>
<td>Hardware</td>
<td>5056M</td>
<td>Offline Stg</td>
<td>5056M</td>
<td>0000000400000000-000000013FFFFFFF</td>
</tr>
<tr>
<td>Fixed Stg</td>
<td>20M</td>
<td>CP Frame Tbl</td>
<td>20M</td>
<td>000000002BF90000-000000013FFFFFFF</td>
</tr>
<tr>
<td>Dynamic Area</td>
<td>804M</td>
<td>Dynam Paging</td>
<td>36K</td>
<td>000000002BF0000-000000002BF9FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trace Tbl 00</td>
<td>400K</td>
<td>000000002B8C000-000000002B8FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dynam Paging</td>
<td>368M</td>
<td>00000002B30000-00000002B8BFFFFF</td>
</tr>
<tr>
<td>Fixed Stg</td>
<td>8192K</td>
<td>Chan Measure</td>
<td>8192K</td>
<td>000000002B2E000-00000002B2FFFFF</td>
</tr>
<tr>
<td>Dynamic Area</td>
<td>33M</td>
<td>Dynam Paging</td>
<td>3548K</td>
<td>0000000027B7000-000000002B2FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trace Tbl 03</td>
<td>300K</td>
<td>000000000276C000-00000000027B6FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dynam Paging</td>
<td>44K</td>
<td>0000000002761000-000000000276BFFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trace Tbl 02</td>
<td>300K</td>
<td>0000000002716000-0000000002760FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dynam Paging</td>
<td>44K</td>
<td>00000000027B0000-0000000002715FFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trace Tbl 01</td>
<td>300K</td>
<td>0000000002760000-000000000270AFFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dynam Paging</td>
<td>26M</td>
<td>00000000000A50000-00000000026BFFFFF</td>
</tr>
<tr>
<td>Fixed Stg</td>
<td>2708K</td>
<td>CP Nucleus</td>
<td>2708K</td>
<td>0000000008000000-000000000A4FFFFF</td>
</tr>
<tr>
<td>V=R Area</td>
<td>8192K</td>
<td>V=R Free Stg</td>
<td>1024K</td>
<td>0000000007000000-0000000007FFFFFFFF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V=R User Area</td>
<td>7168K</td>
<td>0000000000000000-0000000006FFFFF</td>
</tr>
<tr>
<td>Total online</td>
<td>71M</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FMAP - Free Storage Usage Map

The FMAP minor of SYS shows a map of free storage usage. The field descriptions follow the FMAP display.

Area A: Shows allocation and utilization by the following:

TOTAL      Specifies the total size of free storage.
RESERVED   Specifies pages reserved for use when CP free storage has been fully allocated. CP keeps a back pocket of 2 pages per processor.
SYSTEM     Specifies free storage allocated for CP’s use (not charged to a specific user).
USER       Specifies free storage allocated on behalf of a user.
V=R        Specifies the free storage area used exclusively for storage requests made on behalf of the V=R guest.
SUBPOOLS   Specifies free storage pages that are broken into different sized cells for short-term/high-speed allocation.

Area B: Shows a detailed view of the subpool area. For each size subpool, three counts are displayed.

ALOC       The number of subpool blocks allocated.
USED       The number of subpool blocks in use.
RQST       The total number of requests.
MRES - Real Storage Allocated to Users

The MRES command shows the percent of Dynamic Paging Area (DPA) allocated to each user and performance group. A summary of the system utilization is also displayed.

<table>
<thead>
<tr>
<th>MRES</th>
<th>USER DPA UTIL 0_2_4_6_8_0</th>
<th>PERF GROUP 0_2_4_6_8_0</th>
<th>SYSTEM 0_2_4_6_8_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>VTAM 45.2% --&gt; . . .</td>
<td>SERVICE 53% --&gt; . . .</td>
<td>Total DPA Pages 564</td>
</tr>
<tr>
<td>+</td>
<td>TTD12 8.6% --&gt; . . .</td>
<td>OPERATRS 1% &gt; . . .</td>
<td>SHR 7% &gt; . . .</td>
</tr>
<tr>
<td>+</td>
<td>RSCS 6.2% --&gt; . . .</td>
<td>RSRV 21% --&gt; . . .</td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>PVM 3.1% &gt; . . .</td>
<td>VMPG 35% == &gt; . . .</td>
<td>FREE 1% &gt; . . .</td>
</tr>
<tr>
<td>+</td>
<td>OPERATOR .0% &gt; . . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>TDSFO6 .0% &gt; . . .</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Next, you can select an individual user to evaluate storage utilization.

VCOR - Map of Virtual Storage

The VCOR command shows a map of virtual storage for a user. Four groups of 16 pages each are displayed per line. Page numbers for each group appear along the left-hand margin. A minus sign (-) preceding a group indicates it is shared. If a group is entirely blank, it is an indication that it does not exist. By entering the command twice in succession, you can see the differences between two OMEGAMON cycles. Here is an example:

```
VCOR  TDDV02  RESS=170, FWSS=166, PGDR=0, POWR=0
+  00-03F -III...III...III...III...III...III...III...III...III...III...III...III...III...III
+  04-07F I....I....I....I....I....I....I....I....I....I....I....I....I....I....I....I....I
+  08-0B F ..........................                       ..........................
+  0C-0F F ..........................                       ..........................

VCOR  TDDV02  RESS=171, FWSS=172, PGDR=0, POWR=0
+  00-03F -XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX...XXX
+  04-07F XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
+  08-0B F ..........................                       ..........................
+  0C-0F F ..........................                       ..........................
```

The characters in the display above indicate the following:

<table>
<thead>
<tr>
<th>Character</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>A page that was out on previous display but is now in.</td>
</tr>
<tr>
<td>O</td>
<td>A page that was in but is now out.</td>
</tr>
<tr>
<td>S</td>
<td>A page that is shared</td>
</tr>
<tr>
<td>X</td>
<td>A page that was in and is still in.</td>
</tr>
<tr>
<td>.</td>
<td>A page that was out and is still out.</td>
</tr>
<tr>
<td>/</td>
<td>A page that is not addressable</td>
</tr>
<tr>
<td>b</td>
<td>A page that has never been referenced or released</td>
</tr>
</tbody>
</table>
In addition, the following conventions are used:

- A hyphen (-) preceding a segment indicates that it is shared.
- If a segment is entirely blank, it has either never been referenced or has been released by the detach diagnose.
- A capital letter indicates real storage pages above the 16M line, and lower case indicates that the pages are below the line.

**Note:** VCOR supports guests up to 4 TB (4096 GB).
Paging Analysis

The following diagram shows the flow of commands you would use to perform paging analysis.

![Diagram showing the flow of commands for paging analysis]

Use the MPAG major command to display paging rates. If you want to see all users that are above a specified paging rate, use the VPAG command. PAGU provides system paging information. SOWN shows SYSOWNed devices that have PAGE or TEMP space allocated. The ALOC command shows an allocation extents map for SYSOWNed DASD.

The next section starts a discussion of these commands.

**MPAG - System Paging Information**

The MPAG command displays user, group, and system paging statistics. The MPAG display is similar in format to the MCPU display. MPAGnn displays paging statistics over a certain rate, which you must specify as a numeric argument. For example, MPAG00 displays paging for users and groups over 00 pages per second, which in this case represents all paging.

<table>
<thead>
<tr>
<th>MPAG05</th>
<th>USER  PG/S 0_2_4_6_8</th>
<th>GROUP  PG/S 0_2_4_6_8</th>
<th>SYSTEM  PG/S 0_2_4_6_8</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>MVSD 34.8 --&gt; . . .</td>
<td>SERVICE 16.3 --&gt; . . .</td>
<td>PAGINS 36.9 --&gt; . . .</td>
</tr>
<tr>
<td>+</td>
<td>VMXDA 15.4 --&gt; . . .</td>
<td>MVS 34.8 --&gt; . . .</td>
<td>PAGOUT 29.6 --&gt; . . .</td>
</tr>
<tr>
<td>+</td>
<td>EREP 7.7 &gt; . . .</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+</td>
<td>TOTAL: 66.9 --&gt; . .</td>
<td></td>
<td>TOTAL: 66.9 --&gt; . . .</td>
</tr>
</tbody>
</table>

The MPAG command shows:

**USER**

Paging by user. The display includes all users whose paging rate exceeds the threshold specified by \( nn \). It also shows the TOTAL page-in rate for the system.

**GROUP**

Paging by performance group. You can assign a group of virtual machines to a specific performance group using the PGNAME keyword in the OVUSER DATA file. This is described in the *OMEGAMON and EPILOG for VM Installation and Customization Guide*. For each group of users, the MPAG command calculates the paging rate, as in the USER column, except that it is on a group basis. The MPAGnn command displays and plots any group that exceeds the threshold specified.

**SYSTEM**

Total paging for the entire system. This zone shows page-ins and page-outs.
VPAG - User Paging Information

The VPAGnn major command selects all users above a particular paging rate (nn pages per second). The PAGE minor command, when used with the .R operand, shows a per-second paging rate. Enter VPAG10 with PAGE.R directly below it in column 2 to display a paging rate for users over 10 pages per second.

```
VPAG10 USER15
page.R   13.2
```

SOWN - Device Allocation Statistics

The SOWN command selects all SYSOWNed devices that have either PAGE or TEMP space allocated on them.

ALOC - Device Allocation Statistics

The ALOC command shows an allocation extents map for each SYSOWNed DASD. The next figure shows a typical display.
I/O Analysis

The following diagram shows the flow of commands you would use to perform I/O analysis.

A number of exceptions, such as IOWT, DEVQ, CHNQ, DVRT, and VMIO indicate potential problems in the I/O subsystem.

Use the MIOR command to see at a glance which users and devices are sustaining high I/O rates.

The DISK, DSKQ, and DPLT commands focus on DASD activity. Use the VIOS command to select individual users for detailed analysis.

The next section starts a discussion of these commands.

**MIOR - Overall I/O Activity Analysis**

In the following example, the MIOR command pinpoints the five users with the highest I/O rate. The same command also shows the five devices with the highest I/O activity rate. The T in column 1 and the 05 that starts in column 6 modify the command to give the top 5 users and devices.
DISK - Device Analysis

The DISK major and its associated minor commands show information about online disks. Use the DISK major command to select all online disks, and then use its minor commands to display specific types of information. For example:

<table>
<thead>
<tr>
<th>DISK</th>
<th>CICS15</th>
<th>WORK22</th>
<th>CICS16</th>
<th>CICS62</th>
<th>WORK21</th>
<th>IMS120</th>
<th>IMS130</th>
<th>CICS61</th>
</tr>
</thead>
<tbody>
<tr>
<td>dadr</td>
<td>750</td>
<td>751</td>
<td>752</td>
<td>753</td>
<td>754</td>
<td>755</td>
<td>756</td>
<td>757</td>
</tr>
<tr>
<td>datt</td>
<td>750</td>
<td>751</td>
<td>752</td>
<td>753</td>
<td>754</td>
<td>755</td>
<td>756</td>
<td>757</td>
</tr>
<tr>
<td>dcyl</td>
<td>410</td>
<td>673</td>
<td>488</td>
<td>652</td>
<td>361</td>
<td>853</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dio</td>
<td>93</td>
<td>142981</td>
<td>123</td>
<td>108496</td>
<td>259825</td>
<td>52787</td>
<td>18627</td>
<td>78306</td>
</tr>
<tr>
<td>dioq</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
</tbody>
</table>

dadr   Shows unit address of the device.
datt   Shows virtual address of device (if attached or dedicated).
dcyl   Shows current seek cylinder number (for FBA devices, this is a converted cylinder).
dio    Shows I/O count for the drive (.R shows a rate).
dioq   Shows depth of the I/O queue on the drive.

DPLT - Device Utilization

The device plot DPLT disk information minor command inspects the processes that occur between OMEGAMON cycles. For any given major device command, DPLT takes 50 samples at an interval of nn milliseconds. Changes in device status, user, and cylinder are indicated with each sample. DPLT plots only the activity on the first device appearing on the last line displayed for the preceding major command.

If DPLT is used as a minor command of DSKQ, then the sampling is only performed when there is a need, such as when a disk has an I/O queue depth of nn or more and is therefore chosen by the DSKQ\text{nn} command. The following figure is an example.

FIGURE 14. DPLT Minor Command
The fields in the display above indicate the following:

**DBSY**  B Indicates device busy.
**E**  Indicates error recovery is active.
**O**  Indicates device is offline.

**IO**  Number of I/Os completed between samples. (1—9, or A—Z for 10—35.)

**IOQ**  Depth of I/O queue (+ indicates more than 9).

**USER**  Current user of device. (A user ID can be truncated if it does not completely fit at the right end of the screen.)

**CYL**  Cylinder being accessed (> indicates the point a new cylinder is accessed. The number appears to the right of the >.)

**NRDY**  N indicates device not ready.

**RESV**  R indicates device RESERVEd.

The example in Figure 14 on page 95 shows that the DSKQ01 command selected a disk with volser of CMS005 because it had an I/O queue depth of one or more. The minor command DPLT with an argument of 10 plotted the activity on the selected device at 10 millisecond intervals (50 samples are always taken). The interval actually used appears on the right in milliseconds. In this example, the actual interval was 13 milliseconds. The total I/Os observed during the sample also appears on the right (62 in this example). The resulting plot shows that a user (TDDV03A) and the SYSTEM are competing for the disk arm at cylinder addresses 65 through 442. This explains the I/O queue on the selected device.

**VIOS - User I/O**

Use the VIOS command to select users with I/O activity. All of the user selection minor commands are available for detailed analysis. The VDEV, LVIO, VIOC, and AENV minor commands may be of special interest, since they all give I/O-related information. Following is an example:

```
+ VDEV
+ 009 CONS 3210  7 00C RDR 3505  0 00D FUN 3525  0
+ 00E PRT 3211  0 080 PRT 1403  0 081 PRT 1403  0
+ 190 DASD 3380  3 191 DASD 3380  6 193 DASD 3380  8
+ 19D DASD 3380  0 19E DASD 3380  3 300 DASD 3380  9
+ 319 DASD 3380  29 31A DASD 3380  6 5FF GRAP 3278 D 5065
```
Analysis of the DOS/VSE Virtual Machine

MDOS displays status and resource information about a DOS/VSE virtual machine and the DOS/VSE partitions. Here is an example of the display:

```
MDOS DOSPROD
+-----VM Information ---------------------------------------------
+  Cpu  Virt    CP    I/O  Paging   Size   Res   WSS  Pri  State
+  2.3   1.5    .7     .0     .0   16m   118  118  64  Idle
+-----DOS Information -( VM 5750-AAT )---------------------------
+  Part Status  Size  Job name Phase     Cpu  VM%  Dos%  Start Time
+  BG  ACTIVE   0  NO NAME               .000   .0   .0 12/16/01 09:43:25
+  FB  INACTIVE 0  NO NAME               .000   .0   .0          09:37:50
+  FA  ACTIVE   204K VTAM312T ISTINCVT   .312   .2  16.5          09:38:17
+  F5  ACTIVE   0  NO NAME               .000   .0   .0          09:37:50
+  F4  ACTIVE   0  NO NAME               .000   .0   .0          09:37:50
+  F3  ACTIVE   204K PAUSEF3              .000   .0   .0          09:39:31
+  F2  * ACTIVE 1M CICS17SR DFHSIP        1.528  1.2  82.3 12/16/01 09:46:33
+  F1  ACTIVE   65K PWRVS312 IPWPOWER     .000   .1  1.2          09:38:18
```

The top part of the display gives information on the DOS/VSE virtual machine CPU usage, I/O, paging, and storage utilization. The bottom part of the display gives information on DOS/VSE partition activity, such as virtual storage size, the name of the job running, the current program phase running, and CPU usage.

**Note:** MDOS supports guests up to 2 GB.
Symbols

# in column 1 39
+ with AENV minor 86
.BELON immediate 61
.CLR immediate 42
.D argument
.DEF immediate
  .HO 50
  .OFF 50
  .ON 50
.EXM immediate 34
.FGO immediate 47, 54
.i immediate 40
.ILC 34
.INT immediate 36
.LOG immediate 47
.MIN 34
.MJ 34
.MJC 34
.MJI 34
.NXE immediate
  .command 62
.PFK immediate 44, 66
.PLM immediate 48
.PRT immediate 48
.R (rate) argument 38, 84, 93, 95
.R immediate 40
.RC immediate 40
.REP immediate 46
.SCC immediate 43
.SGO immediate 47, 54
.SYN immediate 37
.VAR immediate 51
.ZAP immediate 20
/ in column 1 34
/BELLON INFO-line 61
/BOTTOM 39
/DEF INFO-line
  .HOLD 50
  .OFF 50
  .ON 50
/DOWN 39
/F INFO-line 47
/INT INFO-line 36
/L INFO-line 46
/LOG INFO-line 47
/P INFO-line 48
/P PRINT INFO-line 48
/SAVE INFO-line 48
/SAVE INFO-line 50
/UP 39
/ZOOM INFO-line
    See zooming feature
/? help text 34
/? in column 1 34
/@ZPFK2 44
/@ZPFKDEF 44
/@ZPFKNEW 44

Numerics

3270 terminal 30
3279 color terminal family 41
327x non-color terminal 41
327x terminal 43
3290
  non-color terminal 41
terminal 43

A

Adobe portable document format 11
AENV - Graphic User Environmental Display 85
AENV minor 82, 85, 96
alarm 61
ALOC - Device Allocation Statistics 93
ALOC major 92, 93
analysis of the DOS/VSE virtual machine 97
analysis, CPU 82
analysis, exception 59
analysis, real storage 87
argument 32
arguments, command 38
arguments, rate and difference 38
ASF 31, 62, 66
assigning levels and colors dynamically 64
assigning levels and colors in OVUSER 63
automatic invoking of screen spaces 54
automatic screen facility
  See ASF
automatic update 48
  interval 36
available parameters 62
DASDTG parameter 68
data collecting and reporting for bottleneck analysis 74
data collection 72
  automatic 72
  for Bottleneck Analysis 75
  for impact analysis 73
data reporting for impact analysis 73
date in INFO-line 30
DATG minor 68
datt minor 95
dcyl minor 95
dedicated mode 20
  secondary console support with 43
default, color 42
defer feature 36
definition mode 50
definitions, menu PF keys 25
DEFVMTG parameter 68
degradation, definition 71
delete
  lines 41
  screen space 54
deleting and inserting lines 40
DELT immediate 54
Delta argument
  See difference argument
DELV minor 75
designing screen spaces 51
DEST parameter 46
device
  allocation statistics 93
  analysis 95
  busy 96
  display plot of activity 95
  not ready 96
  offline 96
  reserved 96
  utilization 95
device allocation statistics 93
device analysis 95
device utilization 95
DEX major 73, 75
DEX=AUTO 72
difference and rate arguments 38
difference argument 38
dio minor 95
diqq minor 95
DISK - Device Analysis 95
DISK major 94, 95
display area 29
displays and commands color 43
DOS/VSE partition information (MDOS) 97

DOS/VSE virtual machine analysis 97
down/up scrolling 39
DPA 90
DPLT - Device Utilization 95
DPLT minor 94, 95
DSKQ major 94, 95
dynamic assignment of levels and colors 64
dynamic paging area
  See DPA

END minor 75
environmental display 96
elements, impact and bottleneck analysis 76
elements, usage 76
elements, zoom 56
exception
  available parameters
  box attributes 62
  checking parameter setting 61
  highlight attributes 62
  logging facility, XLF 66
  recommendation screens 65
  severity level 62
  state 62
  system 67
  testing 64
  threshold 62, 64
    changing 61
    RSCS 68
    setting 61
    VM 67
  threshold group for DASD 68
types 67
exception analysis 59–69, 82
  by group 67
  frequency parameter 62
  last and worst occurrence 69
  limit execution 62
  operation 61
  starting 61
  summary by group 69
exception analysis summary commands 69
exception groups 67
  hardware (HD) 67
  I/O related (I/O) 67
  operational (OP) 67
  paging (PG) 67
  real storage (RS) 67
  switches 67
exception investigation 66
exception message
responding  64
routing  62
exception recommendations  25
exception recommendations viewing  65
exceptions, system and VM  67
execution states  74
exit confirmation screen  26
EXNCYC parameter
  keyword  62
EXSY immediate  61
extended color support  41, 42
extended input area  32

F
feature, command synonym  37
feature, defer  36
feature, hold  37
feature, zoom  56
field
  argument  32
  command name  32
  label  32
  output display area  32
FILEDEF, CMS command  46
first screen space (FIRSTSS)  45
first step—exception analysis  82
FIRSTSS  45
FMAP - Free Storage Usage Map  89
FMAP minor  87, 89
FOLD parameter  46
FORM parameter  46
format of commands  32
format of screens  29
free storage
  allocated for CP  89
  allocated to user  89
  pages reserved  89
  subpools  89
  V=R  89
free storage usage map  89

G
graphic display of CPU utilization  83
graphic plot analysis  73
graphic user environmental display  85
group creation, threshold  68
GROUP= keyword  73
groups, exception  67
groups, exception analysis  67

H
H.MIN  34

HDSW minor  67
help  34
  command  34
  extended  34
help for INFO-line  34
help for major, minor, and immediate commands  34
help screens  24
highlight attributes
  exception  62
highlighting  41, 43
highlighting or color support  41
hold
  argument  48
  feature  37
hold feature  37
how bottleneck analysis is used  74
hyphen, with AENV minor  86

I
I/O  95
  activity  96
  analysis  94
  queue depth  95, 96
  rate, displaying  94
I/O analysis  94
IA=AUTO  72
IANL minor  73
immediate commands  33, 40
  list  34
immediate, major, and minor, command help  34
impact analysis  71–79
  commands  73
Impact And Bottleneck Analysis  71
INFO-line  29
INFO-line command help  34
INFO-line commands  29, 33
  format  29
  list  34
inserting and deleting lines  40
intensity, assigning  63
interval  36
introduction to exception analysis  60
introduction to the product  18
investigating the condition  66
invoking screen spaces automatically  54
IOSW minor  67
IPRO minor  73, 75
IUCV interface  72

K
keys, PF  25, 44
OVUSER, assigning levels and colors 63
OVUSERcc 68
DATA file 61
limiting output to log 48
LINECT parameter 46
REPORT parameter 46

assigning screen spaces to 44
cancelling definition 44
defining 44
using

Q

QUE minor 84
question mark in column 1 34

R

rate (.R) argument 84, 93, 95
rate and difference arguments 38
real storage
amount allocated 90
analysis 87
map 88
real storage allocated to users 90
real storage analysis 87
real storage map 88
renaming screen space 53
RENM immediate 53
arguments 53
repeat lines 40
REPORT file 30, 46
report file, page limit 48
reporter
Bottleneck Analysis 74
in impact analysis 73
reporting and collecting for bottleneck analysis 74
reporting, data for impact analysis 73
resident pages 86
resource contention 73
Resource Impact Profile 74, 75
sample 77
response time 75
routing
exception message 62
type 62
RSCSTG parameter 68
RSSW minor 67

S

sample counts of collector data 72
sampling overview 72
screen
deleting lines 41
format 29
freezing 48
inserting lines 40
manipulation 39
partial printing 48
printing 48
screen clear command 39
screen control 39
screen format 29
screen space 31, 36, 49
/SAVE INFO-line 50
branching between 54
creating 50
definition of 29
deleting 54
designing 51
fetch 54
INFO-line commands in 51
invoking 54
listing specific 53
loading 52
logging 46, 48
LSCR immediate 52
modifying 52
names 50
names of 30
predefined 66
printing 48
renaming 53
RENM immediate 53
save 50
using variables in 51
screen space automatic invocation 54
screen space creation 50
screen space listing 53
screen space loading 52
screen space modification 52
screen space, first 45
screen, main body 31
screens, help 24
SCRN immediate 36, 53
scrolling 39
amount of 31
horizontal 40
up/down 39
secondary console control 43
selected line printing 48
selecting users based on multiple criteria 84
SELV minor 75
separator line 51
set a new threshold 64
SETLVLS parameter 63
setting exception parameters 61
setting severity levels and colors 63
setting the color mode 42
setting the color of commands and displays 43
severity level
assigning 63
changing dynamically 64
exception 62
of exception message 63
single screen printing 48
SIO rate 94
slash in column 1 34
SMAP - Real Storage Map 88
SMAP minor 87, 88
SOWN - Device Allocation Statistics 93
SOWN major 92, 93
space, screen 49
starting OMEGAMON 22
statistics, displaying 92
stopping OMEGAMON 26
storage analysis 87
storage map, real 88
storage map, virtual 90
storage usage map, free 89
subpool
area 89
summary commands, exception analysis 69
summary plot display 73
support of VM systems by Version 630 15
support, color or highlighting 41
swap-in
wait 74
SYNONYM keyword 37
synonym, command 37
synonyms, defining 37
SYS major 87, 88
system
exceptions 67
ID 30
system code 15
system exceptions and VM exceptions 67
system paging information 92
systems supported by Version 630 15
system-wide CPU utilization 83

TERM minor 84
terminals
non-3270 20
setting up multiple 43
TEST state for exceptions 62, 64
test, exceptions 64
threshold
exception 60
groups 68
setting 61, 64
threshold group creation 68
threshold groups, displaying 68
zoom examples  56
zooming feature  56, 66
/ZOOM INFO-line  56
cursor pointing  56