

MAINVIEW® for IMS Online Customization Guide

Version 3.3.10

March 2002



Copyright © 1998–2002 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. IBM and DB2 are registered trademarks of International Business Machines Corp. All other registered trademarks or trademarks belong to their respective companies.

THE USE AND CONTENTS OF THIS DOCUMENTATION ARE GOVERNED BY THE SOFTWARE LICENSE AGREEMENT ENCLOSED AT THE BACK OF THIS DOCUMENTATION.

Restricted Rights Legend

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED–RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b), or DFARS 227.7202, as amended from time to time. Send any contract notices to Contractor/Manufacturer:

BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827
USA

Contacting BMC Software

You can access the BMC Software Web site at <http://www.bmc.com>. From this Web site, you can obtain general information about the company, its products, special events, and career opportunities. For a complete list of all BMC Software offices and locations, go to <http://www.bmc.com/corporate/offices.html>.

USA and Canada

Address BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827

Telephone 713 918 8800 or
800 841 2031

Fax 713 918 8000

Outside USA and Canada

Telephone (01) 713 918 8800

Fax (01) 713 918 8000

Customer Support

You can obtain technical support by using the Support page on the BMC Software Web site or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software,” below.

Support Web Site

You can obtain technical support from BMC Software 24 hours a day, seven days a week by accessing the technical support Web site at <http://www.bmc.com/support.html>. From this site, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support via Telephone or E-mail

In the USA and Canada, if you need technical support and do not have access to the Web, call 800 537 1813. Outside the USA and Canada, please contact your local support center for assistance. To find telephone and e-mail contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at www.bmc.com/support.html.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that a technical support analyst can begin working on your problem immediately:

- product information
 - product name
 - product version (release number)
 - license number and password (trial or permanent)
- operating-system and environment information
 - machine type
 - operating system type, version, and service pack or program temporary fix (PTF)
 - system hardware configuration
 - serial numbers
 - related software (database, application, and communication) including type, version, and service pack or PTF
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
 - product error messages
 - messages from the operating system, such as `file system full`
 - messages from related software

Contents

How to Use This Book	xi
----------------------------	----

Part 1. Migration Considerations..... 1

Chapter 1. Migrating to Version 3.3.10 from Version 3.3.00	3
Product Authorization	3
Shared Message Queue Statistics	4
MSC Clock Synchronization	4
New Event Collector Parameter MSCCLOCK	4
New TERM Keyword Operands SYNCLOCK and MSCCLOCK	5
UBBPARM Customization Capabilities	5
Product Initialization Messages	5
Chapter 2. Migrating to Version 3.3.10 from Version 3.2	7
Product Authorization	7
Product Packaging and Keys	8
Detail Trace	8
Shared Message Queue Statistics	9
UBBPARM Customization Capabilities	9
MSC Clock Synchronization	10
New Event Collector Parameter MSCCLOCK	10
New TERM Keyword Operands SYNCLOCK and MSCCLOCK	10
Event Collector Option	11
Product Initialization Messages	11
Easy Menus	11
Chapter 3. Release Compatibility	13
Compatibility with BBI	13
Compatibility with Previous Release Levels of MVIMS	13
Downward Compatibility	13
Upward Compatibility	13
AO Exit	13

Part 2. Customizing MAINVIEW for IMS Functions..... 15

Chapter 4. Resource Utilization	17
CPU Usage – Event Collector	17
CPU Usage - IPSM Samplers	18
CSA Usage	18
Chapter 5. Implementing MAINVIEW Products in IMS	19
Deleting Modules from a Prior Release	20
Setting Up BBPARM Data Sets	21
Customizing BBPARM Members for an IMS System	23

Activating MAINVIEW for IMS Online	25
Enabling Product Initialization	25
Enabling Product Authorization	25
Product Keys	25
Product Authorization Table	26
Modifying the IMS Control Region JCL	27
Allocating a BBPARM Data Set	27
Giving IMS Access to the Event Collector	27
Adding Access to a Product Authorization Table	28
Modifying DLISAS Region JCL	28
Enabling AO Exit Routines	29
Chapter 6. Customizing the Trace Facility	33
Specifying Trace Defaults in IMFBEX00	34
Detail Trace Data Collection	34
Trace Display Buffer Size	34
Trace Duration	35
Trace Logging Options	35
Updating Trace Defaults	36
Specifying Detail Trace Buffers	36
Setting Up and Maintaining a Trace Directory	37
Defining and Initializing a Trace Directory Data Set	37
Identifying the Trace Directory to BBI	37
Verifying Trace Directory Entries	38
Managing Trace Log Data Sets	39
Defining a Trace Log Data Set	39
Archiving a Trace Log Data Set	39
Restoring an Archived Trace Log Data Set	39
Creating a Trace Log Data Set from the IMS Log	39
Printing a Trace Log Data Set	39
From a Batch Job	39
From an Online Application	40
Setting Up Workload History Traces	41
Setting Limits for Transaction Trace Views	42
Chapter 7. Controlling Workload Thresholds and SMQ Data Collection	43
Specifying Thresholds for IMS and IMS Sysplex Activity Views	43
Controlling Shared Message Queue Data Collection	44
CSQ Information Views with Message Age Fields	44
Non-Workflow Views with Message Count Fields	45
Workflow Views with Message Count Fields	45
Chapter 8. Using the Event Collector	47
Starting and Stopping the Event Collector	47
Specifying Event Collector Options	47
Evaluating Data Collection Options	60
Reviewing the Initialization Parameters	61
Parameter Option Sets	65
Standard Option Set	65
Full Option Set	65
Minimum Option Set	66
How the Option Sets Affect Online Service Data	67
Workload Trace (LTRAC, STRAC, and DTRAC)	67
Changing Event Collector Parameters	68

Chapter 9. Specifying Target IMS Systems	69
Chapter 10. Customizing with User Exit Routines	71
Event Collector User Exit Routines	71
Transaction Record User Exit (IMRUTRN)	71
Program Record User Exit (IMRUPGM)	71
IMRUTRN and IMRUPGM Cross-Memory Mode Considerations	72
DL/I-CALL-END User Exit (IMRUDLI)	72
Chapter 11. Service Utility Commands	73
Lock a Service (LOCK)	73
Unlock a Service (ULOCK)	73
Chapter 12. Optional Modifications to MAINVIEW for IMS	75
Modification to Support SAP	75
SMP-Applied Modifications	76
Modification to Support TELON	76
Modifications to Reconcile Response Time Monitoring Data	77
Zero Out Input Queue Time for Selected Transactions	77
Zero Out Input Queue Time for Message Switch Transactions	77
<hr/>	
Part 3. Implementing Product Security	79
Chapter 13. Security for Analyzer and Monitor Services	81
Selective Service Lists by User Group	81
Command Authorization	81
Trace Authorization	82
<hr/>	
Part 4. Appendixes	83
Appendix A. IMS Dump Analysis	85
AO Exit Dispatch	85
MAINVIEW AutoOPERATOR for IMS Routines in IMS	85
Event Collector	85
Appendix B. How Product Libraries Should Be Used	87
Appendix C. BBSAMP Data Set Members	89
Sample Members for MAINVIEW for IMS and MAINVIEW for DBCTL	89
Appendix D. Customizing Analyzer and Monitor Services	93
Modifying Service Characteristics	93
Adding a Service	93
Service Table Definition	94
Examples of Parameter Usage	99
Service Logging	99
Appendix E. Product Initialization Messages	101
Messages Changed with Version 3.3.10	101
Messages Added with Version 3.3.00	101
Messages Removed with Version 3.3.00	102
Messages Changed with Version 3.3.00	103

Part 5. Glossary and Index	105
Glossary	107
Index	119

Tables

1.	MVIMS Online Product Keys	3
2.	MVIMS Online Product Keys	7
3.	BBSAMP Jobs to Delete Old Modules	20
4.	MVIMS Online Product Keys	25
5.	IMFBEX00 Parameters for Trace Transaction Limits.	42
6.	IMFBEX00 Parameters for IMS and IMS Sysplex Activity Workload Thresholds . . .	43
7.	Event Collector Collection Options.	48
8.	Event Collector Recovery Options	59
9.	Product Libraries	87
10.	BBSAMP Data Set JCL Members.	89
11.	Service Table Parameters and Options	95
12.	Product Initialization Messages Changed with Version 3.3.10	101
13.	Product Initialization Messages Added with Version 3.3.00.	101
14.	Product Initialization Messages Removed with Version 3.3.00	102
15.	Product Initialization Messages Changed with MVIMS Online Version 3.3.00. . . .	103
16.	Product Initialization Messages Changed with MVDBC Version 3.3.00	104

How to Use This Book

This book contains procedures for customizing MAINVIEW® for IMS (MVIMS) Online. MVIMS is based on the MAINVIEW environment, which allows a single terminal session (TS) to monitor and manage multiple local or remote targets, whether OS/390 itself (sysplex and nonsysplex) or subsystems like CICS, IMS, and DB2.

For information about what's new in the current release of MVIMS Online, see the product Release Notes, which are available on the BMC Software Support Web pages.

This book can be viewed online with the Adobe Acrobat Reader. Contact your system administrator for assistance.

To install and customize MVIMS Online, follow the instructions in the

1. *MAINVIEW Installation Requirements Guide*—to load the BMC Software product libraries
2. *MAINVIEW Common Customization Guide*—to set up the operational environment of all MAINVIEW products at your site
3. *Implementing Security for MAINVIEW Products*—to secure MAINVIEW product resources with your external security manager (ESM), such as CA-ACF2, CA-TOP SECRET, or RACF
4. *MAINVIEW for IMS Online – Customization Guide* (this book)—to tailor MAINVIEW for IMS to your site's requirements
5. *MAINVIEW Administration Guide*—to maintain the MAINVIEW environment

Notes:

- The *MAINVIEW Common Customization Guide* and *MAINVIEW Administration Guide* consolidate instructions for implementing and administering the environment for all MAINVIEW products.
- You must customize the MAINVIEW operational environment as described in the *MAINVIEW Common Customization Guide* before customizing MVIMS Online.

If you used AutoCustomization as described in the *MAINVIEW Installation Requirements Guide*, you do not need to customize the MAINVIEW environment and MAINVIEW for IMS—it is already done. You can use the *MAINVIEW Common Customization Guide*, the *MAINVIEW Administration Guide*, and this book, the *MAINVIEW for IMS Online – Customization Guide*, for reference.

- If you are installing MVIMS Online or MAINVIEW® AutoOPERATOR™ for IMS, please note that there are some common steps between this guide and the *MAINVIEW AutoOPERATOR Customization Guide* (see Chapter 5, “Implementing MAINVIEW Products in IMS” on page 19).

MAINVIEW for IMS Product Library

The MAINVIEW for IMS product library contains the following documents:

MAINVIEW for IMS Online – Customization Guide

MAINVIEW for IMS Online – Analyzers Reference Manual

MAINVIEW for IMS Online – Monitors and Traces Reference Manual

MAINVIEW for IMS Online – IPISM Reference Manual

MAINVIEW for IMS Online – Release Notes

MAINVIEW for IMS Offline – Customization and Utilities Guide

MAINVIEW for IMS Offline – Performance Reporter Reference Manual

MAINVIEW for IMS Offline – Transaction Accountant Reference Manual

MAINVIEW for IMS Offline – Release Notes

How This Book Is Organized

This book is intended for the system programmer who needs to know how to modify the basic MAINVIEW product installation to include MVIMS Online functions.

This book is organized into the following parts:

- Part 1, “Migration Considerations” on page 1, contains release updates that must be considered if you have a previous release of MVIMS online products.
- Part 2, “Customizing MAINVIEW for IMS Functions” on page 15, describes how to tailor the MAINVIEW for IMS product components.
- Part 3, “Implementing Product Security” on page 79, contains instructions for allowing user access to MVIMS services.
- Part 4, “Appendixes” on page 83, provides supplemental information for MVIMS use.
- Part 5 contains the glossary and index.

Recommended Reading

Before using this book to customize MVIMS Online, you must

1. install the product as described in the *MAINVIEW Installation Requirements Guide*
2. tailor the MAINVIEW environment for the MAINVIEW for IMS functions as described in the *MAINVIEW Common Customization Guide*

Related MAINVIEW Product Libraries

Other MAINVIEW products that work with MAINVIEW for IMS include:

- MAINVIEW[®] AutoOPERATOR[™]
- MAINVIEW[®] for CICS
- MAINVIEW[®] for DB2[®]
- MAINVIEW[®] for DBCTL
- MAINVIEW[®] FOCAL POINT
- MAINVIEW[®] for OS/390
- MAINVIEW[®] VistaPoint
- MAINVIEW[®] for MQSeries

The following manuals document product-specific customization instructions:

- *MAINVIEW AutoOPERATOR Customization Guide*
- *MAINVIEW for CICS Customization Guide*
- *MAINVIEW for DB2 Customization Guide*
- *MAINVIEW for DBCTL Customization Guide*
- *MAINVIEW for OS/390 Customization Guide*

The following books document the use of general services common to MAINVIEW for IMS and to some of the products listed above:

- *MAINVIEW AutoOPERATOR Basic Automation Guide*
- *MAINVIEW AutoOPERATOR Advanced Automation Guide for CLIST EXECs*
- *MAINVIEW AutoOPERATOR Advanced Automation Guide for REXX EXECs*
- *MAINVIEW for CICS PERFORMANCE REPORTER User Guide*
- *MAINVIEW for DB2 User Guide*
- *MAINVIEW for DBCTL Analyzers, Monitors, and Traces Reference Manual*

IBM Publications

OS/390 Initialization and Tuning Guide
IMS Operations Guide

Conventions Used in This Book

The following symbols are used to define command syntax, are *not* part of the command, and should never be typed as part of the command:

- Brackets [] enclose optional parameters or keywords.
- Braces { } enclose a list of parameters; one must be chosen.
- A line | separates alternative options; one can be chosen.
- An underlined parameter is the default.

The following command syntax conventions apply:

- An ITEM IN CAPITAL LETTERS must be typed exactly as shown.
- Items in *italicized, lowercase* letters are values that you supply.
- When a command is shown in uppercase and lowercase letters, such as **HSplit**, the uppercase letters show the command abbreviation that you can use (**HS**, for example). The lowercase letters complete the entire command name. Typing the entire command name is an optional, alternative way of entering the command.
- Commands that do not have an abbreviation (**END**, for example) appear in all uppercase letters.

Note: Although MAINVIEW for IMS is often referred to as “MVIMS” in this book, the abbreviation is used for brevity only and does not represent a legal product name of BMC Software.

Part 1. Migration Considerations

This part of the book provides update considerations for migrating from a previous version of MVIMS Online.

Chapter 1. Migrating to Version 3.3.10 from Version 3.3.00	3
Product Authorization	3
Shared Message Queue Statistics	4
MSC Clock Synchronization	4
New Event Collector Parameter MSCCLOCK	4
New TERM Keyword Operands SYNCLOCK and MSCCLOCK	5
UBBPARM Customization Capabilities	5
Product Initialization Messages	5
Chapter 2. Migrating to Version 3.3.10 from Version 3.2	7
Product Authorization	7
Product Packaging and Keys	8
Detail Trace	8
Shared Message Queue Statistics	9
UBBPARM Customization Capabilities	9
MSC Clock Synchronization	10
New Event Collector Parameter MSCCLOCK	10
New TERM Keyword Operands SYNCLOCK and MSCCLOCK	10
Event Collector Option	11
Product Initialization Messages	11
Easy Menus	11
Chapter 3. Release Compatibility	13
Compatibility with BBI	13
Compatibility with Previous Release Levels of MVIMS	13
Downward Compatibility	13
Upward Compatibility	13
AO Exit	13

Chapter 1. Migrating to Version 3.3.10 from Version 3.3.00

This chapter describes the options you should consider if you are migrating to MAINVIEW for IMS (MVIMS) Online version 3.3.10 from version 3.3.00.

Product Authorization

MVIMS Online version 3.3.10 adds support for the use of product authorization tables to activate products on individual CPUs.

Version 3.3.10 continues to support the product keys method of product activation, and you can continue to use the product keys listed in Table 1.

Table 1. MVIMS Online Product Keys

Key	Product or Component
ION	MAINVIEW for IMS Online
IPX	IMSPlex System Manager
IRA	IMS Resource Analyzer
IRM	IMS Resource Monitor
IWA	IMS Workload Analyzer
IWM	IMS Workload Monitor
IAD	IMS Workload Analyzer Extension for DB2
IMD	IMS Workload Monitor Extension for DB2
ILM	IMS Resource Analyzer and Resource Monitor Extensions for IRLM

If MVIMS finds one of the product keys listed in Table 1 during product initialization and the key is valid, MVIMS will not look for a product authorization table. Therefore, if you want to use a product authorization table to activate MVIMS Online, you must remove all MVIMS Online product keys from the BBKEYS member of the BBPARM data set.

If no valid product key is found during initialization, MVIMS looks for the MVIMS Online product authorization table, IONTBL3P, in the data set pointed to by the BMCPSWD DD statement in the startup JCL for the IMS control region. If the BMCPSWD DD is not present, MVIMS looks in the STEPLIB, JOBLIB, and LNKLST data sets. If the IONTBL3P table is not located in those data sets, MVIMS looks in the data set pointed to by the BBILOAD DD. If the IONTBL3P table is found, its authorization information is used to determine whether the CPU is licensed to run MVIMS Online.

Note: If a product key brings MVIMS up in the IMS control region and a product authorization table brings MVIMS up in the product address space (PAS) *or vice versa*, MVIMS will still work properly as long as the provided authorization is consistent. In other words, you will encounter no initialization problems if you bring up one with a product key and the other with a product authorization table.

The procedure required to use a product authorization table to activate MVIMS Online is provided on page 28.

For more information about product authorization tables and the Product Authorization utility, see the *OS/390 and z/OS Installer Guide*.

Shared Message Queue Statistics

Version 3.3.10 of MVIMS Online includes enhanced support for IMS shared message queues, providing improved realtime and historical shared queues statistics in several of the IPSM workflow, CQS information, transaction, and terminal and user views. Several CQS information views have been added, including the new CQS object menu ISQMR.

When the new Shared Queues Data Server is operating, the views provide shared message queue statistics for IMS systems that use shared queues. You can use the ISQUERY parameter, in BBPARM member IMFBEX00, to turn the Shared Queues Data Server on or off. You can also use the ISQUERY parameter to control the amount of resources used to collect and display shared queues statistics.

For information about how to use the ISQUERY parameter, see “Controlling Shared Message Queue Data Collection” on page 44.

MSC Clock Synchronization

Version 3.3.10 of MVIMS Online provides support for controlling the handling of the arrival timestamp for MSC systems that originate on a remote system. With IMS 6.1 and later, you can define MSC systems that are synchronized to the local IMS system clock, and you can qualify workload monitor and trace requests to include either

- *all* transactions from systems synchronized to the clock of the local IMS
- or
- *only nonlocal* transactions from MSC systems synchronized to the clock of the local IMS

New Event Collector Parameter MSCCLOCK

A new Event Collector parameter, MSCCLOCK, is provided in BBPARM member IMFECPO0. With IMS 6.1 and later, you can use the MSCCLOCK parameter to specify whether or not MSC systems will, by default, be considered synchronized to the clock of the local IMS system. The MSC systems can be in the same time zone or a different time zone.

If an MSC transaction originates from an MSC system with a clock that is synchronized to the clock of the local IMS, the Event Collector will convert its arrival date and time (set by the originating MSC system) to local time. The converted time is then stored in the transaction arrival date and time fields of the transaction log record (X'FA').

If a transaction originates from an MSC system that is not defined as synchronized with the local IMS clock, no conversion is done and the transaction *arrival* date and time are set to the *start* date and time on the local IMS.

Additional MSCCLOCK values (*xxxx,SYNC* and *xxxx,NOSYNC*) are provided so that you can specify IDs for individual MSC systems that will be exceptions to the default you use.

For more information about how to use the MSCCLOCK Event Collector parameter, see page 57.

New TERM Keyword Operands SYNCLOCK and MSCCLOCK

The MSCCLOCK Event Collector parameter works in conjunction with the new monitor TERM keyword operands SYNCLOCK and MSCCLOCK, which are used to qualify workload monitor and trace requests.

TERM=SYNCLOCK defines a workload monitor or trace request to include *all* MSC transactions from systems that are synchronized to the clock of the local IMS (including all MSC transactions that originated on the local IMS). TERM=MSCCLOCK defines the request to include *only nonlocal* transactions from MSC systems synchronized that are to the clock of the local IMS.

UBBPARM Customization Capabilities

You can now use the UBBPARM data set to include unique IMFSYS00 and IMFECP00 parameter members for individual IMS systems, which eliminates the need to create and allocate an *ibbparm* data set for each IMS system. The IMS-specific settings are defined in members that must have the following name formats:

<i>imsidSYS</i>	for IMS-specific IMFSYS00 system parameters
<i>imsidECP</i>	for IMS-specific IMFECP00 Event Collector parameters

The name of each IMS-specific parameter member must begin with the identification code for the IMS system.

For more information about the IMS-specific UBBPARM members, see “Setting Up UBBPARM Data Sets” on page 21.

Product Initialization Messages

Product initialization messages were changed for version 3.3.10. If you have automation that depends on the initialization messages or the sequencing of the messages, please see Appendix E on page 101.

Chapter 2. Migrating to Version 3.3.10 from Version 3.2

This chapter describes options you should consider if you are migrating to MVIMS Online version 3.3.10 from MVIMS version 3.2.

Product Authorization

MVIMS Online version 3.3.10 adds support for the use of product authorization tables to activate products on individual CPUs.

Version 3.3.10 continues to support the product keys method of product activation, and you can continue to use the product keys listed in Table 2..

Table 2. MVIMS Online Product Keys

Key	Product or Component
ION	MAINVIEW for IMS Online
IPX	IMSPlex System Manager
IRA	IMS Resource Analyzer
IRM	IMS Resource Monitor
IWA	IMS Workload Analyzer
IWM	IMS Workload Monitor
IAD	IMS Workload Analyzer Extension for DB2
IMD	IMS Workload Monitor Extension for DB2
ILM	IMS Resource Analyzer and Resource Monitor Extensions for IRLM

If MVIMS finds one of the product keys listed in Table 2 during product initialization and the key is valid, MVIMS will not look for a product authorization table. Therefore, if you want to use a product authorization table to activate MVIMS Online, you must remove all MVIMS Online product keys from the BBKEYS member of the BBPARM data set.

If no valid product key is found during initialization, MVIMS looks for the MVIMS Online product authorization table, IONTBL3P, in the data set pointed to by the BMCPSWD DD statement in the startup JCL for the IMS control region. If the BMCPSWD DD is not present, MVIMS looks in the STEPLIB, JOBLIB, and LNKLST data sets. If the IONTBL3P table is not located in those data sets, MVIMS looks in the data set pointed to by the BBILOAD DD. If the IONTBL3P table is found, its authorization information is used to determine whether the CPU is licensed to run MVIMS Online.

Note: If a product key brings MVIMS up in the IMS control region and a product authorization table brings MVIMS up in the product address space (PAS) *or vice versa*, MVIMS will still work properly as long as the provided authorization is consistent. In other words, you will encounter no initialization problems if you bring up one with a product key and the other with a product authorization table.

The procedure required to use a product authorization table to activate MVIMS Online is provided on page 28.

For more information about product authorization tables and the Product Authorization utility, see the *OS/390 and z/OS Installer Guide*.

Product Packaging and Keys

The following MVIMS online components, which were packaged separately in version 3.2, became part of the MVIMS Online product with the release of MVIMS Online version 3.3.00.

IMS Resource Analyzer	IMS Resource Analyzer Extensions for IRLM
IMS Resource Monitor	IMS Resource Monitor Extensions for IRLM
IMS Workload Analyzer	IMS Workload Analyzer Extension for DB2
IMS Workload Monitor	IMS Workload Monitor Extension for DB2
IMSPlex System Manager	

If you have licensed MVIMS Online and you decide to use product keys, rather than product authorization, to run the product, you should use the product key **ION** to enable all the components listed above.

If you have not licensed MVIMS Online and you decide to use product keys, rather than product authorization, to run online components, you can continue to use your product keys from version 3.2.

The product keys you use must be specified in both the PAS BBPARM member BBKEYS and in the IMS control region IMFPARM member BBKEYS.

Detail Trace

Three levels of information are collected by detail trace: CALLS, I/O, and DATA. The DATA level was added in version 3.3.00, and it includes segment search argument (SSA), key feedback area (KFB), and I/O area (IOA) data. Inclusion of the SSA, KFB and IOA data can significantly affect detail trace buffer usage. To help prevent potential buffer size problems, a maximum of 66 bytes is allocated to each of the three record types. The fact that a single call can have multiple SSAs, however, can cause the buffers to fill quickly.

You can choose not to collect the SSA, KFB, and IOA data by setting TRDATA=NO in the BBPARM member IMFBEX00.

Shared Message Queue Statistics

Version 3.3.10 of MVIMS Online includes enhanced support for IMS shared message queues, providing improved realtime and historical shared queues statistics in several of the IPSM workflow, CQS information, transaction, and terminal and user views. Several CQS information views have been added, including the new CQS object menu ISQMR.

When the new Shared Queues Data Server is operating, the views provide shared message queue statistics for IMS systems that use shared queues. You can use the ISQUERY parameter, in BBPARM member IMFBEX00, to turn the Shared Queues Data Server on or off. You can also use the ISQUERY parameter to control the amount of resources used to collect and display shared queues statistics.

For information about how to use the ISQUERY parameter, see “Controlling Shared Message Queue Data Collection” on page 44.

UBBPARM Customization Capabilities

You can now use the UBBPARM data set to include unique IMFSYS00 and IMFECP00 parameter members for individual IMS systems, which eliminates the need to create and allocate an *ibbparm* data set for each IMS system. The IMS-specific settings are defined in members that must have the following name formats:

<i>imsidSYS</i>	for IMS-specific IMFSYS00 system parameters
<i>imsidECP</i>	for IMS-specific IMFECP00 Event Collector parameters

The name of each IMS-specific parameter member must begin with the identification code for the IMS system.

For more information about the IMS-specific UBBPARM members, see “Setting Up BBPARM Data Sets” on page 21.

MSC Clock Synchronization

Version 3.3.10 of MVIMS Online provides support for controlling the handling of the arrival timestamp for MSC systems that originate on a remote system. With IMS 6.1 and later, you can define MSC systems that are synchronized to the local IMS system clock, and you can qualify workload monitor and trace requests to include either

- *all* transactions from systems synchronized to the clock of the local IMS
- or
- *only nonlocal* transactions from MSC systems synchronized to the clock of the local IMS

New Event Collector Parameter MSCCLOCK

A new Event Collector parameter, MSCCLOCK, is provided in BBPARM member IMFECPO0. With IMS 6.1 and later, you can use the MSCCLOCK parameter to specify whether or not MSC systems will, by default, be considered synchronized to the clock of the local IMS system. The MSC systems can be in the same time zone or a different time zone.

If an MSC transaction originates from an MSC system with a clock that is synchronized to the clock of the local IMS, the Event Collector converts its arrival date and time (set by the originating MSC system) to local time. The converted time is then stored in the transaction arrival date and time fields of the transaction log record (X'FA').

If a transaction originates from an MSC system that is not defined as synchronized with the local IMS clock, no conversion is done and the transaction *arrival* date and time are set to the *start* date and time on the local IMS.

Additional MSCCLOCK values (*xxxx,SYNC* and *xxxx,NOSYNC*) are provided so that you can specify IDs for individual MSC systems that will be exceptions to the default you use.

For more information about how to use the MSCCLOCK Event Collector parameter, see page 57.

New TERM Keyword Operands SYNCLOCK and MSCCLOCK

The MSCCLOCK Event Collector parameter works in conjunction with the new monitor TERM keyword operands SYNCLOCK and MSCCLOCK, which are used to qualify workload monitor and trace requests.

TERM=SYNCLOCK defines a workload monitor or trace request to include *all* MSC transactions from systems that are synchronized to the clock of the local IMS (including all MSC transactions that originated on the local IMS). TERM=MSCCLOCK defines the request to include *only nonlocal* transactions from MSC systems that are synchronized to the clock of the local IMS.

Event Collector Option

A new FEATURE=NOEC parameter option was added in version 3.3.00 to allow you to disable the Event Collector if the MVIMS components you use can function without it.

The only MVIMS components that can function without the Event Collector are the Resource Analyzer and the Resource Monitor. If you use only the Resource Analyzer and Resource Monitor components and you want to prevent Event Collector initialization, you can set FEATURE=NOEC in the PARMLIB member IMFSYS00.

Note: Refer to PARMLIB member IMFSYSBB for additional information.

Product Initialization Messages

Product initialization messages were changed for versions 3.3.10 and 3.3.00, and new messages were added for version 3.3.00. If you have automation that depends on the initialization messages or the sequencing of the messages, please see Appendix E on page 101.

Easy Menus

The EZIMS, EZIFAST, and EZISSI Easy Menus were redesigned in version 3.3.00 to improve access to product functions and to incorporate the new features of the release.

Chapter 3. Release Compatibility

This chapter lists the products packaged with the current BBI release and describes MVIMS Online compatibility with previous MVIMS releases.

Compatibility with BBI

MVIMS Online and the following products are packaged with the current BBI 2.6.0 release:

- MAINVIEW AutoOPERATOR
- MAINVIEW for CICS
- MAINVIEW FOCAL POINT
- MAINVIEW for DB2
- MAINVIEW for DBCTL

Compatibility with Previous Release Levels of MVIMS

The UAS (user address space) must be at the same release and service level as the BBI-SS PAS (product address space) and CAS (coordinating address space) to which it connects.

Downward Compatibility

You can access a previous MVIMS version from a UAS running the current version, but the new features and functions of the current version will not be available.

Help panels for services from a previous release might not be accessible from a UAS at the current release level.

Upward Compatibility

You can access the current version of MVIMS Online from a UAS running an earlier version. However, if a current version of a service was changed to support a new IMS function, that service may not be available to you.

The help panels for services added with a new MVIMS Online version and the help panels modified for revised services cannot be accessed from a UAS at an earlier version level.

To review a list of features or services that are new or have changed with the current version of MVIMS Online, see the release notes, which are available on the BMC Software Web site at <http://www.bmc.com/support.html>.

AO Exit

If a previous version of the BMC Software AO exit was installed on the target IMS, make sure that all remnants of the previously installed version are removed by following the instructions in the section “Deleting Modules from a Prior Release” on page 20.

Part 2. Customizing MAINVIEW for IMS Functions

This part of the book describes how to tailor MVIMS Online to meet your site's needs.

Chapter 4. Resource Utilization	17
CPU Usage – Event Collector	17
CPU Usage - IPSM Samplers.	18
CSA Usage	18
Chapter 5. Implementing MAINVIEW Products in IMS	19
Deleting Modules from a Prior Release	20
Setting Up BBPARM Data Sets	21
Customizing BBPARM Members for an IMS System.	23
Activating MAINVIEW for IMS Online	25
Enabling Product Initialization	25
Enabling Product Authorization	25
Product Keys	25
Product Authorization Table	26
Modifying the IMS Control Region JCL	27
Allocating a BBPARM Data Set	27
Giving IMS Access to the Event Collector	27
Adding Access to a Product Authorization Table	28
Modifying DLISAS Region JCL	28
Enabling AO Exit Routines	29
Chapter 6. Customizing the Trace Facility	33
Specifying Trace Defaults in IMFBEX00	34
Detail Trace Data Collection	34
Trace Display Buffer Size	34
Trace Duration.	35
Trace Logging Options	35
Updating Trace Defaults.	36
Specifying Detail Trace Buffers	36
Setting Up and Maintaining a Trace Directory	37
Defining and Initializing a Trace Directory Data Set	37
Identifying the Trace Directory to BBI	37
Verifying Trace Directory Entries	38
Managing Trace Log Data Sets	39
Defining a Trace Log Data Set	39
Archiving a Trace Log Data Set	39
Restoring an Archived Trace Log Data Set.	39
Creating a Trace Log Data Set from the IMS Log	39
Printing a Trace Log Data Set	39
From a Batch Job	39
From an Online Application	40
Setting Up Workload History Traces	41
Setting Limits for Transaction Trace Views	42
Chapter 7. Controlling Workload Thresholds and SMQ Data Collection	43
Specifying Thresholds for IMS and IMS Sysplex Activity Views.	43
Controlling Shared Message Queue Data Collection.	44
CSQ Information Views with Message Age Fields.	44
Non-Workflow Views with Message Count Fields	45
Workflow Views with Message Count Fields	45

Chapter 8. Using the Event Collector	47
Starting and Stopping the Event Collector	47
Specifying Event Collector Options	47
Evaluating Data Collection Options	60
Reviewing the Initialization Parameters	61
Parameter Option Sets	65
Standard Option Set	65
Full Option Set	65
Minimum Option Set	66
How the Option Sets Affect Online Service Data	67
Workload Trace (LTRAC, STRAC, and DTRAC)	67
Changing Event Collector Parameters	68
Chapter 9. Specifying Target IMS Systems	69
Chapter 10. Customizing with User Exit Routines	71
Event Collector User Exit Routines	71
Transaction Record User Exit (IMRUTRN)	71
Program Record User Exit (IMRUPGM)	71
IMRUTRN and IMRUPGM Cross-Memory Mode Considerations	72
DL/I-CALL-END User Exit (IMRUDLI)	72
Chapter 11. Service Utility Commands	73
Lock a Service (LOCK)	73
Unlock a Service (ULOCK)	73
Chapter 12. Optional Modifications to MAINVIEW for IMS	75
Modification to Support SAP	75
SMP-Applied Modifications	76
Modification to Support TELON	76
Modifications to Reconcile Response Time Monitoring Data	77
Zero Out Input Queue Time for Selected Transactions	77
Zero Out Input Queue Time for Message Switch Transactions	77

Chapter 4. Resource Utilization

This chapter summarizes the options that can be used to control usage of CPU by

- the Event Collector
- IPSM samplers

For more information about the Event Collector options, see Chapter 8, “Using the Event Collector” on page 47. For more information about options that can be used to control IPSM sampler usage of CPU, see the *MAINVIEW for IMS Online – IPSM Reference Manual*.

CPU Usage – Event Collector

The Event Collector is always active, even when there are no active traces or workload monitors. The default Event Collector option, CPU=DEP, provides a reasonable balance between CPU usage and the detail level of the data the Event Collector collects. If the default option results in too much CPU usage for your site, you can change the CPU option in BBPARM member IMFECP00 to reduce CPU usage.

The CPU options that control Event Collector data collection are listed below in order of lowest to highest CPU usage.

1. CPU=NONE

CPU time data is not collected.

2. CPU=DEPPGM (the default)

Collects CPU time data only from dependent region activities, including DB2 CPU time, and times the entire transaction as a single event. This option reduces overhead significantly.

3. CPU=DEPDB2

Functions the same as the DEPPGM default option except that CPU for DB2 events is reported separately from dependent region CPU time. CPU usage depends upon the number of SQL calls issued by the transaction.

4. CPU=DEP

Collects CPU time data from dependent regions.

5. CPU=ALL

Collects all CPU time data

For more information about how the CPU options work, see “Specifying Event Collector Options” on page 47.

Note: The Resource Analyzer and Resource Monitor components can function without the Event Collector. If you use only the Resource Analyzer and Resource Monitor components and you want to prevent Event Collector initialization, you can set FEATURE=NOEC in the PARMLIB member IMFSYS00. (For more information, see the PARMLIB member IMFSYSBB.)

CPU Usage - IPSM Samplers

IPSM collects target system sampling for the components of response time (CORT) and workflow views. Samplers must run continually in the BBI-SS PAS if you want to collect this information. The distributed sampler default provides for a sampling period of 24 hours a day at 2 times per second.

If the default results in too much CPU usage for your site, you can reduce CPU usage by creating and adding a sampler target definition. For more information, see the *MAINVIEW for IMS Online – IPSM Reference Manual*.

CSA Usage

The *MAINVIEW Installation Requirements Guide* lists the virtual storage requirements for all MAINVIEW products.

Chapter 5. Implementing MAINVIEW Products in IMS

This chapter describes how to install and implement BMC Software modules, which are required for

- MAINVIEW AutoOPERATOR for IMS
- MAINVIEW for IMS (MVIMS) Online

Follow the procedures in this chapter if you are customizing one or both of these products and you have not performed AutoCustomization. AutoCustomization tailors your products automatically.

You can refer to the manual customization steps in this chapter if you need help during AutoCustomization.

Note: If you have multiple IMSs, you may want to allocate a BBPARM data set that is unique to an IMS, as described in “Setting Up BBPARM Data Sets” on page 21 and “Modifying the IMS Control Region JCL” on page 27.

For information about how to use product libraries, including parameter libraries (BBPARM and UBBPARM) and sample libraries (BBSAMP and UBBSAMP), see “Using MAINVIEW Product Libraries” in the *MAINVIEW Common Customization Guide* or “Using Product Libraries” in the *MAINVIEW Administration Guide*.

Deleting Modules from a Prior Release

You can skip this section if you are installing MVIMS Online for the first time.

If a previous release of MAINVIEW for IMS, MAINVIEW for DBCTL, or MAINVIEW AutoOPERATOR for IMS was installed on the target IMS, use the information in Table 3 and follow the instructions below.

If you copied BBLINK members to an authorized STEPLIB data set using ICOPY, you can use the following BBSAMP jobs to delete the old modules:

Table 3. BBSAMP Jobs to Delete Old Modules

IMS Release	MVIMS 3.1 MVDBC 2.1 AO 3.1, AO 4.1	MVIMS 3.2 MVDBC 3.2 AO 4.1	MVIMS 3.3.mm MVDBC 3.3.mm AO 5.1, AO 6.1
IMS 5.1	IDEL31\$5	IDEL32\$5	IDEL33\$5
IMS 6.1	n/a	IDEL32\$6	IDEL33\$6
IMS 7.1	n/a	n/a	IDEL33\$7

To delete the old modules:

1. From Table 3, select the delete job that corresponds to the BMC Software product (or products) and IMS release installed at your site.
2. Edit the delete job you selected to change all &RESLIBS to the name of the data set where the old modules reside.
3. Run the delete job.

If you added BBLINK to the IMS STEPLIB concatenation, replace the BBLINK data set in the current IMS STEPLIB concatenation with the new BBLINK data set.

If you included BBLINK in the link list concatenation, replace the BBLINK data set in the current link list concatenation with the new BBLINK data set. This data set replacement will affect all the IMS address spaces running in that OS/390.

Setting Up BBPARM Data Sets

The customization instructions in this chapter refer to the following data sets:

hilevel.ibbparm
hilevel.UBBPARM
hilevel.BBPARM

hilevel The high-level data set name qualifier used at your site

ibbparm A user-defined parameter data set that is unique to this IMS

You can allocate a separate *ibbparm* data set to contain any members that you want to make unique to the IMS, such as IMFSYS00 and IMFECPO0.

The *ibbparm* data set must be allocated; it is not created through AutoCustomization.

Note: If the only members that require customization for an IMS are IMFSYS00, IMFECPO0, or both, an *ibbparm* data set is not required. Instead, you can create a renamed (*imsidSYS*) version of the IMFSYS00 member, a renamed (*imsidECP*) version of the IMFECPO0 member, or both, include them in UBBPARM, and customize them to suit your needs (as explained below in UBBPARM).

UBBPARM A parameter data set that is tailored from the distributed BBPARM data set and is shared by all IMS systems

If you used AutoCustomization, you can use the UBBPARM data set created by AutoCustomization. If you did not use AutoCustomization, allocate UBBPARM, copy the distributed BBPARM data set to it, and tailor UBBPARM to suit your needs.

In the UBBPARM data set, you can include a copied, renamed version of IMFSYS00 to customize the system parameters for an IMS system, including the SUBSYS parameter, which establishes communication between the IMS and the BBI-SS PAS. The name of an IMS-specific system parameter member must be in the following format:

imsidSYS (where *imsid* is the IMS ID code)

You can also include a copied, renamed version of IMFECPO0 to customize the Event Collector parameters for an IMS. The name of an IMS-specific Event Collector parameter member must be in the following format:

imsidECP (where *imsid* is the IMS ID code)

Note: If you do not need to customize *other* UBBPARM members for an IMS system, you do not need to create and allocate an *ibbparm* data set for that system.

BBPARM The target BBPARM data set distributed by BMC Software

Important

Throughout the MVIMS books, parameter library members are normally referred to as BBPARM members, even though customized versions of the members may reside in the UBBPARM data set or in an *ibbparm* data set.

The system parameter member is normally referred to as BBPARM member IMFSYS00, and the Event Collector parameter is normally referred to as BBPARM member IMFEC00, even though the members may reside in a parameter data set with a member name in the format *imsidSYS* or *imsidECP*.

Customizing BBPARM Members for an IMS System

This section is provided primarily for new installations of MVIMS Online. If you have a prior version installed, you can use your existing parameter settings, but this would be a good time to review them.

This following procedure customizes the parameter members required to

- establish communication between an IMS region and the BBI-SS PAS
 - run the Event Collector to collect workload monitor, trace, and wait data
 - run the batch jobs used for report printing
1. Set the system parameters in an *imsidSYS* member in the UBBPARM data set or in an IMFSYS00 member in an *ibbparm* data set. The parameters and their uses are as follows:

SUBSYS. Use this parameter to identify the subsystem name of the BBI PAS that the IMS region should communicate with. The subsystem should be the same as the one specified in the BBIJNT00 member of the UBBPARM data set or an *ibbparm* data set.

MSGLVL1. Set the message level to MTO, WTO, BOTH, or NONE.

KEYWARN. Use this parameter to set a minimum number of days before expiration warning messages are issued for product keys. The default is 45 days. (The KEYWARN parameter applies to MVIMS Online, MVIMS Offline, and MAINVIEW for DBCTL only, and it is used only in the IMS control region, not the BBI-SS PAS.)

AOEINIT. This parameter is used with the DFSAOE00 initialization call. (See BBPARM member IMFSYS00 for more information.)

AOEEXIT and AOIEXIT. Use these parameters to specify the names of user-written AO exit routines. You can also use AOEEXIT and AOIEXIT to specify the order in which AO exit routines get control and whether a return code is padded. See “Enabling AO Exit Routines” on page 29 for more information.

As distributed, IMFSYS00 activates all MVIMS components in IMS. You can use *imsidSYS* in the UBBPARM data set or IMFSYS00 in an *ibbparm* data set to temporarily deactivate one or more components. BBPARM member IMFSYSBB contains information about how to deactivate components. Copy what you need from IMFSYSBB to your UBBPARM *imsidSYS* member or *ibbparm* IMFSYS00 member.

2. Set up the Event Collector data collection parameters in an IMFECPO0 member or an *imsidECP* member.

Note: Use UBBPARM member IMFECPO0 for Event Collector parameters that are to be shared among multiple IMSs. To specify IMS-specific Event Collector parameters, use an *imsidECP* member in the UBBPARM data set or an IMFECPO0 member in an *ibbparm* data set.

The following parameters and recommended values set up Event Collector data collection:

To record BMP data, specify

BMP=YES (default)

To time an entire transaction as a single event, specify

CPU=DEPPGM (default)

To write DBCTL transaction records to the IMS log and send them to the BBI-SS PAS for Workload Analyzer wait and trace processing and for Workload Monitor processing, specify

CI CS=YES (default)

3. Set up the Log Edit utility parameters in the IMFLEPO0 member.

Note: Use UBBPARM member IMFLEPO0 for Log Edit utility parameters that are to be shared by multiple IMSs. For IMS-specific Log Edit utility parameters, use the IMFLEPO0 member in an *ibbparm* data set.

The following parameter in IMFLEPO0 is required only if you have MAINVIEW for IMS Offline installed. The parameter is used by the Log Edit utility (IMFLEDIT).

To specify the release level of the IMS system where the IMS log is created, specify

IMSLEVEL=5100 | 6100 | 7100 | 0000

where

5100 IMS 5.1

6100 IMS 6.1

7100 IMS 7.1

0000 Causes the Log Edit utility to scan up to the first 50K records of the log tape to determine the IMS release

Note: For more information about the Event Collector parameters and their use, see Chapter 8 on page 47. For more information about the Log Edit utility, IMFLEDIT, see the Log Edit chapter in the *MAINVIEW for IMS Offline – Customization and Utilities Guide*.

Activating MAINVIEW for IMS Online

To activate MVIMS Online, you need to enable product initialization and product authorization.

Enabling Product Initialization

To enable MVIMS Online initialization, in BBPARM member BBISSP00, specify

`PRODUCT=MVIMS`

The `PRODUCT=MVIMS` parameter setting initializes all MVIMS Online product components at BBI-SS PAS startup.

Enabling Product Authorization

You can use product keys or a product authorization table to enable MVIMS Online product authorization.

Product Keys

If you have licensed MVIMS Online and you want to use a product key to activate the product, specify product key ION in BBPARM member BBKEYS.

If you have not licensed MVIMS Online and you want to use product keys to activate the online components, you can use the product keys listed in Table 4.

Table 4. MVIMS Online Product Keys

Key	Product or Component
ION	MAINVIEW for IMS Online
IPX	IMSPlex System Manager
IRA	IMS Resource Analyzer
IRM	IMS Resource Monitor
IWA	IMS Workload Analyzer
IWM	IMS Workload Monitor
IAD	IMS Workload Analyzer Extension for DB2
IMD	IMS Workload Monitor Extension for DB2
ILM	IMS Resource Analyzer and Resource Monitor Extensions for IRLM

For detailed information about specifying product keys, see the step called “Specify Product Option Password Keys” in the *MAINVIEW Common Customization Guide*.

Product Authorization Table

If you decide to use a product authorization table to activate the MVIMS Online components, complete the following steps:

1. Create the product authorization table IONTBL3P, either through AutoCustomization or with a batch job. Both methods are described in the *OS/390 and z/OS Installer Guide*.
2. Add access to the product authorization table (as described on page 28).

Note that the product initialization messages will now include BBAP prefixed messages issued by the BMC License Manager.

Modifying the IMS Control Region JCL

The following sections describe how to allocate a BBPARM data set, provide IMS access to the Event Collector, and add access to a product authorization table.

Allocating a BBPARM Data Set

To establish the parameters for MVIMS Online and MAINVIEW AutoOPERATOR for IMS, allocate the BBPARM data sets by adding the following statements to the IMS control region startup procedure:

```
//IMFPARM DD DSN=hi level. ibbparm  
//      DD DSN=hi level. UBBPARM  
//      DD DSN=hi level. BBPARM
```

Giving IMS Access to the Event Collector

The IMS control region must be able to access the Event Collector to enable execution of MVIMS Online and MAINVIEW AutoOPERATOR for IMS.

If the BBLINK data set is in the LNKLST concatenation, Event Collector access is already established. If not, you can provide Event Collector access by modifying your IMS control region JCL or by copying the required modules to the site authorized library (such as IMS RESLIB).

To modify the IMS control region JCL, add the BBLINK data set to the IMS STEPLIB concatenation. (BBLINK must be authorized.)

To copy the modules used in IMS, copy the individual BBLINK members to an authorized STEPLIB data set. Select one of the following jobs in BBSAMP:

- ICOPY8 for IMS 5.1
- ICOPY9 for IMS 6.1
- ICOPY1 for IMS 7.1

Then edit the JCL and run the job. This job copies the appropriate BMC Software modules from the BBLINK library to the site-authorized library, such as IMS RESLIB. You must rerun the JCL each time you apply BMC Software service.

Adding Access to a Product Authorization Table

If you are using the IONTBL3P product authorization table to activate MVIMS Online, the IMS control regions must be able to access the table.

There are four ways you can provide access to the table:

- Concatenate a data set containing the IONTBL3P product authorization table to the STEPLIB data set in the IMS control region JCL.
- Add the following statement to the IMS control region JCL:

```
//BMCPSWD DD DSN=hilevel.BMCPSWD,DISP=SHR
```
- Copy the IONTBL3P table into RESLIB or BBLINK.
- Add the IONTBL3P data set to LNKLIST.

If you want the product authorization table to reside in RESLIB, follow the instructions in the “Product Maintenance or Version Upgrades” section in Appendix A of the *OS/390 and z/OS Installer Guide*.

Note: If you previously used BBKEYS to activate the products, you must remove all keys for MVIMS from the BBKEYS member of BBPARM.

Modifying DLISAS Region JCL

This step modifies the DLISAS region JCL to provide access to BBLINK modules for database activity monitoring in the IMSplex System Manager (IPSM) product.

Make BBLINK modules available to the DLISAS region by adding the BBLINK data set to the DLISAS STEPLIB concatenation (BBLINK must be authorized).

Enabling AO Exit Routines

You can skip this section if you have a prior version of MVIMS Online installed.

IMS gives control to AO exit routines to do initialization processing and to do message processing. BMC Software does not supply an AO exit routine to do initialization processing. If you have your own routine, specify its one- to eight-character load module name in the AOEINIT parameter in an *imsidSYS* member in the UBBPARM data set or an IMFSYS00 member in an *ibbparm* data set.

BMC Software supplies two AO exit routines to do message processing: a type-1 AO exit routine, DFSAQUE0, and a type-2 AO exit routine, DFSAOE00.

Note: See IBM's *IMS Operations Guide* for an explanation of how the DFSAQUE0 and DFSAOE00 routines differ.

The DFSAQUE0 and DFSAOE00 routines capture MTO messages and IMS commands and pass them to MAINVIEW AutoOPERATOR for IMS. They also perform the following functions:

- provide an interface with your AO exit routines (if any)
- automatically start up MVIMS monitors when IMS starts up
- provide an interface between the Event Collector and the BBI-SS PAS for workload data collection
- initialize the Event Collector

On entry to DFSAOE00 with AOE0FUNC=1, MVIMS loads and executes any user exit specified in AOEINIT=xxxxxxx one time only. The DFSAOE00 exit does not forward control to the user's DFSAOE01 exit on the initialization call.

If you require a DFSAOE00 or DFSAQUE0 exit routine in addition to the exit provided by BMC Software, perform Step 1 and Step 2 below.

1. If you have
 - a. **One DFSAOE00 exit:** Rename it DFSAOE01, or choose a different name and use the control statements described in Step b.
 - b. **Multiple DFSAOE00 exits in addition to the BMC Software-supplied exit, or one exit not named DFSAOE01:** You must add one or more of the following AOEEXIT control statements to an *imsidSYS* member in UBBPARM or to an IMFSYS00 member in an *ibbparm* data set.

For example:

```
AOEEXIT=MYNAME  
AOEEXIT=DFSAOE02
```

In the example above, both exits are loaded and executed by the BMC Software DFSAOE00 exit. Program MYNAME executes first.

2. If you have

- a. **One DFSAOUE0 exit:** Rename it DFSAOUE1, or choose a different name and use the control statements described in Step b.
- b. **Multiple DFSAOUE0 exits in addition to the BMC Software-supplied exit, or one or more exits not named DFSAOUE1:** You must add one or more of the following AOIEXIT control statements to an *imsidSYS* member in UBBPARM or to an IMFSYS00 member in an *ibbparm* data set.

For example:

```
AOIEXIT=MYNAME  
AOIEXIT=DFSAOUE2
```

In the example above, both exits are loaded and executed by the BMC Software DFSAOE00 exit. Program MYNAME executes first.

Flow of Control: By default, the BMC Software DFSAOE00 exit invokes the BMC Software AO exit routine (IELOAD) before it invokes your user exits. The return code set by the last user exit processed is passed to IMS. You can change the default processing in an *imsidSYS* member in the UBBPARM data set or an IMFSYS00 member in an *ibbparm* data set by specifying

- the order that exits are to be processed
- the exit return code

For example:

```
AOEEXIT=(DFSAOE01,RC)  
AOEEXIT=IELOAD
```

In the example above, the DFSAOE01 user exit assumes control before the IELOAD exit and the DFSAOE01 return code is passed to IMS.

IMS then executes the DFSAOUE0 exit unless indicated otherwise by the return code set by the DFSAOE00 exit. The BMC Software DFSAOUE0 exit calls your exits in the order you specified with the AOIEXIT control statements. By default, the return code set by the last user exit processed is passed to IMS. You can pass the return code from another exit by specifying the RC parameter with the AOIEXIT control statement as shown in the following example:

```
AOIEXIT=(USEMINE,RC)  
AOIEXIT=IGNOREME
```

Enhancement to the IMS DFSAOE00 Interface: BMC Software enhances the programming interface when your DFSAOE00 exit is invoked in the following ways:

- Register 11 contains the address of the IMS SCD.
- Register 13 contains the address of 15 prechained save areas.

MAINVIEW AutoOPERATOR for IMS Considerations: When you install your user exit with a BMC Software exit, you should consider the following information:

- If the BMC Software exit is executed before the user exit and the user exit's return code cancels the processing of additional message segments of a multisegment message, the message segments are also canceled for MAINVIEW AutoOPERATOR. The result is that incomplete IMS messages are passed to the MAINVIEW AutoOPERATOR and LAST SEG LOST messages from BBI. The MAINVIEW AutoOPERATOR AO exit holds the first segment for a certain length of time while waiting for additional segments, which can cause the messages to be processed out of timestamp sequence by MAINVIEW AutoOPERATOR.
- When the user exit is executed before the BMC Software exit, changes to the messages made by the user exit are received by the MAINVIEW AutoOPERATOR AO exit.
 - If the user exit sets the length code of a message (or a segment of a multisegment message) to zero, MAINVIEW AutoOPERATOR does not process the message (or segment).
 - If the user exit sets the length code of the first segment to 0 and the return code to 4, all further segments are canceled for MAINVIEW AutoOPERATOR also.

Note: If any modules are specified with the AOIEXIT parameter in IMFSYS00 or *imsidSYS*, but the BMC Software exit is not specified, the exit still executes after all specified exits.

The MAINVIEW AutoOPERATOR AO exit and any other AO exit you use are under ESTAE protection when invoked by the BMC Software routines. If an abend occurs in one of these exits, only that routine is disabled; the other routine and IMS itself are not affected.

Chapter 6. Customizing the Trace Facility

This section describes how to manually

- Specify trace defaults in BBPARM member IMFBEX00
- Set up and maintain a trace directory and trace log data sets

A request for a trace can request that the trace data be recorded to VSAM data sets for later viewing or printing. The logging of trace data requires a pre-allocated trace directory that must be identified to the BBI-SS PAS. Setup of the trace directory can be done automatically by AutoCustomization, as described in the *MAINVIEW Installation Requirements Guide*.

An alternative is to set up the trace directory manually, as described in this section. Trace log data sets can be pre-allocated, as described here, or they can be dynamically allocated at the time of the trace request, as described in the *MAINVIEW for IMS Online – Monitors and Traces Reference Manual (Monitors and Traces Reference Manual)*.

Tuning Tip: If one or more summary traces are defined to start automatically, a continuous workload history is available for later viewing.

Specifying Trace Defaults in IMFBEX00

The IMFBEX00 member of the BBPARM data set defines trace request defaults. These defaults prime the options for the trace request data entry panels.

This section describes the IMFBEX00 keywords that define option defaults for

- All trace requests
- Trace logging

Detail Trace Data Collection

Three levels of information are collected by detail trace: CALLS, I/O, and DATA. The DATA level includes segment search argument (SSA), key feedback (KFB), and I/O area (IOA) data. Inclusion of the SSA, KFB and IOA data can significantly affect detail trace buffer allocation. To help prevent potential buffer size problems, a maximum of 66 bytes is allocated to each of the three record types. The fact that a single call can have multiple SSAs, however, can cause the buffers to fill quickly.

You can choose not to collect the SSA, KFB, and IOA data by setting TRDATA=NO in BBPARM member IMFBEX00.

Trace Display Buffer Size

The following keyword defines the default for the trace display buffer size (STORAGE option). This option applies for any trace and is presented when a trace is requested.

STORAGE Specifies the size of the display buffer for the requested trace. This value overrides any value defined in BBIISP00.

MTRAC stores trace data in the private storage area of the BBI-SS PAS address space in extended private storage. This should be considered when setting up storage requirements for the BBI-SS PAS (see the *MAINVIEW Common Customization Guide* for a description of the storage requirements). When the available allocated area is full, the newest data wraps around and overlays the oldest data unless WRAP=NO is specified with the MTRAC request.

The size of the trace data storage area in the BBI-SS PAS can be specified with the STORAGE parameter in the MTRAC request data entry panel; for example:

```
STORAGE ==> 100K
```

requests a GETMAIN storage of 100K. The default can be specified in either the IMFBEX00 or the BBIISP00 member of the BBI-SS PAS BBPARM data set. IMFBEX00 has priority over BBIISP00.

A summary trace entry requires a minimum of 776 bytes of storage per transaction. The largest amount of storage that can be used is 24,776 bytes. The amount of storage required depends on the number of database trailers used by the Event Collector for the transaction.

Specifying a larger storage value prevents frequent wrapping.

A detail trace generally requires a much larger storage area than a summary trace; 52 bytes are required per detail line. So a detail trace of a transaction with 20 detail lines requires about $20 * 52 +$ summary trace bytes of storage.

Trace Duration

The following keyword defines the default for the trace duration (STOP option). This option applies for any trace and is presented when a trace is requested.

TRTIME=*n* Specifies the default duration of a trace in minutes (1 to 32,000). The default is no limit.

Note: If TRTIME is specified, the STOP keyword in the MTRAC Start IMS Trace Request data entry panel is primed with this value. If a value is not specified, STOP is not primed. A STOP value that is not in the hh: mm: ss format is interpreted as a STOPCNT value in minutes.

Trace Logging Options

The following keywords define the defaults for a trace log data set allocation request:

TRPREFIX Defines the data set name prefix for trace log data sets if the value for the Log DSN option on the Start IMS Trace Request panel is specified without quotation marks.

If a value for TRPREFIX is not defined, the ID of the user requesting the trace is used.

TRREUSE Requests data to be overwritten if a log data set is not reset. N (NO) indicates that data is not to be overwritten. The default is Y (YES).

If the request specifies a 1 for the number of logs and N is defined for TRREUSE, data is not recorded. If the request specifies a 1 and Y is defined for TRREUSE, previous data recorded in the log is overwritten.

TRVOLS=(*x, x, . . .*) Specifies the ID of the default volume(s) for trace log data set allocation. Up to seven volumes can be specified. The default value specified in IMFBEX00 with the TRVOLS keyword is SYSDA.

TRCYL Defines the primary allocation default in cylinders (CYLS option) for trace log data sets. The default value is 3.

TRSUFFIX Defines the default suffix to add to the name of the trace cluster data set (Data DSN Suffix option) to make the data set name for the data component. The default value is D.

TRSMSSCL Defines the default name of the SMS storage class for trace log data set allocation. There is no default value.

TRMSDCL Defines the default name of the SMS data class for trace log data set allocation. There is no default value.

TRMSMCL Defines the default name of the SMS management class for trace log data set allocation. There is no default value.

Updating Trace Defaults

You can refresh the parameters specified in BBPARM member IMFBEX00 by starting or stopping the BBI-SS PAS or by using the following command:

```
. RESET PARM IMFBEX00
```

This command refreshes IMFBEX00 only. For more information about the RESET command, see the *MAINVIEW Administration Guide*.

Specifying Detail Trace Buffers

The following keywords define the default for the TRBUFF and TRSIZE data collection buffers:

TRBUFF=*nnn* Specifies the total number of detail trace buffers to be allocated. The number should be at least equal to the number of concurrent active regions plus 2. (This value overrides any value defined in BBIISP00.)

TRSIZE=*nnn* Specifies the size of each buffer in bytes. You can specify the value as *nnn* or as *nnK*. The number is rounded to a multiple of 1K. (This value overrides any value defined in BBIISP00.)

For example, if **TRBUFF=20** and **TRSIZE=32K**, total size is 20*32K=640K.

Note: The detail trace is truncated if more events are being traced than can fit in one buffer. To trace long-running batch programs, you may need to increase TRSIZE.

A detail trace stores trace data for active transactions in ECSA buffers. The number and size of the detail trace buffers can be specified in either the IMFBEX00 or the BBIISP00 member of the BBPARM data set. IMFBEX00 has priority over BBIISP00. The defaults are

```
TRBUFF=10
TRSIZE=4
```

TRBUFF defines the number of detail trace buffers. TRSIZE defines the size of each trace buffer. When an active transaction completes, the contents of its buffer are moved to the trace areas in the BBI-SS PAS and can then be displayed by using the DTRAC service.

A pool of buffers is GETMAINED in ECSA when a detail trace is activated. The pool is shared if multiple detail traces are activated. Then buffers are dynamically allocated to the dependent regions as needed. The buffers are returned to the pool when the trace areas in the BBI-SS PAS are updated. If, during transaction initialization, one of the buffers cannot be obtained, only a summary record is generated for the transaction. The buffer pool is FREEMAINED either when the last detail trace stops (and the program running in each region which has a detail trace buffer allocated terminates) or when the target system terminates. The buffers in ECSA are obtained only if a detail trace is activated.

Note: To calculate an appropriate TRSIZE, estimate about 16K per 100 DL/I or SQL calls. The recommended value for TRBUFF is the maximum number of IMS dependent regions running concurrently plus 2. If there are not enough detail trace buffers, only a summary trace record is created.

Setting Up and Maintaining a Trace Directory

You can bypass this step if you used AutoCustomization.

Before a request for trace logging can be started, a trace directory must be pre-allocated and initialized. This section describes how to set up the trace log directory using sample members in the BBSAMP data set and BBPARM member BBIISP00.

Note: If a security management system is installed, you may need to grant BBI-SS PAS authorization to allocate trace log data sets dynamically (see *Implementing Security for MAINVIEW Products*).

Defining and Initializing a Trace Directory Data Set

There is one trace directory per BBI-SS PAS. The trace directory is a VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information. Entries can be added to or deleted from the directory to allow trace logs to be moved between systems.

To define and initialize the trace directory, use BBSAMP sample member JXT001. Follow these steps:

1. Add your job statement.
2. Update the symbolics as necessary.
3. Submit the job.

Identifying the Trace Directory to BBI

To identify the trace directory to BBI, use BBPARM member BBIISP00 and specify

```
TRDIR=dsn
```

The value `dsn` represents the data set name of a trace log data set directory (there is no default name). The directory must be allocated and initialized before any trace can be started with trace logging. BBSAMP member JXT001 creates the trace directory.

Verifying Trace Directory Entries

Trace directory entries are not updated automatically by events occurring outside of the BBI-SS PAS, such as data set deletion or archival. So, you may occasionally need to synchronize the trace directory information with the actual status of the data sets.

To verify, purge, or print directory entries, use BBSAMP member JXT003. This member checks for the existence of a trace log data set in the system catalog.

Note: Since every entry in the trace directory is allocated dynamically and read to verify its current status, this process could run for some time.

To synchronize trace directory information with the actual status of the data sets, follow these steps:

1. Add your job statement.
2. Update the symbolics as necessary.
3. Specify one of the following processing options for PARM. If PARM is not specified, no action is taken.

Blank Causes an uncataloged entry to be marked as INV (INVALID). The same thing happens when PARM is not specified. (Blank is the default).

ARCVOL= Specifies an archive volume serial number. This value is matched against the volume serial number in the system catalog for each entry in the directory. If there is a match, the data set is not verified. You can use this option to bypass recalling all trace log data sets from archives.

LIST Lists the directory entries that are changed. If NOVERIFY is specified or implied, all entries are listed (equivalent to LISTALL).

LISTALL Lists all entries.

NOLIST Does not list changed entries.

PURGE Deletes any data sets in the directory that are invalid trace data sets.

NOPURGE Does not delete invalid data sets (marked as INV (invalid) in the directory).

VERIFY Verifies each of the entries in the trace directory.

Note: If VERIFY is specified, the defaults are LIST, WRITE, and NOPURGE.

NOVERIFY Does not verify entries in the trace directory.

WRITE Updates trace directory with status changes.

NOWRITE Does not update trace directory with changes detected.

4. Submit the job.

Managing Trace Log Data Sets

This section describes how to create and manage trace log data sets manually using sample members from the BBSAMP data set.

Defining a Trace Log Data Set

You can define different trace logs as often as you need them, or you can let the BBI-SS PAS allocate them for you dynamically (see the JXT011 sample job description in the *Monitors and Traces Reference Manual*).

Archiving a Trace Log Data Set

A trace request can be defined to archive a log data set automatically when it is full, as described in the IMFTARC sample job description in the *Monitors and Traces Reference Manual*.

You can manually submit the IMFTARC job to archive a trace log data set that is no longer active.

Restoring an Archived Trace Log Data Set

Use BBSAMP member IMFTRLOD to restore an archived trace log data set.

Note: You can also add the linear data set to the online trace directory and view the contents online. Use the NEW command in the History Traces application, as described in the *Monitors and Traces Reference Manual*.

Creating a Trace Log Data Set from the IMS Log

You can use BBSAMP member IMFLOGTR to create a user-selected summary trace from the IMS log. See the *Monitors and Traces Reference Manual* for a description of how to use IMFLOGTR.

Note: You can add the trace log data set created by IMFLOGTR to the online trace directory and view the contents online. Use the NEW command in the History Traces application, as described in the *Monitors and Traces Reference Manual*.

Printing a Trace Log Data Set

You can print a trace log data set either from a batch job or from an online application.

From a Batch Job

You can use BBSAMP member WATBTRAC to print a trace log data set. For a full description of how to print a trace from a batch job, see the chapter about printing history traces in the *Monitors and Traces Reference Manual*.

From an Online Application

You can print a trace log data set from the online History Traces application if you are executing your TS from ISPF.

Before printing from the online application, you must first copy the skeleton JCL located in member WATBPRNT of the BBPROF library. Copy this member to an individual user data set (BBPROF) or to a site data set (SBBPROF). The BBPROF or SBBPROF data set must be defined in the CLIST (MAINVIEW CLIST) used to start the terminal session. For more information about BBPROF, see the *MAINVIEW Common Customization Guide*.

To print from the online application:

1. Select option 4 from the Primary Option Menu.
2. In the History Traces panel, enter the P line command next to the data set you want to print. (For a full description of how to use line commands in the History Traces application, see the *Monitors and Traces Reference Manual*.)
3. Enter the required information in the next panel that appears. (This panel gives you options that allow you to tailor the print job output to your needs.)
4. Press the End key.

The printed trace data has the same format and content as the online display.

Setting Up Workload History Traces

The BBIISP00 member of the BBPARM data set allows you to select a group of timer-driven monitor and trace requests to start automatically. These requests are defined in another member of the BBPARM data set. If you specify a default block request member BLKIMFT, for example, in BBIISP00 (TARGET=i msi d, BLK=BLKI MFT), a starter set of monitors and the following summary trace could be requested:

```
REQ=MTRAC TRHIST TYPE=SUMMARY TITLE=' TRACE HI STORY' STORAGE=4000K
*          LOGTRAC=Y  TRNUMDS=3  TRSWTIME=24: 00
```

The example above is a summary trace of the complete IMS workload. It should be run as a standard request to provide viewing of trace history. You can access it directly from the History Traces application. It adds very minimal overhead since it requires only a summarized trace.

The second line is set up as a comment to show you how you could define trace logging to a set of three data sets, automatically switching to a new data set at midnight. Depending on your IMS workload volume and operations procedures, you may need to modify some of these keyword values or specify others.

All options are defined in the *Monitors and Traces Reference Manual* in “Requesting Workload Trace Data Collection (MTRAC).” Setting defaults for all traces, such as the volumes to be used for allocation, is described in “Specifying Trace Defaults in IMFBEX00” on page 34 of this manual.

Although there are many options available, there are basically two ways to set up continuous trace logging. You must evaluate your system characteristics before choosing which is better.

- The first method uses automatic allocation of one or more new trace log data sets each time the trace request is started (at BBI-SS PAS startup), as shown in the preceding example. No DSN is specified so that the generated name is always unique (specifying TRPREFIX in IMFBEX00 defines the high level node).

This method can be used if OS/390 and the BBI-SS PAS are rarely brought down. The only consideration is that if the trace log data set allocation fails, perhaps because of lack of space, the trace request also fails.

- The second method is to set up a group of pre-allocated trace log data sets (any number of them) that are continually reused. An archive job can be defined to run automatically (log full, log switched, or trace complete) to save the data and mark that log for reuse. Each time the BBI-SS PAS starts, and this trace request is started, the next available log with the oldest data is chosen automatically for output.

This method uses fewer online log data sets. However, if you require archiving, this method may require intervention after any unplanned outage of OS/390 or the BBI-SS PAS, since the archive job on the current trace log cannot run. If you do not require archiving, specify TRREUSE=Y to allow overwriting of a log without it being reset.

Setting Limits for Transaction Trace Views

The parameters shown in Table 5 can be used in BBPARM member IMFBEX00 to set limits for transaction trace (ITA*) views. The default values are underlined.

Table 5. IMFBEX00 Parameters for Trace Transaction Limits

Parameter	Valid Values	Description
TRMAXRD	<u>5000</u> 1 to 99999999	Maximum number of trace transaction records read before query terminates.
TRMAXWR	<u>1000</u> 1 to 99999999	Maximum number of trace transaction records written before query terminates.

Note: To activate changes you make to IMFBEX00 parameters, specify RESET PARM IMFBEX00 or restart the BBI-SS PAS.

Chapter 7. Controlling Workload Thresholds and SMQ Data Collection

This chapter describes how to use parameters in BBPARM member IMFBEX00 to

- control workload thresholds for IMS and IMS Sysplex activity (IMS*) views
- control the collection and refresh rates of shared message queue (SMQ) data in IPSM views that show message statistics

To activate changes you make to IMFBEX00 parameters, specify RESET PARMIMFBEX00 or restart the BBI-SS PAS.

Specifying Thresholds for IMS and IMS Sysplex Activity Views

The parameters shown in Table 6 can be used in BBPARM member IMFBEX00 to control workload thresholds for IMS and IMS Sysplex activity (IMS*) views. The default values are underlined.

Table 6. IMFBEX00 Parameters for IMS and IMS Sysplex Activity Workload Thresholds

Parameter	Valid Values	Description
EXCSINP	<u>0.2</u> 0 to 99.99	Input queue time in seconds. EXCSINP specifies a threshold for excessive transaction input queue time.
EXCSELP	<u>0.5</u> 0 to 99.99	Elapsed time in seconds. EXCSELP specifies a threshold for excessive transaction elapsed time.
EXCSCPU	<u>1</u> 0 to 100	Region CPU time percentage. EXCSCPU specifies a threshold for excessive region CPU time.
EXCSDLI	<u>50</u> 0 to 99999	Region DL/I calls count. EXCSDLI specifies a threshold for excessive DL/I calls.
EXCSSQL	<u>50</u> 0 to 99999	SQL call count. EXCSSQL specifies a threshold for excessive SQL calls.

Controlling Shared Message Queue Data Collection

Several of the workflow, CQS information, and terminal and user views can display shared queue statistics for IMS systems that use shared message queues.

The ISQUERY parameter in BBPARM member IMFBEX00 is used to control how frequently CQS is queried to refresh the IPSM shared queues data tables. You can use the parameter to control the amount of resources used to collect shared message queues data. Views that use the shared queues data tables will display static values in fields that display message count and message age data until expiration of the refresh period defined with the ISQUERY parameter.

The following is a representation of the ISQUERY format:

ISQUERY=(*x*, *y*, *z*)

- The *x* value controls the refresh rate of message age data in the CSQ information views (the ISQ* views).
- The *y* value controls the refresh rate of message count data for shared message queues in non-workflow views.
- The *z* value controls the refresh rate of message count fields for shared message queues in workflow views.

CSQ Information Views with Message Age Fields

The *x* value in ISQUERY=(*x*, *y*, *z*) applies to CSQ information views (ISQ*) that show message age data. The *x* value is used

- to specify whether or not message age data will be collected and displayed for shared message queues
- to define the refresh rate (in seconds) for the data in message age fields if such data is collected

If a refresh rate is specified for the message data, no matter how frequently a CSQ view is refreshed, message ages that are reported in the view will not be refreshed until the specified refresh period expires.

The *x* value can be 0 or from 60 through 7200. The default is 60.

A value of zero in the *x* position prevents the collection of message age data for display in the CSQ views. When the age data is not being collected, the letters SMQ are displayed in message age fields, rather than message age data.

Note: Because frequent refreshing of message age data can consume a significant amount of processing time and can delay IMS and other CQS activity, if you allow collection of the data, you should try to set the *x* value reasonably high. If you are concerned only with messages that are older than an hour, for example, you should consider setting the value as high as 3600.

Non-Workflow Views with Message Count Fields

The y value in $ISQQUERY=(x, y, z)$ applies to non-workflow views that show message count data, such as the CQS information (ISQ*), transaction (ITR*), and user/terminal information (IU*) views. The data in the views is collected on demand in real time. The y value is used to

- specify whether or not message count data will be collected and displayed for shared message queues
- define the refresh rate (in seconds) for the data in message count fields if such data is collected

If a refresh rate is specified for the message data, no matter how frequently a view is refreshed, message counts that are reported in the view will not be refreshed until the specified refresh period expires.

The y value can be 0, 5, 10 or 15. The default is 5.

A value of zero in the y position prevents the collection of message count data for display in non-workflow views. When the message count data is not being collected, the letters SMQ are displayed in message count fields, rather than message count data.

Workflow Views with Message Count Fields

The z value in $ISQQUERY=(x, y, z)$ applies only to workload views. The data for workload views is collected continuously by the workflow sampler, which provides realtime and historical data. The z value is used to

- specify whether or not message count data will be collected and displayed for shared message queues
- define the refresh rate (in seconds) for the data in message count fields if such data is collected

If a refresh rate is specified for the message data, no matter how frequently a view is refreshed, message counts that are reported in the view will not be refreshed until the specified refresh period expires.

The z value can be 0 or 5. The default is 5.

A value of zero in the z position prevents the collection of message count data for display in workflow views. When the message count data is not being collected, the letters SMQ are displayed in message count fields, rather than message count data.

Chapter 8. Using the Event Collector

Event Collector data is written to the IMS log as user record logs X' FA' (transaction records) and X' F9' (program records) for later processing by the offline components, Performance Reporter and Transaction Accountant. The data can be used by user-developed programs or other offline processing systems. Transaction data collected by the Event Collector is also passed to the online Workload Monitor and Workload Analyzer components for workload wait and trace services.

Note: The Resource Analyzer and Resource Monitor components and the MAINVIEW AutoOPERATOR component do not interact with the Event Collector.

Initialization parameters specified in BBPARM member IMFECPO0 affect what the Event Collector does during an IMS session. These parameters control

- configuration, such as SYSID specification
- function, such as the type and amount of data to collect
- recovery, such as action on abend

Parameter options are provided to allow you to limit the data collected by the Event Collector and thereby limit the resources used. These options are described in this chapter.

Starting and Stopping the Event Collector

The Event Collector cannot be dynamically started or stopped. It initializes immediately after IMS initialization is completed and it stays active until IMS terminates.

Specifying Event Collector Options

The Event Collector initialization parameters are specified in BBPARM member IMFECPO0. These parameters are described in the two tables that follow. The parameters specify the data to be collected, the functions to be performed during error recovery, and MVIMS diagnostics. Some of the available collection, recovery, and diagnostic parameters can be CPU-intensive and careful consideration should be given to their selection. The parameters are read and processed by MVIMS at IMS initialization and remain in effect throughout the IMS session.

The tables in this section define each parameter, the data that is collected, and the impact upon CPU usage. The parameters are specified in BBPARM member IMFECPO0.

Table 7 lists Event Collector collection options listed alphabetically by parameter name..

Table 7. Event Collector Collection Options

Parameter	Value	Function, Data Collected, and CPU Usage
BHTO	OFF	<p>Function: Includes buffer handler time (BHT) data in DL/I CPU time data.</p> <p>Note: BHTO=OFF is forced if DBIO option is not BFALTERS.</p>
		<p>Data: Included in DL/I CPU time data.</p> <p>Note: DEDB and MSDB BHT for IMS Fast Path is always included in DL/I CPU time data.</p> <p>Performance Reporter and Transaction Accountant report BHT CPU=0. Performance Reporter, Transaction Accountant, and Workload Analyzer trace services report increased DL/I CPU time ($DL/I \text{ CPU} = DL/I + BHT$), which produces increased chargeable CPU time in Transaction Accountant. Workload Analyzer trace services report increased DL/I CPU time ($DL/I \text{ CPU} = DL/I + BHT$).</p>
		<p>CPU Usage: Most efficient option.</p>
	ON	<p>Function: Collects BHT data separately.</p>
		<p>Data: BHT data is available for Performance Reporter and Transaction Accountant</p>
		<p>CPU Usage: Time that Event Collector adds to IMS usage can be increased 20 to 40 percent; that is, if total MVIMS overhead is 10 percent of IMS CPU, the option could cause the total overhead to be 12 to 14 percent. This increase depends on the percentage of DL/I database calls in the total IMS workload.</p>

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
BILLOVHD	NO	Function: Includes prior transaction termination, current transaction scheduling, program load, and schedule-to-first DL/I as dependent-region-overhead CPU time.
		Data: Performance Reporter, Transaction Accountant, and Workload Analyzer trace services report actual program CPU time, resulting in less chargeable CPU time in Transaction Accountant.
		CPU Usage: Distribution change only.
YES	YES	Function: Includes prior transaction termination, current transaction scheduling, program load, and schedule-to-first DL/I as dependent region-chargeable CPU time.
		Data: Performance Reporter and Transaction Accountant report increased chargeable CPU time in Transaction Accountant.
		Notes: <ul style="list-style-type: none"> • This may cause variations in the usually stable message program CPU time, especially between the first and subsequent transactions processed in a single program scheduling. • Overhead information cannot be collected for DBCTL regions.
CPU Usage: Distribution change only.		
SCHEDDLI	SCHEDDLI	Function: Includes prior transaction termination and current transaction scheduling as <p style="text-align: center;">Dependent region-overhead CPU time</p> Includes program load and schedule-to-first DL/I as <p style="text-align: center;">Dependent region-chargeable CPU time</p>
		Data: Performance Reporter and Transaction Accountant will report increased chargeable CPU time in Transaction Accountant if BI LLOVHD=NO (default) is used and is changed to BI LLOVHD=SCHEDDLI .
		Notes: <ul style="list-style-type: none"> • This may cause variations in the usually stable message program CPU time, especially between the first and subsequent transactions processed in a single program scheduling. • Overhead information cannot be collected for DBCTL regions.
CPU Usage: Distribution change only.		

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
BMP	<u>YES</u>	Function: Collects BMP transaction and program activity data.
		Data: Data available for Performance Reporter, Transaction Accountant, and workload trace.
		CPU Usage: Event Collector CPU usage depends on the number and activity of all BMPs, but usage is higher than if NO or NOCPU is specified.
	NO	Function: Does not measure BMP activity.
		Data: No data available for Performance Reporter, Transaction Accountant, and workload trace.
		CPU Usage: Event Collector CPU usage depends on the number and activity of all BMPs, but usage is lower than if YES or NOCPU is specified.
	NOCPU	Function: Collects BMP transaction and program activity data, but not CPU time usage.
		Data: No CPU data available for BMPs. All other data, such as DL/I counts and database accesses, are still available for Performance Reporter, Transaction Accountant, and workload trace.
		CPU Usage: CPU time measurement, the largest overhead item, is not taken, which enables the collection of other statistics.

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
CICS	<u>YES</u>	Function: Combines ONLINE and OFFLINE functions.
		Data: Data available for offline batch report products, Workload Analyzer wait and trace services, and Workload Monitor services.
		CPU Usage: It depends on the number and activity of CICS transactions and programs.
	ONLINE	Function: Records data for CICS regions and send it to the BBI-SS PAS for processing by Workload Analyzer wait and trace services and Workload Monitor services, but do not write the data to the IMS log.
		Data: Data is available for workload monitors, wait, and trace.
		CPU Usage: It depends on the number and activity of CICS transactions and programs.
	OFFLINE	Function: Records data for CICS regions and write it to the IMS log for batch report processing.
		Data: Data available for offline batch products, such as Performance Reporter and Transaction Accountant, but not for the Workload Analyzer wait or trace services, nor for the Workload Monitor services.
		CPU Usage: It depends on the number and activity of CICS transactions and programs.
	NO	Function: Does not measure CICS region activity data.
		Data: No data available for Performance Reporter or Transaction Accountant, Workload Analyzer wait or trace services, or Workload Monitor services.
		CPU Usage: It depends on the number and activity of CICS transactions and programs.
CPICDB2	<u>TERM</u>	Function: For CPI-C (explicit APPC) conversations, writes a single transaction record for each conversation.
		Note: The CPICDB2 options are in effect for a conversation until the application does an APSB call. When an APSB call is issued, the CPICDLI options are then in effect for the remainder of the application.
		Data: Workload Analyzer, Workload Monitor, Performance Reporter, and Transaction Accountant report one transaction for each CPI-C conversation.
	CPU Usage: No additional CPU.	
	SYNC	Function: For CPI-C (explicit APPC) conversations, writes a transaction record for each SRRCMIT.
		Data: Workload Analyzer, Workload Monitor, Performance Reporter, and Transaction Accountant report one transaction for each CPI-C conversation.
CPU Usage: Minimal increase occurs at sync point.		

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
CPICDLI	<u>APSB</u>	<p>Function: For CPI-C (explicit APPC) conversations, writes a transaction record each time a new PSB is allocated by an APSB call.</p> <p>Note: The CPICDLI options override the CPICDB2 options if and when the application does an APSB call.</p>
		<p>Data: Workload Analyzer, Workload Monitor, Performance Reporter, and Transaction Accountant report one transaction for each APSB.</p>
		<p>CPU Usage: No additional CPU.</p>
	SYNC	<p>Function: For CPI-C (explicit APPC) conversations, writes a transaction record for each SRRCMIT.</p>
		<p>Data: Workload Analyzer, Workload Monitor, Performance Reporter, and Transaction Accountant report one transaction for each sync point.</p>
		<p>CPU Usage: Minimal increase occurs at sync point.</p>

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
CPU	DEPPGM	Function: Collects CPU time data from dependent regions only. It is recorded as application time and includes DL/I and DB2 time. Note: Most chargeable CPU time is still collected.
		Data: CPU fields for CONTROL and DLISAS are zero. Application program CPU contains all dependent region chargeable CPU and message DL/I, buffer, and DB2 CPU contain zeros. This is reflected in all CPU data shown by Performance Reporter, Transaction Accountant, and workload trace.
		CPU Usage: Biggest overhead reduction, since it times the entire transaction as a single event instead of timing each DL/I and SQL call.
	DEPDB2	Function: Collects CPU time data from dependent regions only. It is recorded as application and DB2 time. The DL/I CPU time is included in the CPU time.
		Data: CPU fields for CONTROL and DLISAS are zero. Application program CPU contains all dependent region chargeable CPU except DB2 CPU. Message DL/I and buffer CPU contain zeros. This is reflected in all CPU data shown by Performance Reporter, Transaction Accountant, and workload trace.
		CPU Usage: The potential amount of overhead saved from this option is highly dependent on how many SQL calls the program issues.
	DEP	Function: Collects CPU time data for transaction processing from dependent regions only. It is recorded as application, DL/I, and DB2 time. Note: Most chargeable CPU time is still collected unless LSO=Y or BMPs are run with nonparallel DL/I.
		Data: CPU fields for CONTROL and DLISAS are zero. Performance Reporter and Transaction Accountant scheduling and open/close CPU fields are zero.
		CPU Usage: Usage is less than with ALL, and most of the CPU time data is collected.
	ALL	Function: Collects all CPU time data.
		Data: All CPU time data is available. Reported DL/I CPU time is approximately 5 to 15 percent higher than with CPU=DEP option, depending on the amount of DL/I and DB2 activity and the LSO option.
		CPU Usage: An increase of 3 to 12 percent over CPU=DEP, depending upon the number of DL/I and DB2 message calls.
NONE	Function: Does not collect CPU time data.	
	Data: All CPU time fields in records contain zeroes.	
	CPU Usage: Least usage but greatest data loss.	

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
CPUOVHD	<u>YES</u>	<p>Function: Collects control region overhead and DLISAS region overhead in the MVIMS program log record (X'F9'), even if CPU=ALL is not specified.</p> <p>Note: Control region overhead and DLISAS region overhead are always collected when CPU=ALL is specified; therefore, CPUOVHD=YES has no effect when CPU=ALL is specified. CPUOVHD=YES has no effect on dependent region overhead.</p>
		<p>Data: The overhead CPU fields in the records include control region overhead and DLISAS region overhead values.</p>
		<p>CPU Usage: CPUOVHD does not affect CPU usage.</p>
	REFCPU	<p>Function: Collects overhead CPU values in the MVIMS program log record (X'F9') based on the CPU keyword specification.</p> <p>Data: REFCPU means REFer CPU. What is collected is determined by the use of the CPU and CPUOVHD parameters together:</p> <p>CPUOVHD=REFCPU CPU=NONE</p> <p>Does not collect CPU overhead data.</p> <p>CPUOVHD=REFCPU CPU=ALL</p> <p>Collects all CPU overhead data from control, DLISAS, and dependent regions.</p> <p>CPUOVHD=REFCPU CPU=DEP</p> <p>Collects only CPU overhead data from the dependent region. Do not collect DLISAS or control region CPU overhead data.</p> <p>Note: The CPU overhead values are set to zero when no CPU timing is done. That is, control region overhead CPU is set to zero when CPU=NONE or CPU=DEP. Message region overhead CPU is also set to zero when CPU=NONE.</p>
		<p>CPU Usage: CPUOVHD does not affect CPU usage.</p>
<p>Note: The CPUOVHD parameter applies only to CPU overhead; it does not affect chargeable CPU time.</p>		
DBFP	<u>NO</u>	<p>Function: Collects counts of NONKEY WRITES and NO I/O ALTERS and reports them in the database trailer.</p>
		<p>Data: Counts of NONKEY WRITES and NO I/O ALTERS are collected and are available for performance analysis and billing.</p>
	YES	<p>Function: Bypasses collection of NONKEY WRITES and NO I/O ALTERS.</p>
		<p>Data: NONKEY WRITES and NO I/O ALTERS counts are not reported in the database trailer. This value is recommended if the statistics are not required for performance analysis or billing.</p>
		<p>CPU Usage: Reduces CPU usage for Fast Path transactions.</p>

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
DBIO	<u>IOWAITS</u>	Function: Collects reads for each database; collects writes at the transaction level. Forces BHT0=OFF. NO-I/Os are not collected.
		Data: All database I/O collected at I/O IWAIT. Performance Reporter, Transaction Accountant, and workload trace services I/O counts per transaction are very close to values with BFALTERS option. Performance Reporter NO-I/O counts are zero. Performance Reporter database report shows READs as usual; most WRITEs, which occur during sync point, are collected for the transaction in a special database trailer, ALLDBS, instead of per database.
		CPU Usage: Substantially less than with BFALTERS with minimal loss of data.
	BFALTERS	Function: Collects all I/O data for each database.
		Data: Database I/O and NO-I/O collected in the buffer handler interface (during the DL/I call).
		CPU Usage: Usage can be increased 30 to 40 percent over IOWAITS because of the high ratio of buffer handler activity to DL/I calls. Increase depends on the percentage of DL/I database calls in the total IMS workload.
	NONE	Function: Forces BHT0=OFF. No I/O data is collected.
		Data: DATABASE I/O and NO-I/O fields are zero. Performance Reporter, Transaction Accountant, and Workload Analyzer trace services' database I/O and NO-I/O fields contain zeroes.
		CPU Usage: None
Note: DBIO=IOWAITS and DBIO=NONE do not apply to Fast Path.		
DBTNAME	<u>DB</u>	Function: Collects both DL/I call counts and database I/O call counts at the database level.
		Data: Collects DL/I call counts and I/O call counts at the DBPCB name (database) level. A database trailer (DBT) is created for each DBPCB name that contains DL/I and I/O calls.
		CPU Usage: Minimal usage.
	DD	Function: Collects DL/I call counts at the database level. Collects database I/O counts at the data set level.
		Data: DL/I call counts are collected at the DBPCB name level. I/O call counts are collected at the ddname (data set name) level. A DBT is created for each DBPCB name containing DL/I calls. A DBT is created for each ddname containing I/O calls. This option uses more database trailers than the DB option.
		CPU Usage: Least efficient option.
Note: DBTNAME=DD does not apply to Fast Path.		

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
DBTS	20	Function: Sets the maximum number of database trailers allowed per non-BMP region. Valid values are 2 to 500. The recommended value is 20. Default value is 10.
		Data: If a transaction accesses more than this number of databases, resource data is collected in an overflow trailer named OTHERS.
		CPU Usage: None. Affects ECSA requirement for each region.
DBTS4BMP	30	Function: Sets the maximum number of database trailers allowed per BMP region. Valid values are 2 to 500 (30 is recommended). If a value is not specified, the value specified for DBTS is the default.
		Data: If a transaction accesses more than this number of databases, resource data is collected in an overflow trailer named OTHERS.
		CPU Usage: None. Affects ECSA requirement for each region.

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage	
MSCCLOCK (IMS 6.1 or later)	<u>DEFAULT,</u> <u>NOTSYNC</u>	<p>Function: Specifies whether or not MSC systems are to be considered, by default, synchronized to the clock of the local IMS. Exceptions to the DEFAULT can be specified with MSCCLOCK=<i>xxxx</i>,NOTSYNC or MSCCLOCK=<i>xxxx</i>,SYNC.</p> <p>If a transaction originates from an MSC system that is defined to have a synchronized clock, the Event Collector will convert its arrival date and time (set by the originating MSC system) to local time. The converted time is then stored in the transaction arrival date and time fields of the transaction log record (X'FA'). The MSC systems can be in the same time zone as the local IMS or in a different time zone.</p> <p>If a transaction originates from an MSC system that is not defined as synchronized with the local IMS clock, no conversion is done and the transaction <i>arrival</i> date and time are set to the <i>start</i> date and time on the local IMS.</p> <p>Only one MSCCLOCK=DEFAULT,NOTSYNC DEFAULT,SYNC record is accepted (the first one specified); if another DEFAULT record is encountered, it will be ignored.</p> <p>Note: The MSCCLOCK Event Collector parameter works in conjunction with the monitor TERM keyword operands SYNCLOCK and MSCCLOCK, which are used to qualify workload monitor and trace requests. TERM=SYNCLOCK defines a workload monitor or trace request to include <i>all</i> transactions from systems synchronized to the clock of the local IMS. TERM=MSCCLOCK defines the request to include <i>only nonlocal</i> transactions from MSC systems synchronized to the clock of the local IMS.</p>	
	DEFAULT, SYNC		
			Data: None.
			CPU Usage: None.
	<i>xxxx</i> , NOTSYNC	<p>Function: Used to specify exceptions to the MSCCLOCK DEFAULT setting. The value <i>xxxx</i> represents an MSC ID, which can range from 0001 to 2036. An MSC exception specification is valid whether it precedes or follows the MSCCLOCK=DEFAULT,NOTSYNC SYNC record.</p>	
	<i>xxxx</i> , SYNC		Data: None.
			CPU Usage: None.
SYSID	1	<p>Function: Identifies the MVIMS system for Performance Reporter and Transaction Accountant. Valid values are 1 to 9 or A to Z. Used to identify and select data from other IMS systems.</p>	
		Data: None.	
		CPU Usage: None.	

Table 7. Event Collector Collection Options (continued)

Parameter	Value	Function, Data Collected, and CPU Usage
TELON	NO	Function: Disables specific data collection for the TELON application package.
		Data: None.
		CPU Usage: None.
	YES	Function: Supports the TELON application package by replacing the AGN field in the transaction and program records with the internal TELON transaction name carried in the transaction input message. (For other TELON support options, see “Modification to Support TELON” on page 76.)
		Data: None.
		CPU Usage: None.
TRNSYNC	NO	Function: Does not write a transaction record at BMP checkpoint.
		Data: A record is written only per each successful MESSAGE-GET-UNIQUE or at program end.
		CPU Usage: None.
	YES	Function: Writes a transaction record at BMP checkpoint.
		Data: A record is written.
		CPU Usage: Minimal.

Table 8 lists Event Collector recovery options.

Table 8. Event Collector Recovery Options

Parameter	Value	Function, Data Collected, and CPU Usage
ABCOUNT	02	Function: Specifies the number of abend retries allowed. Valid values are 01 to 99.
BACKOUT	<u>YES</u>	Function: For severe errors, back out Event Collector; do not abend IMS.
	NO	Function: For severe errors abend IMS; do not back out Event Collector.
DEPREC	<u>YES</u>	Function: Performs extended recovery.
		Data: Performs recovery for additional abend conditions.
		CPU Usage: Usage increased 10 to 15 percent, depending on the other selected Event Collector parameter options. This option should be set to NO after MVIMS is thoroughly tested with YES selected.
	NO	Function: Performs basic recovery.
		Data: Some potential abend conditions cannot be recovered.
		CPU Effect: Most efficient. Usage depends on the other selected Event Collector parameter options. This option should be used after MVIMS is thoroughly tested with YES selected.
DUMPS	<u>YES</u>	Function: Takes SVC dumps.
	NO	Function: Does not take SVC dumps; produces a LOGREC only.
RGNIOPT	<u>ABEND</u>	Function: Abends the IMS dependent region if MVIMS initialization fails because of CSA shortage.
		Data: Full recording ensured.
		CPU Usage: None.
	CONTINUE	Function: Continues the IMS dependent region if MVIMS initialization fails.
		Data: No recording is done for that region.
		CPU Usage: None.

Evaluating Data Collection Options

Each data collection option has an effect on the function provided by MVIMS and on Event Collector CPU usage. Event Collector CPU usage is extremely dependent on the workload and configuration characteristics of the IMS system being monitored. This dependency, coupled with the different ways each site uses MVIMS data, make it impossible to summarize all these variables into standard CPU usage estimates or option setting recommendations.

This section provides a short description of the parameters, with an overview of their effect on the data collected and on CPU usage. These descriptions can help you evaluate the options and choose those most suited to your environment.

Note: All CPU usage values are expressed in relation to the overhead that the Event Collector adds to total IMS CPU. For example, if the Event Collector usage is defined as 10 percent and total IMS CPU has a theoretical value of 200, the total IMS + MVIMS usage is 220, as shown in Example #1, below.

Example #1:

IMS = 200
Event Collector = 10% of IMS = $200 * .10 = 20$
Total = IMS + Event Collector = $200 + 20 = 220$

An indicated percentage increase in Event Collector CPU usage (for example, 30 to 40 percent for DBI0=BFALTERS) is relative to the Event Collector usage value (20 in the previous example). Thus, for the previous example, the DBI0=BFALTERS would increase the total CPU usage by a value of 6 to 8 (30 to 40 percent of 20), as shown in Example #2, below.

Example #2:

IMS = 200
Event Collector = 10% of IMS = $200 * .10 = 20$
Total = IMS + Event Collector + BFALTERS = $200 + 20 + 8 = 228$

Reviewing the Initialization Parameters

This section describes how some of the more important IMFECPO0 parameters work. Review these descriptions to determine their usefulness for your site's needs and their CPU usage.

Dependent Overhead CPU

BILLOVHD=NO | YES | SCHEDDLI

The BILLOVHD parameter determines whether dependent region CPU time is treated as overhead or as chargeable CPU time per user when the CPU time is spent in

1. prior transaction termination
2. current transaction scheduling
3. program load, if any
4. schedule-to-first DL/I

If BILLOVHD=NO, CPU time spent on items 1 through 4 is treated as overhead.

If BILLOVHD=YES, CPU time spent on items 1 through 4 is charged to the user.

If BILLOVHD=SCHEDDLI, CPU time spent on items 1 and 2 is treated as overhead and CPU time spent on items 3 and 4 is charged.

DBCTL Threads

CICS=YES | ONLINE | OFFLINE | NO

The CICS parameter controls whether records are collected for DBCTL thread data. The CPU usage is the same for all options. CPU usage depends on the number and activity of CICS transaction programs.

BMP Data

BMP=YES | NO | NOCPU

The BMP parameter controls whether activity data for BMP transactions and programs is collected. If BMP processing is causing bottlenecks in the IMS online system, you may want to avoid the extra overhead that MVIMS monitoring adds. However, this option is viable only if the MVIMS BMP data is not required for accounting or IMS performance analysis. In general, most sites will want to collect BMP data.

The effect of this parameter on MVIMS CPU usage depends on the number and activity of all BMPs.

Buffer Handler Timing

BHTO=OFF | ON

The BHTO parameter controls whether IMS buffer handler activity is included with DL/I CPU or timed separately. The default is to include it with DL/I (BHTO=OFF).

The high ratio of buffer handler calls to application program DL/I calls in IMS makes separate collection of buffer handler CPU very CPU-intensive for MVIMS. The ratio can be as high as 20 to 1, so collecting separate CPU time data for each buffer handler request can become too expensive when compared with the worth of the data. Depending on the number of database calls and the amount of buffer handler activity, BHTO=ON can increase MVIMS CPU usage by 20 to 40 percent.

The BHTO=ON option is provided for product compatibility, but it is not a recommended option.

CPU Data Collection Options

CPU=DEPPGM | DEPDB2 | DEP | ALL | NONE

The CPU parameter controls the level of CPU data collected by the Event Collector in all IMS regions, in just the dependent regions, or in none.

CPU=DEPPGM causes the Event Collector to time only the dependent region activities. It times the entire transaction as a single event and does not time individual DL/I or DB2 calls. The single resulting CPU time (representing all the chargeable time for the transaction) is attributed to application program CPU time. All other chargeable timings are zero. Overhead CPU time, however, is still kept separately.

CPU=DEPPGM offers the biggest overhead reduction, since it times the entire transaction as a single event instead of timing each DL/I and SQL call. However, the amount of overhead saved depends to a large extent on the current transaction processing profiles. For example, a BMP program issuing 10,000 DL/I calls saves more than an MPP program issuing only 10 DL/I calls. However, even when savings from each transaction are small, they add up quickly.

CPU=DEPDB2 functions similarly to DEPPGM except that the Event Collector separates the dependent region DB2/SQL time from the application program CPU time.

CPU=DEPDB2 causes the Event Collector to time the DB2 events (SQL calls). As a result, the potential amount of overhead saved from this option is highly dependent on how many SQL calls the transaction/program issues. For example, if an MPP program issues only two DL/I calls and 100 SQL calls, the amount saved is minimal.

CPU=DEP causes the Event Collector to attribute chargeable CPU application program or DL/I processing CPU time to a specific transaction and user. When CPU=DEP is used, Event Collector CPU usage increases 25 to 35 percent, depending on the amount of DL/I activity, over CPU=NONE.

CPU=ALL adds collection of DL/I processing CPU in the control region and measurement of various overhead categories such as program scheduling activity. It can increase MVIMS CPU usage by 3 to 12 percent over the CPU=DEP option.

CPU=ALL is the best choice if the various overhead CPU categories are needed for performance analysis or if any of the following conditions are true of the monitored IMS:

- The IMS parameter LSO equals Y.
- BMPs are run in nonparallel DL/I mode.
- The percentage of message queue DL/I calls compared to database calls is high. (On average, message queue calls are 5 to 15 percent of the total DL/I calls.)

All of these factors increase the amount of IMS CPU incurred in either the control or DLISAS regions.

Database I/O Options

DBIO=IOWAITS | BFALTERS | NONE

DBIO controls the level of database I/O data to be collected by the Event Collector. The DBIO=BFALTERS option collects all database activity indicators at the database level for each transaction.

With the BFALTERS option, reporting can be made by transaction and user, and by database for extended performance analysis. NO-I/O counts (the number of reads without I/Os), which show buffer handler activity, can also be collected when BFALTERS is selected. BFALTERS uses an IMS buffer handler interface, which is expensive because of the high ratio of requests to the buffer handler compared with DL/I calls and actual I/O.

DBIO=IOWAITS activates a more efficient method of data collection. DL/I calls are collected by database. I/Os are measured at actual occurrence (using the DC Monitor IWAIT interface) instead of in the buffer handler. With the IWAIT interface, reads and writes that occur during call processing are collected by database, but writes that occur at sync point (the majority) can be associated only with the transaction and user, not with the specific database. Most writes are collected at the transaction level and reported under a special database entry ALLDBS. NO-I/O counts are not collected.

IOWAITS provides the same level of data as BFALTERS for accounting and for the transaction, program, and totals levels of I/O analysis. For performance analysis at the database level, DL/I calls, reads, and some writes are still available. The other writes are reported per program.

IOWAITS is the default and recommended option because Event Collector CPU usage is significantly less than with BFALTERS, which can increase MVIMS CPU usage by 30 to 40 percent over the IOWAITS option, depending on the amount of database activity. Using the IOWAITS option increases the MVIMS CPU usage by 5 to 10 percent over DBIO=NONE, depending on the number of database I/Os.

DBIO=NONE specifies that reads, writes, and NO-I/O counts are not collected. DL/I calls are still available by database.

Note: The DBIO parameter does not affect Fast Path databases.

Extended Recovery

DEPREC=YES | NO

DEPREC controls whether recovery from additional abend conditions in dependent regions is enabled and performed as necessary.

MVIMS CPU usage may be increased 10 to 30 percent over the DEPREC=NO option, depending on the options chosen for other parameters (because the more work the Event Collector does, the more overhead added by this option).

The default should remain set until MVIMS is thoroughly tested and stable in each environment. If CPU utilization is still of concern once the other options are chosen, this parameter could then be set to NO for additional savings.

Parameter Option Sets

This section describes three sets of parameter definitions.

Standard Option Set

The standard option set is distributed with the product. These options are defined in BBPARM member IMFECPO0.

```
BMP=YES
CPU=DEPPGM
DBIO=IOWAITS
BHIO=OFF
DEPREC=YES
CICS=YES
```

These standard settings define a level of data collection and CPU usage that is acceptable for most configurations and users. Most I/O and CPU data is available and MVIMS CPU usage is noticeably less than with the full option set.

Note: Additional savings are possible without loss of data if you specify DEPREC=NO. Depending on some of the IMS characteristics described previously, you may want to use CPU=ALL, even though it increases MVIMS CPU usage.

The standard options, either as distributed or with the variations mentioned, generally result in MVIMS CPU usage of 6 to 16 percent of total IMS CPU.

Full Option Set

The full option set defines the maximum level of data collection by the Event Collector. These settings result in the greatest amount of MVIMS CPU usage.

```
BMP=YES
CPU=ALL
DBIO=BFALTERS
BHIO=OFF
DEPREC=YES
CICS=YES
```

More detailed database I/O analysis is possible with the full option set than with the standard set because the full option set collects writes per database and NO I/O counts.

For the CPU parameter, you can substitute the DEPPGM option for the ALL option to conserve CPU usage. Selecting the DEPPGM option is likely to be a one-time decision, although you might decide to use CPU=ALL occasionally for performance analysis.

For the DBIO parameter, the level of data collection provided by the BFALTERS option is rarely used on a daily basis. You can reduce collection overhead by specifying the IOWAITS option instead. You can then specify the BFALTERS option on only those occasions where more data is needed for database analysis.

BHIO=ON is not recommended because of its cost in relation to the value of the data collected (buffer handler CPU time).

You can decrease CPU usage by specifying `DEPREC=NO` without losing any data.

The full option set generally results in MAINVIEW for DBCTL CPU usage of 15 to 25 percent of total IMS CPU, though some individual BMPs with a large amount of DL/I activity can be higher.

Minimum Option Set

The following option settings define the minimum level of data collection by the Event Collector. These settings result in the least amount of MVIMS CPU usage.

```
BMP=NO (optional)
CPU=NONE
DBFP=YES
DBIO=NONE
BHIO=OFF
DEPREC=NO
CICS=NO
```

The minimum option settings allow enough data collection for true transaction-level accounting and performance analysis, while keeping Event Collector CPU usage at a minimum. Records containing all the identifiers (transaction, program, region, and so on) available with MVIMS are still created for each transaction and program. Data collected includes

- All elapsed timings and storage usage
- DL/I call counts per call type
- DL/I call counts per database
- DL/I call counts per LTERM

The minimum option set generally results in MVIMS CPU usage of 3 to 10 percent of total IMS CPU.

How the Option Sets Affect Online Service Data

The following section lists the online services that report one or more of the Event Collector data elements.

The basis for comparison is the level of data collection when the full option set, shown below, is used.

BMP=YES
CPU=ALL
DBI O=BFALTERS
BH TO=OFF
DEPREC=YES
CI CS=YES

Workload Trace (LTRAC, STRAC, and DTRAC)

The following list indicates the effect of specifying values different from those shown above.

- BMP trace
If BMP=NO, BMP transactions and programs cannot be traced.
- LTRAC
If CPU=NONE or CPU=DEPPGM, the DB2TIME parameter is not applicable for the LTRAC service.
If CPU=NONE, CPU=DEPPGM, or CPU=DEPDB2, the DLITIME parameter is not applicable for the LTRAC service.
- STRAC
If CPU=NONE, all CPU data is zero.
- #I/O, AVG #I/O
If DBI O=NONE, the values in these fields are zero.
- DTRAC DL/I CPU times
If CPU=DEPPGM, CPU=DEPDB2, or CPU=NONE, the values in these fields are zero.
- DTRAC DB2 CPU times
If CPU=DEPPGM or CPU=NONE, the values in these fields are zero.
- DBCTL thread tracing
If CI CS=NO or CI CS=OFFLINE, no trace data is collected.
- DBCTL thread monitoring
If CI CS=NO or CI CS=OFFLINE, no data is collected for the Workload Monitor services.

Changing Event Collector Parameters

If you change an Event Collector parameter in BBPARM member IMFECPO0, the change does not take effect until IMS restarts.

Chapter 9. Specifying Target IMS Systems

MAINVIEW for IMS operates in the MAINVIEW environment, which allows a terminal session (TS) to communicate with multiple targets associated with a BBI-SS product address space (PAS). As described in the *MAINVIEW Common Customization Guide*, BBPARM member BBIJNT00 is used to define all eligible target systems and associate them with the subsystem IDs of a BBI-SS PAS.

To define a target IMS system, use the jobname of the IMS region with the TARGET parameter in BBIJNT00.

Chapter 10. Customizing with User Exit Routines

This chapter describes the sample user exit routines provided by BMC Software. You can customize these user exit routines to meet your needs.

Event Collector User Exit Routines

The Event Collector user exit routines are skeleton programs in BBSAMP from which you can build more extensive routines.

Transaction Record User Exit (IMRUTRN)

This routine receives control from the Event Collector just before a transaction record (X' FA') is written. Control is always received in the IMS control region. The captured record can then be evaluated and changed if necessary before control is returned to the Event Collector. When control is returned to the Event Collector, Register 15 (R15) must be zero or the record is not logged. The routine must be logically reentrant.

Documentation about how to activate this routine is in BBSAMP member IMRUTRN3.

See “IMRUTRN and IMRUPGM Cross-Memory Mode Considerations” on page 72 for special considerations about using this exit.

Program Record User Exit (IMRUPGM)

This routine receives control from the Event Collector just before a program record (X' F9') is written. The captured record can then be evaluated and changed if necessary before control is returned to the Event Collector. When control is returned to the Event Collector, R15 must be zero or the record is not logged. The routine must be logically reentrant.

Documentation about how to activate this routine is in BBSAMP member IMRUPGM3.

See “IMRUTRN and IMRUPGM Cross-Memory Mode Considerations” on page 72 for special considerations about using this exit.

IMRUTRN and IMRUPGM Cross-Memory Mode Considerations

IMRUTRN and IMRUPGM user exits can be invoked in cross-memory mode. However, when you invoke these exits from cross-memory mode, SVCs cannot be issued from the exits. Use OS/390 services that can be issued in cross-memory mode with an EUT FRR in effect. For example, instead of GETMAIN, use the STORAGE macro or a branch entry to GETMAIN.

These exits gain control with primary addressability set to the control region. The Event Collector sets an EUT FRR to provide recovery while these exits are in control, regardless of the DEPREC parameter value in BBPARM member IMFECPO0. If the routine abends, a warning message, a LOGREC, and (optionally) a dump are produced. The warning message is an action message that does not scroll off the operator's console.

Abends in these exits are charged against the Event Collector. If the number of abends exceeds the ABCOUNT value specified in IMFECPO0, MVIMS takes one of the following actions.

- If BACKOUT=NO, MVIMS abends the IMS control region.
- If BACKOUT=YES, MVIMS disables the Event Collector.

The BACKOUT parameter, specified in IMFECPO0, has a default of YES (see Table 8 on page 59).

DL/I-CALL-END User Exit (IMRUDLI)

This routine can be used to extract job accounting information from the dependent regions or to extract user activity information from the transaction input message. This information can be especially valuable for BMP accounting and for application generators such as ADF and TELON.

The routine receives control from the Event Collector in the dependent region just after the first DC and the first DB DL/I call completed by IMS for a transaction or program. Therefore, for most transactions the exit is given control twice. This process accommodates programs that may not issue calls of both types (for example, a non-message-driven BMP that issues only DB calls or an MPP that issues only DC calls to perform message switching).

Caution

The transaction record is not in its final format at the time this user exit receives control. Modifications to any fields other than those reserved for the user may be overwritten by later processing.

This routine is loaded into CSA at IMS/MVIMS initialization time and must be logically reentrant.

Documentation about how to activate this routine is in BBSAMP member IMRUDLI3.

Chapter 11. Service Utility Commands

This chapter documents service utility commands, which are used by system programmers to control and maintain service availability.

Lock a Service (LOCK)

The LOCK command can be used to remove a service from general availability. It prevents the use of a specific service until a corresponding ULOCK request is issued. To issue the LOCK command, either

- type LOCK in the SERV field of an application, or
- select a service from a list application with an L line command.

Unlock a Service (ULOCK)

A service may become locked if

- the service abended
- the service was the target of a LOCK command
- the service routine could not be loaded

After the condition is corrected, the ULOCK command can be issued to unlock the service. To issue the ULOCK command, either

- type ULOCK in the SERV field of an application, or
- select a service from a list application with a U line command

Chapter 12. Optional Modifications to MAINVIEW for IMS

This chapter documents optional modifications to MAINVIEW for IMS that are available to the user.

Modification to Support SAP

LINKSAP in the BBSAMP distribution data set is a sample job that supports SAP application programs. This support replaces the IMS transaction code with the SAP identifier in the Event Collector transaction record (X' FA'). The SAP identifier is usually the SAP report ID. If the report ID is not available, it is the SAP transaction code.

This support allows you to analyze your SAP workload using

- Performance Reporter and Transaction Accountant batch reports that provide the SAP identifier instead of the IMS transaction name
- Workload Monitor selection criteria with a service request to limit data collection based on the SAP identifier instead of the IMS transaction name
- the PRSPRINT report which indicates whether the SAP exit was invoked
- the PRSSELEC utility to select SAP-related records

To activate this support, you must

- install SAP 5.0c or later

Contact SAP Corporation of Germany for SAP installation information.

- make a copy of the distributed BBSAMP member LINKSAP and modify and run the JCL as instructed in this member

LINKSAP links a user exit, called SAPEXIT, into the SAP program library. This exit is called by the SAP routines for each transaction to pass the SAP identifier to the Event Collector.

SMP-Applied Modifications

The IBM System Modification Program (SMP) should be used to apply the modifications described below. Using SMP ensures that these modifications are not regressed when a MAINVIEW for IMS service is applied. If a MAINVIEW for IMS service is applied to any of the user-modified CSECTs, the user modifications must be restored by doing an SMP RESTORE and APPLY after the service is applied.

Use the IBM service aid utility AMBLIST to locate the offsets for the entry points in the CSECTs. Sample JCL and a control statement for using the AMBLIST utility are shown below.

```
//          JOB
//          EXEC PGM=AMBLIST
//SYSPRINT DD   SYSOUT=*
//SYSLIB  DD   DSN=&prefix.BBLINK,DISP=SHR   <=== BBLINK LIBRARY
//SYSIN   DD   *
          LISTLOAD OUTPUT=XREF, MEMBER=(ELOAD,IMECSAU1,IMECSAU8,IMECSAU9)
/*
```

Modification to Support TELON

This modification supports the TELON application development package. There are two ways to activate MVIMS support for the TELON application package:

1. Specify TELON=YES in BBPARM member IMFECPO0.

TELON=YES causes the AGN field in the transaction and program records to be replaced with the internal TELON transaction name carried in the transaction input message.

If the first three bytes of the IMS transaction code match the internal TELON transaction code name, the internal TELON transaction code replaces the IMS transaction code and PSB name in the MVIMS log records.

Specifying TELON=YES makes the information available to the IMFLEDIT user exit routines.

2. ZAP the unconditional branches (X' 47F0') at the locations shown in the following table to NOPs (X' 4700'). If more than one release of IMS is in use, SUPERZAP the appropriate CSECT.

Note: If you want only the internal TELON transaction code to be placed in the AGN field and do not want to replace the transaction code and PSB name, take these steps:

- Specify TELON=N0 in BBPARM member IMFECPO0.
- Use the TELON1 ZAP, rather than the TELON2 ZAP.

Load Module	CSECT	Entry Point
IMECSAUx	IMEDL3Ex	TELON1
IMECSAUx	IMELTRNx	TELON2

where x can be the

- 8 suffix for IMS 5.1
- 9 suffix for IMS 6.1
- 1 suffix for IMS 7.1

Note: If the MAINVIEW for IMS modules are already copied to the IMS library (see “Modifying the IMS Control Region JCL” on page 27), the modified version of IMECSAUx must be copied to IMS.

Modifications to Reconcile Response Time Monitoring Data

Certain transactions (such as message switch transactions, which have the same arrival time as the original transaction) can distort the average response times in Workload Analyzer and Workload Monitor monitor data by indicating a large input queue time.

Zero Out Input Queue Time for Selected Transactions

To zero out the input queue time for selected transactions in Workload Analyzer and Workload Monitor *response data only*, ZAP up to 30 entries into the distributed table located at:

Load Module	CSECT	Entry Point
EILOAD	EIPOST	TRANTAB

Note: The table to be zapped is distributed as thirty 8-byte entries containing binary zeros. Each entry is an 8-byte transaction name, including trailing blanks. (CLC instruction with 8-byte length is used.) The table search is terminated either after 30 entries or the first binary zero in the first byte of an entry.

Zero Out Input Queue Time for Message Switch Transactions

To zero out the input queue time for all message switch transactions in Workload Analyzer and Workload Monitor *response data only*, ZAP the unconditional branch (X' 47F0') to NOP (X' 4700') at this location:

Load Module	CSECT	Entry Point
EILOAD	EIPOST	TRANMSW

Part 3. Implementing Product Security

This part describes how users can be authorized to access to MAINVIEW for IMS services.

Chapter 13. Security for Analyzer and Monitor Services	81
Selective Service Lists by User Group	81
Command Authorization	81
Trace Authorization	82

Chapter 13. Security for Analyzer and Monitor Services

Security for analyzer and monitor services is defined both in service tables that you can modify (see the ACCESS keyword in “Service Table Definition” on page 94) and by the user. Users are authorized to access analyzer and monitor services through the PMACC keyword in the user’s BBPARM authorization (USERID) member (see *Implementing Security for MAINVIEW Products*). The security level for each analyzer or monitor service is shown in the analyzer and monitor list applications.

Selective Service Lists by User Group

Service lists selected from the following Service Selection Menu options can be restricted to list only the services for which the user has authority.

- Analyzer display services
- Data collection monitors
- Active timer requests
- General commands

The SERVLIST keyword in BBPARM member BBIISP00 determines whether this feature is activated. The default is SERVLIST=ALL, which means you see all services on the selection list displays. SERVLIST=RESTRICT specifies that you see only those services for which you are authorized. The default service security code is A.

BBPARM member IMFSTD00 is an example of how to set up the service security codes by IMS IMFSTD00 functional area, such as IRLM functions or IMS workload. Each service is assigned a security code according to its area. USERID members can then be created either for groups (such as MTOs or system programmers) or for individuals to access only specific services by listing one or more security codes. If the corresponding security code is not defined in the user’s authorization member, the user does not see those services on the service selection displays.

Command Authorization

Users must be authorized to issue commands or use applications against a BBI-SS PAS target system. Command authorization is defined in user authorization members of the BBPARM data set or by using BBPARM member \$GENERIC. The authorization command is

PMACC Service class authorization, global authorization, or request authorization
(modify or purge).

The ACCESS keyword described on page 96 must match the user’s authorized access code specified for PMACC.

A description of how to use the PMACC command is in *Implementing Security for MAINVIEW Products*. The comment field of the USERID member of the BBPARM data set also describes this command and its valid parameters.

Trace Authorization

The USERID member of the BBPARM member data set can be used to provide user authorization for traces. As described in the *MAINVIEW Common Customization Guide* (see “BBPARM Authorization Member Parameters” in that book), users can have their own members with their user IDs as the member names or the default \$USERID member can be used.

The trace authorization parameters in BBPARM member USERID are TRACE, TRALLOC, and PMACC. TRACE determines whether a user can start a summary or detail trace. TRALLOC authorizes a user for dynamic allocation of trace log data sets by the BBI-SS PAS. PMACC provides trace service access and change authority.

Part 4. Appendixes

This part provides supplemental information for MAINVIEW for IMS use.

Appendix A. IMS Dump Analysis	85
AO Exit Dispatch	85
MAINVIEW AutoOPERATOR for IMS Routines in IMS	85
Event Collector	85
Appendix B. How Product Libraries Should Be Used	87
Appendix C. BBSAMP Data Set Members	89
Sample Members for MAINVIEW for IMS and MAINVIEW for DBCTL	89
Appendix D. Customizing Analyzer and Monitor Services	93
Modifying Service Characteristics	93
Adding a Service	93
Service Table Definition	94
Examples of Parameter Usage	99
Service Logging	99
Appendix E. Product Initialization Messages	101
Messages Changed with Version 3.3.10	101
Messages Added with Version 3.3.00	101
Messages Removed with Version 3.3.00	102
Messages Changed with Version 3.3.00	103

Appendix A. IMS Dump Analysis

This appendix describes how to analyze an IMS dump with MAINVIEW for IMS or MAINVIEW AutoOPERATOR for IMS installed.

AO Exit Dispatch

Except for MAINVIEW AutoOPERATOR for IMS MTO message capture for the Journal log, all AO code merely passes control to the specified routines during operation. During initialization, special protection exists while the Event Collector is being set up.

MAINVIEW AutoOPERATOR for IMS Routines in IMS

During initialization, the MAINVIEW AutoOPERATOR for IMS AO code creates two subtasks under the IMS control task. Each of these is protected by ESTAE routines and uses different control blocks than IMS. These subtasks generally can be ignored during IMS dump analysis since they do not affect the IMS flow. These subtasks are terminated correctly at IMS termination.

Event Collector

The following information should be noted about IMS dumps.

- MAINVIEW for IMS register save areas are in MAINVIEW for IMS data areas, not in the IMS prechained save areas. The IMS chains remain unchanged.
- Sometimes the R14 return register in an IMS save area does not point back into the calling IMS module. The IMS R14 value can be found 4 bytes in front of the address pointed to by R14.
- A MAINVIEW for IMS module at entry saves the registers of an IMS module in the next IMS prechained save area pointed to by R13.
- MAINVIEW for IMS module registers are always saved in MAINVIEW for IMS save areas. These save areas are assigned dynamically as required. Normally, one of several preallocated save areas per region is used.

Each active IMS region has a MAINVIEW for IMS data area acquired for it at region initialization. This data area is in ECSA and is named IMERD nnn , where nnn is the PST number. The preallocated save areas are in this block.

If more save areas are required, a dynamic storage pool is used. This pool is also in ECSA and is named IMFSP000.

Any area in actual use as a MAINVIEW for IMS save area, whether in IMERD nnn or IMFSP000, is identified with SAR or ISA.

- In most cases, only a save area backward pointer exists, pointing from the MAINVIEW for IMS area to the previous IMS save area. While a MAINVIEW for IMS module is in control, the current R12 is its base register and the current R13 points to its SAR.

- In some cases, a MAINVIEW for IMS module transfers control to an IMS module, but needs to regain control after it completes processing. In this case also, the IMS save area chains remain unchanged.

The only sign of the presence of a MAINVIEW for IMS module is an R14 value in an IMS save area which does not point back to the calling IMS module. If you need to verify the IMS path, the IMS R14 value is saved in the MAINVIEW for IMS SAR, 4 bytes in front of the address pointed to by the R14 in the IMS save area.

Note: When MAINVIEW for IMS interfaces between two IMS modules, MAINVIEW for IMS is transparent to the IMS modules. All registers are preserved.

IMECSRvx and IMFCSRvx appear in dumps as active ITASKS. This situation is normal as long as the current save area is for DFSIWAIT.

Appendix B. How Product Libraries Should Be Used

Several distributed libraries are included with your MAINVIEW products, including a parameter library (BBPARM), a sample library (BBSAMP), and a profile library (BBPROF). Use the contents of these distributed libraries as models to create site-customized product libraries, either manually or automatically, with AutoCustomization.

Warning

The distributed libraries should never be modified. If you change the distributed libraries, subsequent SMP maintenance will overwrite your changes.

Throughout the MAINVIEW documentation set, references to these libraries use the distributed name. However, when you need to make changes, be sure to use the corresponding library that has been customized for your site. Table 9 lists the distributed name, the corresponding customized library created by AutoCustomization, and leaves space for you to note any other corresponding library that may have been created for your site.

Table 9. Product Libraries

Distributed Library Name	Library Created by AutoCustomization	Other Site-Customized Copy
BBPARM	UBBPARM	
BBSAMP	UBBSAMP	
BBPROF	SBBPROF	

For detailed information about all the product libraries, see “Using MAINVIEW Product Libraries” in the *MAINVIEW Common Customization Guide* or “Using Product Libraries” in the *MAINVIEW Administration Guide*.

Appendix C. BBSAMP Data Set Members

To help you understand and use your BMC Software product and make it easier to use, the BBSAMP data set contains members that you can edit for your site's use. These members contain macros, sample JCL, sample user exit routines, and sample statements for a variety of functions.

The tables in this appendix provide information about these BBSAMP data set members.

Sample Members for MAINVIEW for IMS and MAINVIEW for DBCTL

Table 10 describes BBSAMP members used for MAINVIEW for IMS and MAINVIEW for DBCTL.

Table 10. BBSAMP Data Set JCL Members

BBSAMP Member Name	Description
ARCHCTL1	Sample SYSIN for the log archive utility (see ARCHJCL) that selects only MAINVIEW for IMS log records.
ARCHCTL2	Sample SYSIN for the log archive utility (see ARCHJCL) that selects all MAINVIEW for IMS log records and some IMS log records.
ARCHJCL	Sample JCL for the IMS log archive utility, which produces an MAINVIEW for IMS user file.
BLKDBTW	Sample starter set of monitors for MAINVIEW for DBCTL.
BLKIMFW	Sample starter set of monitors for MAINVIEW for IMS.
CIMLAR01	Sample COBOL layout for IRUF terminal (LTERM) accounting record (LAR).
CIMPAR01	Sample COBOL layout for IRUF program accounting record (PAR).
CIMTAR01	Sample COBOL layout for IRUF transaction accounting record (TAR) for MVIMS version 3.2 and earlier.
CIMTAR02	Sample COBOL layout for IRUF transaction accounting record (TAR) for MVIMS version 3.3.
CIRUFR01	Macro to map all IRUF records in Assembler for MVIMS version 3.2 and earlier.
CIRUFR02	Macro to map all IRUF records in Assembler for MVIMS version 3.3.
FPORUN	Sample JCL to read the IMS log and create an IRUF with response option and produce several reports.
GTFIMF	Diagnostics. Sample JCL (see GTFUSE).
GTFIMFP	Diagnostics. Sample JCL (see GTFUSE).
GTFIMF00	Diagnostics. Sample parameters (see GTFUSE).

Table 10. BBSAMP Data Set JCL Members

BBSAMP Member Name	Description
GTFUSE	Diagnostics. How to use GTF trace facility for MAINVIEW for IMS Event Collector.
ICOPY n	Sample JCL to copy BMC Software modules from the BBLINK library to a site-authorized library, where n applies to an IMS release.
IMEDBT	Macro to map the database trailer (DBT) for the MAINVIEW for IMS transaction log record in Assembler.
IMEPGM	Macro to map the MAINVIEW for IMS program log record in Assembler.
IMETMEQU	Macro used within IMETR N .
IMETR N	Macro to map the MAINVIEW for IMS transaction log record in Assembler.
IMFACTIV	IMS PR. Sample JCL to create a general activity analysis for all LTERMs and databases.
IMFARB	Macro referenced when assembling some user-written services.
IMFASYDS	Macro referenced when assembling some user-written services.
IMFCLNDR	Performance Reporter. Sample JCL to produce sample calendar reports. One sample of each type of calendar report is produced.
IMFCOSTR	Sample JCL to sort an IRUF into customer ID sequence and summarize it (program TASCOSTR).
IMFFNSUM	Transaction Accountant. Sample JCL to create a financial summary analysis from a costed IRUF.
IMFFPRPT	Sample JCL to create Fast Path transaction processing statistics.
IMFLEDIT	Sample JCL to create an IRUF by editing the IMS system log.
IMFMVSXA	Macro referenced when assembling some user-written services. This macro can be referenced in the BBXS macro library BBMAC, or the BBSAMP member IMFMVSXA can be renamed to BBXMVSXA.
IMFPLOT	Sample JCL to produce X-Y plots of selected variables.
IMFPROG	Performance Reporter. Sample JCL to produce program processing statistics.
IMFREGUT	Performance Reporter. Sample JCL to produce message region utilization analysis.
IMFRESP	Performance Reporter. Sample JCL to produce a response-time distribution report.
IMFRPTS	Sample JCL to read the IMS log, create an IRUF with response option, and produce several reports.
IMFSELEC	Sample JCL to select a subset of IRUF records with which to generate reports.

Table 10. BBSAMP Data Set JCL Members

BBSAMP Member Name	Description
IMFSETAM	Macro referenced when assembling some user-written services. This macro can be referenced in the BBXS macro library BBMAC, or the BBSAMP member IMFSETAM can be renamed to BBXSETAM.
IMFTARC	Sample job to archive a trace log data set.
IMFTRAN	Performance Reporter. Sample JCL to produce transaction processing statistics.
IMFTRLOD	Sample job to restore an archived trace log data set.
IMFTRND2	Performance Reporter. Sample JCL to produce DB2 transaction processing statistics.
IMFTRNFP	Performance Reporter. Sample JCL to produce Fast Path transaction processing statistics.
IMFVT	Macro referenced when assembling some user-written services.
IMRUDLI3	Sample Event Collector user exit routine for DL/I user exit routine CALL/END.
IMRUPGM3	Sample Event Collector user exit routine to access program records for evaluation.
IMRUTRN3	Sample Event Collector user exit routine to access transaction records for evaluation.
JASEXIT	Program to extract IMS charges from a summarized IRUF and create a charge record that can be used as input to the CONTROL/SMF charge-out system.
JRNLMMSG	Macro referenced when assembling some user-written services.
JXT001	Sample job to set up and maintain a trace log directory.
JXT003	Sample job to verify existence of trace log data set in the system catalog.
JXT011	Sample job to define a trace log data set using IDCAMS.
LNKCEXIT	Linkage editor statements to replace default MAINVIEW for IMS Log Edit user exit routine that defines the customer ID with a user-written routine.
LNKLEXIT	Linkage editor statements to replace default MAINVIEW for IMS Log Edit user exit routine that accesses the log file with a user-written routine.
LINKSAP	Sample job to link SAPEXIT into SAP program library.
LOGCPU13	Sample JCL to produce a summary of the CPU times collected by the MAINVIEW for IMS Event Collector from the log records.
LOGREC	Sample JCL to print the software LOGREC records in SYS1.LOGREC.LOGREC

Table 10. BBSAMP Data Set JCL Members

BBSAMP Member Name	Description
PRSCEXIT	Sample Assembler user exit routine to define the customer ID field in the IRUF.
PRSCEXITC	Sample COBOL user exit routine to define the customer ID field in the IRUF.
PRSLEXIT	Sample Assembler user exit routine to access a log record read from the IMS system log.
PRSLEXTC	Sample COBOL user exit routine to access a log record read from the IMS system log.
PRSLEXTA	Sample user exit routine to access a log record read from the IMS system log and write the record to an external file.
PRSPRINT	Sample JCL to select and print IRUF records.
RARGEN	Resource Analyzer. Macro to set global values from defined parameters. These values are then used in macro RARGFN to generate code for Resource Analyzer region displays.
RARGFN	Resource Analyzer. Macro to generate code for functions selected in macro RARGEN.
RAUSR00	Resource Analyzer. Sample Resource Analyzer user analyzer prototype. This prototype establishes the interfaces to MAINVIEW for IMS and to the IMS main control block (SCD), from which most IMS and OS/390 control blocks can be accessed.
RMUSR01	Resource Monitor. Sample Resource Monitor user analyzer prototype. This DL/I Resource Monitor prototype establishes the interfaces to MAINVIEW for IMS and to the IMS main control block (SCD), from which most IMS and OS/390 control blocks can be accessed.
RTOPTCH	Macro to generate a patch area for MAINVIEW for IMS modules. This macro should not be issued more than once in a module.
SASIRUF	SAS definition of IRUF terminal, program, and transaction accounting records.
TACCOSTR	Transaction Accountant. Sample linkage editor statements to link a user exit (TASEXIT) into the Transaction Accountant version of program TASCOSTR.
TASxxxxx	Sample layouts for the IRUF records used in Transaction Accountant.
TASEXIT	Sample user exit routine for TASCOSTR IRUF summarization.
WATBTRAC	Sample batch JCL to print history trace data.

Appendix D. Customizing Analyzer and Monitor Services

MAINVIEW product environment customization is described in the *MAINVIEW Common Customization Guide*. This appendix describes how to tailor MAINVIEW for IMS Online services to your site's needs.

MAINVIEW for IMS Online services have a modular, table-driven design so that you can tailor them to meet specific needs. An example of this would be tailoring the services to provide end-user access security (as shown by the examples on page 99).

Modifying Service Characteristics

MAINVIEW for IMS services are defined in service tables that specify the characteristics of the service. Service characteristics can be modified dynamically by altering the service table entries, or by creating new service table entries. Service tables are located in BBLINK library load modules.

Use BBPARM member IMFSTD00 (Service Table Definition member) to change the IMFSTD00 characteristics of any service or to define new services. Any changes made to IMFSTD00 dynamically modify or create MAINVIEW for IMS services when MAINVIEW for IMS is started.

Note: The two most common characteristics changed by users of MAINVIEW for IMS are the security and title specifications. See “Security for Analyzer and Monitor Services” on page 81 and refer to *Implementing Security for MAINVIEW Products* before you change these specifications.

Adding a Service

To add a service:

1. Code the new service routine by using one of the user service prototypes in the BBSAMP data set as a model.
2. Assemble and link edit the new service routine into BBLINK.
3. Add the definition of the new service to the service table using BBPARM member IMFSTD00.
4. Restart BBI-SS PAS so that the IMFSTD00 member is processed.
5. Test the new service.

Service Table Definition

BBPARM member IMFSTD00 is used to define new services and to change the specifications of existing services. A maximum of 50 new services is allowed.

Note: Message IM2103E is issued if this limit is exceeded.

The following rules apply when creating IMFSTD00:

- The IMFSTD00 must be defined in the BBI-SS PAS BBPARM data set. For more information about the BBI-SS PAS, see the *MAINVIEW Common Customization Guide*.
- All 80 columns of each statement can be used for specifying the various parameters and their values. Sequence numbers can be placed in columns 73 to 80, but there must be at least one blank between the last specification and the sequence number.
- All the parameters needed to define a new service or to modify a given service can be specified either all in one statement or split over multiple statements.
- A specific parameter and its values must be contained in the same specification statement.
- An asterisk can be placed in column 1 of any statement to designate it as a comment. Comment statements can be interspersed with specification statements.
- Commas can be used as delimiters in statements; leading blanks are ignored.
- Comments are allowed within specification statements if one blank separates the specification from the comment.
- If REQUEST is used, it must be the first parameter in a statement and be followed immediately by the SERVICE parameter. If REQUEST is not specified, the SERVICE parameter must be first in the statement (REQUEST defaults to DEFINE).
- If any syntax errors are found in a request to define or modify a service, the accepted parameters up to the error are used to execute a partial change to the service table.

Table 11 lists valid parameters and values for IMFSTD00 and describes the syntax for each parameter. The default for each parameter is underlined.

Table 11. Service Table Parameters and Options

Parameter	Parameter Description	Options
REQUEST	Identifies the start of a new service table entry addition or change. It must be the first parameter in a statement. If REQUEST is not specified, it defaults to <u>DEFINE</u> and SERVICE must be the first parameter (see the examples on page 99).	<p><u>DEFINE</u> Specifies that a new service is to be defined in the MVIMS service table. This new service is listed on the General Commands service display.</p> <p>MODIFY Specifies that an existing service definition is to be changed.</p>
SERVICE	Identifies the name of the service to be added or modified. Note: SERVICE must be the first parameter in a statement if it is used to define a new service instead of using REQUEST (see the example on page 99).	<p>xxxxx Specifies the name of the service to be added or modified. Service names can be from 1 to 5 characters.</p> <p>If REQUEST=<u>DEFI NE</u>, the service name must be unique and not in use at the time of the <u>DEFINE</u>.</p> <p>If REQUEST=<u>MODI FY</u>, the service name must be already defined and exist on one of these lists:</p> <ul style="list-style-type: none"> • Analyzer Display Services • Data Collection Monitors • General Commands <p>Note: The first two to three characters of a user service must not be the same as any of these commands. For example, a user service named ATIV could not be used since MAINVIEW for IMS would transfer to AT, the Active Timer Requests display.</p>
IMSREL RELEASE	Identifies the IMS release for which this service is valid. Note: This parameter is required and must follow the SERVICE specification.	<p>nnnn Specifies the IMS release as</p> <ul style="list-style-type: none"> • 0510 - IMS 5.1 • 0610 - IMS 6.1 • 0710 - IMS 7.1
TITLE	Identifies the title to be given to a service.	<p>x...x Specifies the title. The title can be from 1 to 24 characters. The title must be contained in single quotation marks if the title contains embedded blanks or commas. If REQUEST=<u>DEFI NE</u>, the title is assigned to the new service. If REQUEST=<u>MODI FY</u>, the existing title is changed.</p>

Table 11. Service Table Parameters and Options

Parameter	Parameter Description	Options
ACCESS	Identifies the access code required to use this service.	<u>A</u> Specifies the access code that is to be matched with the user's authorized access code. Can be any alphabetic character, A through Z. The user's authorized access code is specified by the PMACC parameter in BBPARM member USERID.
TYPE	Identifies the service type.	<p><u>ANALYZER</u> Specifies that this is an analyzer service. It can be invoked directly from a terminal and produces a full-screen display. The service may be invoked asynchronously using the SET command to log the display to the image log.</p> <p>MONITOR Specifies that this is a monitor service. It measures the value of one or more system variables at periodic intervals and compares the value to a threshold. The measurements taken by the service may be plotted using PLOT.</p>
CB	Identifies the data requirements of the service.	<p><u>LOCAL</u> Specifies that the service requires no access to data in the IMS control region private area.</p> <p>IMS Specifies that the service requires access to data in the IMS or DBCTL control region private area.</p>
LOG	Identifies the default logging option for analyzer services. It is invalid for monitors. The LOG parameter on a SET request overrides this specification in the service table.	<p>NO Specifies that the display produced by the service should not be logged.</p> <p>SYNC Specifies that the display produced by the service should be logged when it is invoked directly from a terminal.</p> <p>ASYNC Specifies that the display produced by the service should be logged when it is invoked indirectly as a result of a SET service request.</p> <p><u>BOTH</u> Specifies the combination of both SYNC and ASYNC.</p>

Table 11. Service Table Parameters and Options

Parameter	Parameter Description	Options	
MONTYPE	Identifies the type of measurement taken by a monitor. This information is used by the PLOT service and when a monitor warning message is generated.	<u>COUNT</u>	Specifies that the value collected represents the status of the data item when sampled. The value must remain constant or increase with each measurement. It can never decrease.
		AVERAGE	Specifies that the value collected is an average value over an interval.
		STATUS	Specifies that the value collected represents the status of the data item when sampled. The value may decrease, remain constant or increase with each measurement.
		PERCENT	Specifies that the value collected is a percentage of the total of the data item sampled.
		WARNING	Specifies that the monitor does not maintain history information. It is used only to alert the operator of a problem. This type of monitor can never be plotted.
PARAM	Indicates whether the service uses a parameter when invoked.	NO	Specifies that a parameter is never allowed when invoking this service.
		<u>OPT</u>	Specifies that a parameter is optional, if one is provided.
		REQ	Specifies that a parameter is required. The service cannot be invoked without it.

Table 11. Service Table Parameters and Options

Parameter	Parameter Description	Options	
PARMTYPE	Identifies the type of parameter needed by this service. If PARM=NO was specified, PARMTYPE should not be used.	<u>NAME</u>	Specifies that the service uses a parameter in name format, from 1 to 8 characters.
		QNAME	Specifies that the service uses a parameter in name format, from 1 to 8 characters. The name can be qualified by using plus signs (+) as positional qualifiers.
		INTEGER	Specifies that the service uses a parameter in integer format, from 1 to 8 decimal digits.
		HEX	Specifies that the service uses a parameter in hexadecimal format, from 1 to 8 hexadecimal digits.
MODULE	Identifies the load module name of the routine to be invoked by MAINVIEW for IMS when this service is requested.	xxxxxxx	Specifies the load module name. Valid entries are from 1 to 8 characters. The module name defaults to the 1- to 5-character service name specified with the SERVICE parameter.

Examples of Parameter Usage

The following example shows how to use Table 11 parameters to change the security access code of an existing service:

```
REQUEST=MODIFY, SERVICE=DBST, IMSREL=0310, ACCESS=B
```

In another example of Table 11 parameter usage, SERVICE is used to define a new service (REQUEST defaults to DEFINE):

```
SERVICE=NEWSV, IMSREL=0310, TYPE=ANALYZER,  
TITLE=' new service title', ACCESS=A
```

In this example, NEWSV is to be added as a new service.

Service Logging

A service can be recorded to an image log data set synchronously and asynchronously:

- Synchronous logging occurs when a service is requested from a terminal and the service table specifies logging. The type of service logging can also be changed dynamically on the panel when the display is requested (these displays are written to the TS Image log).
- Asynchronous logging occurs as a result of a timer-driven request to log an analyzer service or a general command display, such as logging PLOT or DMON, of monitor measurement values. These displays are written to the BBI-SS PAS Image log.

To change the default logging option for any service, see the LOG parameter definition on page 96. For example, the following statement in BBPARM member IMFSTD00 changes the logging option for the DREGN service:

```
REQUEST=MOD, SERVICE=DREGN, LOG=NO
```


Appendix E. Product Initialization Messages

Product initialization messages changed for versions 3.3.10 and 3.3.00 of MAINVIEW for IMS Online, MAINVIEW for IMS Offline, and MAINVIEW for DBCTL, and new messages were added for version 3.3.00. The message changes and additions are listed in this appendix.

Messages Changed with Version 3.3.10

Table 12 lists the product initialization messages that were changed with version 3.3.10.

Table 12. Product Initialization Messages Changed with Version 3.3.10

Message ID	Message Text	Sub-system
IM1113I	MAINVIEW FOR IMS ONLINE 3.3.10 INITIALIZED PUT (<i>level</i>)	PAS
IM1113I	MAINVIEW FOR IMS DBCTL 3.3.10 INITIALIZED PUT (<i>level</i>)	PAS
IM0109I	MAINVIEW FOR IMS 3.3.10 CICS=YES OPTION ACTIVE	IMS
IM0109I	MAINVIEW FOR DBCTL 3.3.10 CICS=YES OPTION ACTIVE	IMS
IM0113I	MAINVIEW FOR IMS ONLINE 3.3.10 ACTIVE PUT (<i>level</i>)	IMS
IM0113I	MAINVIEW FOR IMS OFFLINE 3.3.10 ACTIVE PUT (<i>level</i>)	IMS
IM0113I	MAINVIEW FOR DBCTL 3.3.10 ACTIVE PUT (<i>level</i>)	IMS

Messages Added with Version 3.3.00

Table 13 lists the product initialization messages that were added with version 3.3.00.

Table 13. Product Initialization Messages Added with Version 3.3.00

Message ID	Message Text	Sub-system
IM0202I	MAINVIEW FOR IMS DD FUNCTION ENABLED IMSID	IMS
IM0202I	MAINVIEW FOR DBCTL DD FUNCTION ENABLED IMSID	IMS
IM1113I	MAINVIEW FOR IMS ONLINE 3.3.0 INITIALIZED SSID (This message is produced if you are using the new ION key.)	PAS
IM1113I	PRODUCT IMF VERSION 3.3.0 INITIALIZED SSID (This message is produced if you are using the old IRA, IRW, IWA, IWM keys.)	PAS
IM0113I	MAINVIEW FOR IMS OFFLINE 3.3.0 ACTIVE	IMS

Note: An IOF initialization key is required for the MAINVIEW for IMS Offline components.

Messages Removed with Version 3.3.00

Table 14 lists the product initialization messages that were removed with version 3.3.00, and lists the new messages that replaced them.

Table 14. Product Initialization Messages Removed with Version 3.3.00

Message ID	Message Text	Replaced with	Subsystem
IM0201I	IMF/EC DD FUNCTION ENABLED IMSI D	IM0109I	IMS
IM0101I	IMF/PR COMPONENT VERSION 3.2.0 ACTIVE	IM0110I	IMS
IM0107I	IMF/PD COMPONENT VERSION 3.2.0 ACTIVE	IM0110I	IMS
IM0102I	IMF/TA COMPONENT VERSION 3.2.0 ACTIVE	IM0110I	IMS
IM0108I	IMF/TD COMPONENT VERSION 3.2.0 ACTIVE	IM0110I	IMS
IM1202I	MAINVIEW FOR DBCTL VERSION 3.2.0 ACTIVE	IM0110I	IMS
IM0131I	IMF FAST PATH OPTION ACTIVE	—	IMS
IM1116I	IMF/RA COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/RM COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/WA COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/WM COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/LM COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/AD COMPONENT ACTIVE	IM1113I	PAS
IM1116I	IMF/MD COMPONENT ACTIVE	IM1113I	PAS
IM1114I	IMF COMPONENT RA KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF COMPONENT RM KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF COMPONENT WA KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF COMPONENT WM KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF EXTENSION LM KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF EXTENSION AD KEY NOT SELECTED	IM1118W	PAS
IM1114I	IMF EXTENSION MD KEY NOT SELECTED	IM1118W	PAS

Note: The IM0131I IMF FAST PATH OPTION ACTIVE message is no longer produced. Versions 3.3.00 and 3.3.10 of MVIMS Online, MVIMS Offline, and MVDBC continue to provide the same Fast Path support provided with the Release 3.2 versions of the products.

Messages Changed with Version 3.3.00

Table 15 lists the MVIMS Online product initialization messages that were changed with version 3.3.00.

Note: The messages identified as “old” in the Key column are produced if you use the old IRA, IRW, IWA, and IWM keys. The messages identified as “new” in the Key column are produced if you use the new ION key.

Table 15. Product Initialization Messages Changed with MVIMS Online Version 3.3.00

Message ID	Key	Message Text	Sub-system
IM0100I	old	IMF/EC COMPONENT VERSION 3.2.0 ACTIVE	IMS
IM0110I	new	MAINVIEW FOR IMS 3.3.0 EVENT COLLECTOR ACTIVE IMSID	IMS
IM0109I	old	CICS=YES IN EFFECT IMSID	IMS
	new	MAINVIEW FOR IMS 3.3.0 CICS=YES OPTION ACTIVE IMSID	IMS
IM0120I	old	- IMF EVENT COLLECTOR NOT STARTED - NO USING COMPONENTS WERE INITIALIZED	IMS
	new	MAINVIEW FOR IMS OR DBCTL FOUND NO VALID PRODUCT KEYS	IMS
IM0109I	old	PRODUCT IMF VERSION 3.2.0 INITIALIZED SSID	PAS
	old	PRODUCT IMF VERSION 3.3.0 INITIALIZED SSID	PAS
	new	MAINVIEW FOR IMS ONLINE 3.3.0 INITIALIZED SSID	PAS
IM1118W	old	WARNING: ALL IMF COMPONENTS FAILED TO ACTIVATE SSID	PAS
	new	WARNING: MAINVIEW FOR IMS FAILED TO ACTIVATE SSID	PAS

Table 16 lists the MVDBC product initialization messages that were changed with version 3.3.00.

Note: The messages identified as “old” in the Key column are produced if you use the old IRA, IRW, IWA, and IWM keys. The messages identified as “new” in the Key column are produced if you use the DBC key.

Table 16. Product Initialization Messages Changed with MVDBC Version 3.3.00

Message ID	Key	Message Text	Sub-system
IM0100I	old	IMF EVENT COLLECTOR ACTIVE - VERSION = MAINVIEW FOR DBCTL 3.2.0	IMS
	new	MAINVIEW FOR DBCTL 3.3.0 EVENT COLLECTOR ACTIVE IMSID	IMS
IM0109I	old	CICS=YES IN EFFECT IMSID	IMS
	new	MAINVIEW FOR DBCTL 3.3.0 CICS=YES OPTION ACTIVE IMSID	IMS
IM0120I	old	- IMF EVENT COLLECTOR NOT STARTED - NO USING COMPONENTS WERE INITIALIZED	IMS
	new	MAINVIEW FOR IMS OR DBCTL FOUND NO VALID PRODUCT KEYS	IMS
IM0109I	old	MAINVIEW FOR DBCTL 3.2.0 INITIALIZED SSID	PAS
	new	MAINVIEW FOR DBCTL 3.3.0 INITIALIZED SSID	PAS
IM1118W	old	WARNING: ALL IMF COMPONENTS FAILED TO ACTIVATE SSID	PAS
	new	WARNING: MAINVIEW FOR IMS FAILED TO ACTIVATE SSID	PAS

Part 5. Glossary and Index

Glossary	107
Index	119

Glossary

This glossary defines BMC Software terminology. Other dictionaries and glossaries can be used in conjunction with this glossary.

Since this glossary pertains to BMC Software-related products, some of the terms defined might not appear in this book.

To help you find the information you need, this glossary uses the following cross-references:

Contrast with	Indicates a term that has a contrary or contradictory meaning.
See	Indicates an entry that is a synonym or contains expanded information.
See also	Indicates an entry that contains related information.

A

action. Defined operation, such as modifying a MAINVIEW window, that is performed in response to a command. *See* object.

active window. Any MAINVIEW window in which data can be refreshed. *See* alternate window, current window, window.

administrative view. Display from which a product's management tasks are performed, such as the DSLIST view for managing historical data sets. *See* view.

ALT WIN field. Input field that allows you to specify the window identifier for an alternate window where the results of a hyperlink are displayed. *See* alternate window.

Alternate Access. *See* MAINVIEW Alternate Access.

alternate form. View requested through the FORM command that changes the format of a previously displayed view to show related information. *See also* form, query.

alternate window. (1) Window that is specifically selected to display the results of a hyperlink. (2) Window whose identifier is defined to the ALT WIN field. *Contrast with* current window. *See* active window, window, ALT WIN field.

analyzer. (1) Online display that presents a snapshot of status and activity data and indicates problem areas. (2) Component of CMF MONITOR. *See* CMF MONITOR Analyzer.

application. (1) Program that performs a specific set of tasks within a MAINVIEW product. (2) In MAINVIEW VistaPoint, combination of workloads to enable display of their transaction performance data in a single view.

application trace. *See* trace.

ASCH workload. Workload comprising Advanced Program-to-Program Communication (APPC) address spaces.

AutoCustomization. Online facility for customizing the installation of products. AutoCustomization provides an ISPF panel interface that both presents customization steps in sequence and provides current status information about the progress of the installation.

automatic screen update. Usage mode wherein the currently displayed screen is refreshed automatically with new data at an interval you specify. Invoked by the ASU command.

B

batch workload. Workload consisting of address spaces running batch jobs.

BBI. Basic architecture that distributes work between workstations and multiple OS/390 targets for BMC Software MAINVIEW products.

BBI-SS PAS. *See* BBI subsystem product address space.

BBI subsystem product address space (BBI-SS PAS). OS/390 subsystem address space that manages communication between local and remote systems and that contains one or more of the following products:

- Command MQ for S/390
- MAINVIEW AutoOPERATOR
- MAINVIEW for CICS
- MAINVIEW for DB2
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for MQSeries
- MAINVIEW SRM
- MAINVIEW VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)

BBPARM. *See* parameter library.

BBPROC. *See* procedure library.

BBPROF. See profile library.

BBSAMP. See sample library.

BBV. See MAINVIEW Alternate Access.

BBXS. BMC Software Subsystem Services. Common set of service routines loaded into common storage and used by several BMC Software MAINVIEW products.

border. Visual indication of the boundaries of a window.

bottleneck analysis. Process of determining which resources have insufficient capacity to provide acceptable service levels and that therefore can cause performance problems.

C

CA-Disk. Data management system by Computer Associates that replaced the DMS product.

CAS. Coordinating address space. One of the address spaces used by the MAINVIEW windows environment architecture. The CAS supplies common services and enables communication between linked systems. Each OS/390 or z/OS image requires a separate CAS. Cross-system communication is established through the CAS using VTAM and XCF communication links.

CFMON. See coupling facility monitoring.

chart. Display format for graphical data. See also graph.

CICSplex. User-defined set of one or more CICS systems that are controlled and managed as a single functional entity.

CMF MONITOR. Comprehensive Management Facility MONITOR. Product that measures and reports on all critical system resources, such as CPU, channel, and device usage; memory, paging, and swapping activity; and workload performance.

CMF MONITOR Analyzer. Batch component of CMF MONITOR that reads the SMF user and 70 series records created by the CMF MONITOR Extractor and/or the RMF Extractor and formats them into printed system performance reports.

CMF MONITOR Extractor. Component of CMF that collects performance statistics for CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390, and RMF postprocessor. See CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390.

CMF MONITOR Online. Component of CMF that uses the MAINVIEW window interface to present data on all address spaces, their use of various system resources, and the delays that each address space incurs while waiting for access to these resources. See CMF MONITOR, MAINVIEW for OS/390.

CMF Type 79 API. Application programming interface, provided by CMF, that provides access to MAINVIEW SMF-type 79 records.

CMFMON. Component of CMF MONITOR that simplifies online retrieval of information about system hardware and application performance and creates MAINVIEW SMF-type 79 records.

The CMFMON *online facility* can be used to view data in one or more formatted screens.

The CMFMON *write facility* can be used to write collected data as MAINVIEW SMF-type 79 records to an SMF or sequential data set.

CMRDETL. MAINVIEW for CICS data set that stores detail transaction records (type 6E) and abend records (type 6D). Detail records are logged for each successful transaction. Abend records are written when an abend occurs. Both records have the same format when stored on CMRDETL.

CMRSTATS. MAINVIEW for CICS data set that stores both CICS operational statistic records, at five-minute intervals, and other records, at intervals defined by parameters specified during customization (using CMRSOPT).

column. Vertical component of a view or display, typically containing fields of the same type of information, that varies by the objects associated in each row.

collection interval. Length of time data is collected. See also delta mode, total mode.

command delimiter. Special character, usually a ; (semicolon), used to stack commands typed concurrently on the COMMAND line for sequential execution.

COMMAND line. Line in the control area of the display screen where primary commands can be typed. Contrast with line command column.

Command MQ Automation D/S. Command MQ agents, which provide local proactive monitoring for both MQSeries and MSMQ (Microsoft message queue manager). The Command MQ agents operate at the local node level where they continue to perform functions regardless of the availability of the MQM (message queue manager) network. Functionality includes automatic monitoring and restarts of channels, queue managers, queues and command servers. In cases where automated recovery is not possible, the agents transport critical alert information to a central console.

Command MQ Automation S/390. Command MQ component, which monitors the MQM (message queue manager) networks and intercedes to perform corrective actions when problems arise. Solutions include:

- Dead-Letter Queue management
- System Queue Archival
- Service Interval Performance solutions
- Channel Availability

These solutions help ensure immediate relief to some of the most pressing MQM operations and performance problems.

Command MQ for D/S. Command MQ for D/S utilizes a true client/server architecture and employs resident agents to provide configuration, administration, performance monitoring and operations management for the MQM (message queue manager) network.

Command MQ for S/390. See MAINVIEW for MQSeries.

COMMON STORAGE MONITOR. Component of MAINVIEW for OS/390 that monitors usage and reconfigures OS/390 or z/OS common storage blocks.

composite workload. Workload made up of a WLM workload or other workloads, which are called *constituent workloads*.

constituent workload. Member of a composite workload. Constituent workloads in a composite usually belong to a single workload class, but sometimes are mixed.

contention. Occurs when there are more requests for service than there are servers available.

context. In a Plex Manager view, field that contains the name of a target or group of targets specified with the CONTEXT command. See scope, service point, SSI context, target context.

CONTEXT command. Specifies either a MAINVIEW product and a specific target for that product (see target context) or a MAINVIEW product and a name representing one or more targets (see *SSI context*) for that product.

control statement. (1) Statement that interrupts a sequence of instructions and transfers control to another part of the program. (2) Statement that names samplers and other parameters that configure the MAINVIEW components to perform specified functions. (3) In CMF MONITOR, statement in a parameter library member used to identify a sampler in the extractor or a report in the analyzer, or to describe either component's processing requirements to the operating system.

coupling facility monitoring (CFMON). Coupling facility views that monitor the activity of your system's coupling facilities.

current data. Data that reflects the system in its current state. The two types of current data are realtime data and interval data. Contrast with historical data. See also interval data and realtime data.

current window. In the MAINVIEW window environment, window where the main dialog with the application takes place. The current window is used as the default window destination for commands issued on the COMMAND line when no window number is specified. Contrast with alternate window. See active window, window.

D

DASD. Direct Access Storage Device. (1) A device with rotating recording surfaces that provides immediate access to stored data. (2) Any device that responds to a DASD program.

data collector. Program that belongs to a MAINVIEW product and that collects data from various sources and stores the data in records used by views. For example, MAINVIEW for OS/390 data collectors obtain data from OS/390 or z/OS services, OS/390 or z/OS control blocks, CMF MONITOR

Extractor control blocks, and other sources. Contrast with extractor.

delta mode. (1) In MAINVIEW for DB2 analyzer displays, difference between the value sampled at the start of the current statistics interval and the value sampled by the current analyzer request. See also *statistics interval*. (2) In CMFMON, usage mode wherein certain columns of data reflect the difference in values between one sample cycle and the next. Invoked by the DELTA ON command. See also collection interval, sample cycle, total mode.

DFSMS. Data Facility Storage Management System. Data management, backup, and HSM software from IBM for OS/390 or z/OS mainframes.

DMR. See MAINVIEW for DB2.

DMS. Data Management System. See CA-Disk.

DMS2HSM. See MAINVIEW SRM DMS2HSM.

DSO. Data Set Optimizer. CMF MONITOR Extractor component that uses CMF MONITOR Extractor data to produce reports specifying the optimal ordering of data sets on moveable head devices.

E

EasyHSM. See MAINVIEW SRM EasyHSM.

EasyPOOL. See MAINVIEW SRM EasyPOOL.

EasySMS. See MAINVIEW SRM EasySMS.

element. (1) Data component of a data collector record, shown in a view as a field. (2) Internal value of a field in a view, used in product functions.

element help. Online help for a field in a view. The preferred term is *field help*.

Enterprise Storage Automation. See MAINVIEW SRM Enterprise Storage Automation.

event. A message issued by Enterprise Storage Automation. User-defined storage occurrences generate events in the form of messages. These events provide an early warning system for storage problems and are routed to user-specified destinations for central viewing and management.

Event Collector. Component for MAINVIEW for IMS Online, MAINVIEW for IMS Offline, and MAINVIEW for DBCTL that collects data about events in the IMS environment. This data is required for Workload Monitor and optional for Workload Analyzer (except for the workload trace service). This data also is recorded as transaction records (X'FA') and program records (X'F9') on the IMS system log for later use by the MAINVIEW for IMS Offline components: Performance Reporter and Transaction Accountant.

expand. Predefined link from one display to a related display. See also hyperlink.

extractor. Program that collects data from various sources and keeps the data control blocks to be written as records. Extractors obtain data from services, control blocks, and other sources. *Contrast with* data collector.

extractor interval. *See* collection interval.

F

fast path. Predefined link between one screen and another. To use the fast path, place the cursor on a single value in a field and press Enter. The resulting screen displays more detailed information about the selected value. *See also* hyperlink.

field. Group of character positions within a screen or report used to type or display specific information.

field help. Online help describing the purpose or contents of a field on a screen. To display field help, place the cursor anywhere in a field and press PF1 (HELP). In some products, field help is accessible from the screen help that is displayed when you press PF1.

filter. Selection criteria used to limit the number of rows displayed in a view. Data that does not meet the selection criteria is not displayed. A filter is composed of an element, an operator, and an operand (a number or character string). Filters can be implemented in view customization, through the PARM/QPARM commands, or through the Where/QWhere commands. Filters are established against elements of data.

fire. The term used to indicate that an event has triggered an action. In MAINVIEW AutoOPERATOR, when a rule selection criteria matches an incoming event and *fires*, the user-specified automation actions are performed. This process is also called *handling* the event.

fixed field. Field that remains stationary at the left margin of a screen that is scrolled either right or left.

FOCAL POINT. MAINVIEW product that displays a summary of key performance indicators across systems, sites, and applications from a single terminal.

form. One of two constituent parts of a view; the other is query. A form defines how the data is presented; a query identifies the data required for the view. *See also* query, view.

full-screen mode. Display of a MAINVIEW product application or service on the entire screen. There is no window information line. *Contrast with* windows mode.

G

global command. Any MAINVIEW window interface command that can affect all windows in the window area of a MAINVIEW display.

graph. Graphical display of data that you select from a MAINVIEW window environment view. *See also* chart.

H

hilevel. For MAINVIEW products, high-level data set qualifier required by a site's naming conventions.

historical data. (1) Data that reflects the system as it existed at the end of a past recording interval or the duration of several intervals. (2) Any data stored in the historical database and retrieved using the TIME command. *Contrast with* current data, interval data and realtime data.

historical database. Collection of performance data written at the end of each installation-defined recording interval and containing up to 100 VSAM clusters. Data is extracted from the historical database with the TIME command. *See* historical data.

historical data set. In MAINVIEW products that display historical data, VSAM cluster file in which data is recorded at regular intervals.

HSM. (Hierarchical Storage Management) Automatic movement of files from hard disk to slower, less-expensive storage media. The typical hierarchy is from magnetic disk to optical disk to tape.

hyperlink. (1) Preset field in a view or an EXPAND line on a display that permits you to

- Access cursor-sensitive help
- Issue commands
- Link to another view or display

The transfer can be either within a single product or to a related display/view in a different BMC Software product. Generally, hyperlinked fields are highlighted. (2) Cursor-activated short path from a topic or term in online help to related information. *See also* fast path.

I

Image log. Collection of screen-display records. Image logs can be created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Image log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Image log stops when both data sets are filled and the first data set is not processed by the archive program.

The TS Image log is a single data set that wraps around when full.

IMSplex System Manager (IPSM). MVIMS Online and MVDBC service that provides Single System Image views of resources and bottlenecks for applications across one or more IMS regions and systems.

interval data. Cumulative data collected during a collection interval. Intervals usually last from 15 to 30 minutes depending on how the recording interval is specified during product customization. *Contrast with* historical data.

Note: If change is made to the workloads, a new interval will be started.

See also current data and realtime data.

InTune. Product for improving application program performance. It monitors the program and provides information used to reduce bottlenecks and delays.

IRUF. IMS Resource Utilization File (IRUF). IRUFs can be either detail (one event, one record) or summarized (more than one event, one record). A detail IRUF is created by processing the IMS system log through a program called IMFLEDIT. A summarized IRUF is created by processing one or more detail IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program.

J

job activity view. Report about address space consumption of resources. See view.

journal. Special-purpose data set that stores the chronological records of operator and system actions.

Journal log. Collection of messages. Journal logs are created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Journal log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Journal log stops when both data sets are filled and the first data set is not being processed by the archive program.

The TS Journal log is a single data set that wraps around when full.

L

line command. Command that you type in the line command column in a view or display. Line commands initiate actions that apply to the data displayed in that particular row.

line command column. Command input column on the left side of a view or display. Contrast with COMMAND line.

Log Edit. In the MAINVIEW for IMS Offline program named IMFLEDIT, function that extracts transaction (X'FA') and program (X'F9') records from the IMS system log. IMFLEDIT also extracts certain records that were recorded on the system log by IMS. IMFLEDIT then formats the records into a file called the IMS Resource Utilization File (IRUF).

M

MAINVIEW. BMC Software integrated systems management architecture.

MAINVIEW Alarm Manager. In conjunction with other MAINVIEW products, notifies you when an exception condition occurs. MAINVIEW Alarm Manager is capable of monitoring multiple systems simultaneously, which means that MAINVIEW Alarm Manager installed on one system keeps track of your entire sysplex. You can then display a single view that show exceptions for all MAINVIEW performance monitors within your OS/390 or z/OS enterprise.

MAINVIEW Alternate Access. Enables MAINVIEW products to be used without TSO by providing access through EXCP and VTAM interfaces.

MAINVIEW Application Program Interface. REXX- or CLIST-based, callable interface that allows MAINVIEW AutoOPERATOR EXECs to access MAINVIEW monitor product view data.

MAINVIEW AutoOPERATOR. Product that uses tools, techniques, and facilities to automate routine operator tasks and provide online performance monitoring, and that achieves high availability through error minimization, improved productivity, and problem prediction and prevention.

MAINVIEW control area. In the MAINVIEW window environment, first three lines at the top of the view containing the window information line and the COMMAND, SCROLL, CURR WIN, and ALT WIN lines. The control area cannot be customized and is part of the information display. Contrast with MAINVIEW display area, MAINVIEW window area.

MAINVIEW display area. See MAINVIEW window area.

MAINVIEW Explorer. Product that provides access to MAINVIEW products from a Web browser running under Windows. MAINVIEW Explorer replaces MAINVIEW Desktop.

MAINVIEW for CICS. Product (formerly MV MANAGER for CICS) that provides realtime application performance analysis and monitoring for CICS system management.

MAINVIEW for DB2. Product (formerly MV MANAGER for DB2) that provides realtime and historical application performance analysis and monitoring for DB2 subsystem management.

MAINVIEW for DBCTL. Product (formerly MV MANAGER for DBCTL) that provides realtime application performance analysis and monitoring for DBCTL management.

MAINVIEW for IMS (MVIMS) Offline. Product with a Performance Reporter component that organizes data and prints reports used to analyze IMS performance and a Transaction Accountant component that produces cost accounting and user charge-back records and reports.

MAINVIEW for IMS (MVIMS) Online. Product that provides realtime application performance analysis and monitoring for IMS management.

MAINVIEW for IP. Product that monitors OS/390 mission-critical application performance as it relates to IP stack usage. Collected data includes: connections, response time statistics, application availability, application throughput, and IP configuration.

MAINVIEW for Linux-Servers. Product that allows you to monitor the performance of your Linux systems from the MAINVIEW windows interface.

MAINVIEW for MQSeries. Delivers comprehensive capabilities for configuration, administration, performance monitoring and operations management for an entire MQM (message queue manager) network.

MAINVIEW for OS/390. System management application (known as MAINVIEW for MVS prior to version 2.5). Built upon the MAINVIEW window environment architecture, it uses the window interface to provide access to system performance data and other functions necessary in the overall management of an enterprise.

MAINVIEW for UNIX System Services. System management application that allows you to monitor the performance of the Unix System Services from a MAINVIEW window interface.

MAINVIEW for VTAM. Product that displays application performance data by application, transaction ID, and LU name. This collected data includes: connections, response time statistics, application availability, and application throughput.

MAINVIEW for WebSphere. Product that provides Web monitoring and management for applications integrated with IBM WebSphere Application Server for OS/390 or z/OS.

MAINVIEW Selection Menu. ISPF selection panel that provides access to all MAINVIEW windows-mode and full-screen mode products.

MAINVIEW SRM. *See* MAINVIEW Storage Resource Manager (SRM).

MAINVIEW SRM DMS2HSM. Product that facilitates the conversion of CA-Disk, formerly known as DMS, to HSM.

MAINVIEW SRM EasyHSM. Product that provides online monitoring and reporting to help storage managers use DFHSM efficiently.

MAINVIEW SRM EasyPOOL. Product that provides control over data set allocation and enforcement of allocation and naming standards. EasyPOOL functions operate at the operating system level to intercept normal job processing, thus providing services without any JCL changes.

MAINVIEW SRM EasySMS. Product that provides tools that aid in the conversion to DFSMS and provides enhancement to the DFSMS environment after implementation. EasySMS consists of the EasyACS functions, the SMSACSTE function, and the Monitoring and Positioning Facility.

MAINVIEW SRM Enterprise Storage Automation. Product that delivers powerful event generation and storage automation technology across the storage enterprise. Used in conjunction with MAINVIEW AutoOPERATOR, automated solutions to perform pool, volume, application, or data set-level manipulation can be created and used in response to any condition or invoked to perform ad hoc requests

MAINVIEW SRM SG-Auto. Product that provides early warning notification of storage anomalies and automated responses to those anomalies based on conditions in the storage subsystem.

MAINVIEW SRM SG-Control. Product that provides real-time monitoring, budgeting, and control of DASD space utilization.

MAINVIEW SRM StopX37/II. Product that provides enhancements to OS/390 or z/OS space management, reducing the incidence of space-related processing problems. The StopX37/II functions operate at the system level to interceptabend conditions or standards violations, thus providing services without any JCL changes.

MAINVIEW SRM StorageGUARD. Product that monitors and reports on DASD consumption and provides historical views to help control current and future DASD usage.

MAINVIEW Storage Resource Manager (SRM). Suite of products that assists in all phases of OS/390 or z/OS storage management. MAINVIEW SRM consists of products that perform automation, reporting, trend analysis, and error correction for storage management.

MAINVIEW SYSPROG Services. *See* SYSPROG Services.

MAINVIEW VistaPoint. Product that provides enterprise-wide views of performance. Application and workload views are available for CICS, DB2, DBCTL, IMS, and OS/390. Data is summarized at the level of detail needed; for example, views can be for a single target, an OS/390 or z/OS image, or an entire enterprise.

MAINVIEW window area. Portion of the information display that is not the control area and in which views are displayed and windows opened. It includes all but the first three lines of the information display. *Contrast with* MAINVIEW control area.

monitor. Online service that measures resources or workloads at user-defined intervals and issues warnings when user-defined thresholds are exceeded.

Multi-Level Automation (MLA). The user-defined, multiple step process in Enterprise Storage Automation that implements solutions in a tiered approach, where solutions are invoked one after another until the condition is resolved.

MVALARM. *See* MAINVIEW Alarm Manager.

MVAPI. *See* MAINVIEW Application Program Interface.

MVCICS. *See* MAINVIEW for CICS.

MVDB2. *See* MAINVIEW for DB2.

MVDBC. *See* MAINVIEW for DBCTL.

MVIMS. *See* MAINVIEW for IMS.

MVLNX. *See* MAINVIEW for Linux-Servers.

MVMQ. *See* MAINVIEW for MQSeries.

MVMVS. *See* MAINVIEW for OS/390.

MVScope. MAINVIEW for OS/390 application that traces both CPU usage down to the CSECT level and I/O usage down to the channel program level.

MVSRM. *See* MAINVIEW Storage Resource Manager (SRM).

MVSRMHSM. *See* MAINVIEW SRM EasyHSM.

MVSRMSGC. *See* MAINVIEW SRM SG-Control.
MVSRMSGD. *See* MAINVIEW SRM StorageGUARD.
MVSRMSGP. *See* MAINVIEW SRM StorageGUARD.
MVVP. *See* MAINVIEW VistaPoint.
MVVTAM. *See* MAINVIEW for VTAM.
MVWEB. *See* MAINVIEW for WebSphere.

N

nested help. Multiple layers of help pop-up windows. Each successive layer is accessed by clicking a hyperlink from the previous layer.

O

object. Anything you can manipulate as a single unit. MAINVIEW objects can be any of the following: product, secondary window, view, row, column, or field.

You can issue an action against an object by issuing a line command in the line command column to the left of the object. *See* action.

OMVS workload. Workload consisting of OS/390 OpenEdition address spaces.

online help. Help information that is accessible online.

OS/390 and z/OS Installer. BMC Software common installation system for mainframe products.

OS/390 product address space (PAS). Address space containing OS/390 or z/OS data collectors, including the CMF MONITOR Extractor. Used by the MAINVIEW for OS/390, MAINVIEW for Unix System Services, and CMF MONITOR products. *See* PAS.

P

parameter library. Data set consisting of members that contain parameters for specific MAINVIEW products or a support component. There can be several versions:

- The distributed parameter library, called BBPARAM
- A site-specific parameter library or libraries

These can be

- A library created by AutoCustomization, called UBBPARAM
- A library created manually, with a unique name

PAS. Product address space. Used by the MAINVIEW products. Contains data collectors and other product functions. *See* OS/390 product address space (PAS), BBI subsystem product address space (BBI-SS PAS).

performance group workload. Collection of address spaced defined to OS/390 or z/OS. If you are running OS/390 or z/OS with WLM in compatibility mode, MAINVIEW for OS/390 creates a performance group workload instead of a service class. *See* service class workload, workload definition.

PERFORMANCE MANAGER. MAINVIEW for CICS online service for monitoring and managing current performance of CICS regions.

Performance Reporter (MVIMS Offline). MVIMS Offline component that organizes data and prints reports that can be used to analyze IMS performance.

Performance Reporter. Product component that generates offline batch reports. The following products can generate these reports:

- MAINVIEW for DB2
- MAINVIEW for CICS

Plex Manager. Product through which cross-system communication, MAINVIEW security, and an SSI context are established and controlled. Plex Manager is shipped with MAINVIEW window environment products as part of the coordinating address space (CAS) and is accessible as a menu option from the MAINVIEW Selection Menu.

PRGP workload. In MVS/SP 5.0 or earlier, or in compatibility mode in MVS/SP 5.1 or later, composite of service classes. MAINVIEW for OS/390 creates a performance group workload for each performance group defined in the current IEAIPS.xx member.

procedure library. Data set consisting of members that contain executable procedures used by MAINVIEW AutoOPERATOR. These procedures are execute command lists (EXECs) that automate site functions. There can be several versions:

- The distributed parameter library, called BBPROC
- A site-specific parameter library or libraries

These can be

- A library created by AutoCustomization, called UBBPROC
- A library created manually, with a unique name

The site-created EXECs can be either user-written or customized MAINVIEW AutoOPERATOR-supplied EXECs from BBPROC.

product address space. *See* PAS.

profile library. Data set consisting of members that contain profile information and cycle refresh definitions for a terminal session connected to a BBI-SS PAS. Other members are dynamically created by MAINVIEW applications. There can be several versions:

- The distributed profile library, called BBPROF
- A site-specific profile library or libraries

These can be

- A library created by AutoCustomization, called SBBPROF
- A library created manually, with a unique name

The site library is a common profile shared by all site users. The terminal session CLIST creates a user profile automatically if one does not exist; it is called userid.BBPROF, where userid is your logon ID. User profile libraries allow each user to specify unique PF keys, CYCLE commands, target system defaults, a Primary Option Menu, and a unique set of application profiles.

Q

query. One of two constituent parts of a view; the other is form. A query defines the data for a view; a form defines the display format. *See also* form, view.

R

realtime data. Performance data as it exists at the moment of inquiry. Realtime data is recorded during the smallest unit of time for data collection. *Contrast with* historical data. *See also* current data and interval data.

Resource Analyzer. Online realtime displays used to analyze IMS resources and determine which are affected by specific workload problems.

Resource Monitor. Online data collection services used to monitor IMS resources and issue warnings when defined utilization thresholds are exceeded.

row. (1) Horizontal component of a view or display comprising all the fields pertaining to a single device, address space, user, etc. (2) Horizontal component of a DB2 table consisting of a sequence of values, one for each column of the table.

RxD2. Product that provides access to DB2 from REXX. It provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more.

S

sample cycle. Time between data samples.

For the CMF MONITOR Extractor, this is the time specified in the extractor control statements (usually 1 to 5 seconds).

For realtime data, the cycle is not fixed. Data is sampled each time you press Enter.

sample library. Data set consisting of members each of which contains one of the following:

- Sample JCL that can be edited to perform specific functions
- A macro that is referenced in the assembly of user-written services
- A sample user exit routine

There can be several versions:

- The distributed sample library, called BBSAMP
- A site-specific sample library or libraries

These can be

- A library created by AutoCustomization, called UBBSAMP
- A library created manually, with a unique name

sampler. Program that monitors a specific aspect of system performance. Includes utilization thresholds used by the Exception Monitor. The CMF MONITOR Extractor contains samplers.

SBBPROF. *See* profile library.

scope. Subset of an SSI context. The scope could be all the data for the context or a subset of data within the context. It is user- or site-defined. *See* SSI context, target.

screen definition. Configuration of one or more views that have been stored with the SAVEScr command and assigned a unique name. A screen includes the layout of the windows and the view, context, system, and product active in each window.

selection view. In MAINVIEW products, view displaying a list of available views.

service class workload. Collection of address spaces defined to OS/390 or z/OS. If you are running Workload Manager (WLM) in goal mode, MAINVIEW for OS/390 creates a service class workload for each service class that you define through WLM definition dialogs.

If you are running MVS 4.3 or earlier, or MVS/SP 5.1 or later with WLM in compatibility mode, MVS creates a performance group workload instead of a service class. *See* performance group workload.

service objective. Workload performance goal, specified in terms of response time for TSO workloads or turnaround time for batch workloads. Performance group workloads can be measured by either objective. Composite workload service objectives consist of user-defined weighting factors assigned to each constituent workload. For compatibility mode, neither OS/390 nor z/OS provides any way to measure service.

service point. Specification, to MAINVIEW, of the services required to enable a specific product. Services can be actions, selectors, or views. Each target (for example, CICS, DB2, or IMS) has its own service point.

The PLEX view lists all the defined service points known to the CAS to which the terminal session is connected.

service request block (SRB). Control block that represents a routine to be dispatched. SRB mode routines generally perform work for the operating system at a high priority. An SRB is similar to a task control block (TCB) in that it identifies a unit of work to the system. *See also* task control block.

service select code. Code entered to invoke analyzers, monitors, and general services. This code is also the name of the individual service.

session. Total period of time an address space has been active. A session begins when monitoring can be performed. If the product address space (PAS) starts after the job, the session starts with the PAS.

SG-Auto. *See* MAINVIEW SRM SG-Auto.

SG-Control. *See* MAINVIEW SRM SG-Control.

single system image (SSI). Feature of the MAINVIEW window environment architecture where you can view and perform actions on multiple OS/390 systems as though they were a single system. The rows of a single tabular view can contain rows from different OS/390 or z/OS images.

Skeleton Tailoring Facility. A facility in MAINVIEW AutoOPERATOR that allows skeleton JCL to be used during job submission. Skeleton JCL can contain variables within the JCL statements to be substituted with data values at job submission time. Directive statements can be used in the skeleton JCL to cause the repetition of a set of skeleton statements. This facility functions similar to the TSO skeleton tailoring facility.

SRB. *See* service request block.

SSI. *See* single system image.

SSI context. Name created to represent one or more targets for a given product. *See* context, target.

started task workload. Address spaces running jobs that were initiated programmatically.

statistics interval. For MAINVIEW for DB2, cumulative count within a predefined interval (30-minute default set by the DB2STATS parameter in the distributed BBPARM member BBIISP00) for an analyzer service DELTA or RATE display. Specifying the DELTA parameter displays the current value as the difference between the value sampled by the current analyzer request and the value sampled at the start of the current interval. Specifying the RATE parameter displays the current value by minute (DELTA divided by the number of elapsed minutes).

stem variables. A REXX facility, supported in MAINVIEW AutoOPERATOR REXX EXECs and the Skeleton Tailoring Facility, where variable names end with a period followed by a

number, such as &POOL.1. This configuration allows each variable to actually represent a table or array of data, with the zero variable containing the number of entries in the array. For example, &POOL.0 = 5 would indicate variables &POOL.1 through &POOL.5 exist.

StopX37/II. *See* MAINVIEW SRM StopX37/II.

StorageGUARD. *See* MAINVIEW SRM StorageGUARD.

summary view. View created from a tabular view using the Summarize option in view customization. A summary view compresses several rows of data into a single row based on the summarize criteria.

SYSPROG services. Component of MAINVIEW for OS/390. Over 100 services that detect, diagnose, and correct OS/390 or z/OS system problems as they occur. Accessible from the OS/390 Performance and Control Main Menu. Note that this component is also available as a stand-alone product MAINVIEW SYSPROG Services.

system resource. *See* object.

T

target. Entity monitored by one or more MAINVIEW products, such as an OS/390 or z/OS image, an IMS or DB2 subsystem, a CICS region, or related workloads across systems. *See* context, scope, SSI context.

target context. Single target/product combination. *See* context.

TASCOSTR. MAINVIEW for IMS Offline program that summarizes detail and summary IMS Resource Utilization Files (IRUFs) to be used as input to the offline components.

task control block (TCB). Address space-specific control block that represents a unit of work that is dispatched in the address space in which it was created. *See also* service request block.

TCB. *See* task control block.

terminal session (TS). Single point of control for MAINVIEW products, allowing data manipulation and data display and providing other terminal user services for MAINVIEW products. The terminal session runs in a user address space (either a TSO address space or a standalone address space for EXCP/VTAM access).

TDIR. *See* trace log directory.

threshold. Specified value used to determine whether the data in a field meets specific criteria.

TLDS. *See* trace log data set.

total mode. Usage mode in CMFMON wherein certain columns of data reflect the cumulative value between collection intervals. Invoked by the DELTA OFF command. *See also* collection interval, delta mode.

trace. (1) Record of a series of events chronologically listed as they occur. (2) Online data collection and display services that track transaction activity through DB2, IMS, or CICS.

trace log data set (TLDS). Single or multiple external VSAM data sets containing summary or detail trace data for later viewing or printing. The trace log(s) can be defined as needed or dynamically allocated by the BBI-SS PAS. Each trace request is assigned its own trace log data set(s).

trace log directory (TDIR). VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information.

transaction. Specific set of input data that initiates a predefined process or job.

Transaction Accountant. MVIMS Offline component that produces cost accounting and user charge-back records and reports.

TS. *See* terminal session.

TSO workload. Workload that consists of address spaces running TSO sessions.

U

UAS. *See* user address space.

UBBPARM. *See* parameter library.

UBBPROC. *See* procedure library.

UBBSAMP. *See* sample library.

user address space. Runs a MAINVIEW terminal session (TS) in TSO, VTAM, or EXCP mode.

User BBPROF. *See* profile library.

V

view. Formatted data within a MAINVIEW window, acquired from a product as a result of a view command or action. A view consists of two parts: query and form. *See also* form, job activity view, query.

view definition. Meaning of data that appears online, including source of data, selection criteria for data field inclusion and placement, data format, summarization, context, product, view name, hyperlink fields, and threshold conditions.

view command. Name of a view that you type on the COMMAND line to display that view.

view command stack. Internal stack of up to 10 queries. For each command, the stack contains the filter parameters, sort order, context, product, and timeframe that accompany the view.

view help. Online help describing the purpose of a view. To display view help, place the cursor on the view name on the window information line and press PF1 (HELP).

W

window. Area of the MAINVIEW screen in which views and resources are presented. A window has visible boundaries and can be smaller than or equal in size to the MAINVIEW window area. *See* active window, alternate window, current window, MAINVIEW window area.

window information line. Top border of a window. Shows the window identifier, the name of the view displayed in the window, the system, the scope, the product reflected by the window, and the timeframe for which the data in the window is relevant. *See also* window status field.

window number. Sequential number assigned by MAINVIEW to each window when it is opened. The window number is the second character in the window status field. *See also* window status field.

window status. One-character letter in the window status field that indicates when a window is ready to receive commands, is busy processing commands, is not to be updated, or contains no data. It also indicates when an error has occurred in a window. The window status is the first character in the window status field. *See also* window information line, window status field.

window status field. Field on the window information line that shows the current status and assigned number of the window. *See also* window number, window status.

windows mode. Display of one or more MAINVIEW product views on a screen that can be divided into a maximum of 20 windows. A window information line defines the top border of each window. *Contrast with* full-screen mode.

WLM workload. In goal mode in MVS/SP 5.1 and later, a composite of service classes. MAINVIEW for OS/390 creates a workload for each WLM workload defined in the active service policy.

workflow. Measure of system activity that indicates how efficiently system resources are serving the jobs in a workload.

workload. (1) Systematic grouping of units of work (e.g., address spaces, CICS transactions, IMS transactions) according to classification criteria established by a system administrator. (2) In OS/390 or z/OS, a group of service classes within a service definition.

workload activity view. Tracks workload activity as the workload accesses system resources. A workload activity view measures workload activity in terms of resource consumption and how well the workload activity meets its service objectives.

Workload Analyzer. Online data collection and display services used to analyze IMS workloads and determine problem causes.

workload definition. Workload created through the WKLIST view. Contains a unique name, a description, an initial status, a current status, and selection criteria by which address spaces are selected for inclusion in the workload. *See* Workload Definition Facility.

Workload Definition Facility. In MAINVIEW for OS/390, WKLIST view and its associated dialogs through which workloads are defined and service objectives set.

workload delay view. Tracks workload performance as the workload accesses system resources. A workload delay view measures any delay a workload experiences as it contends for those resources.

Workload Monitor. Online data collection services used to monitor IMS workloads and issue warnings when defined thresholds are exceeded.

workload objectives. Performance goals for a workload, defined in WKLIST. Objectives can include measures of performance such as response times and batch turnaround times.

Index

A

ABCOUNT parameter 59
abend
 diagnostics 85
ACCESS keyword 81, 96
activating product components 25
adding service 93
ADF 72
ALLDBS 63
AMBLIST utility
 sample JCL 76
 using 76
analyzer
 customizing services 93
 security 81
ANALYZER option 96
AO exit 85
 execution 30
 return code 30
 routines 29
AOEEXIT
 control statements 29, 30
AOIEXIT
 parameter 30
APSB 52
ARCHCTL1 89
ARCHCTL2 89
architecture
 MAINVIEW xi
archiving
 trace log data set 39
ARCHJCL 89
ASYNCR option 96
authorization
 command 81
 trace 82
 user 81
AutoOPERATOR for IMS
 dump analysis 85
AVERAGE option 97

B

BACKOUT parameter 59
BBI
 2.6.0 products 13
 security 81
BBIISP00
 trace directory parameter 37
 trace directory setup 37
BBI-SS PAS 69
BBKEYS 25
BBLINK 27, 93
BBLINK, deleting old modules 20

BBPARAM 47, 81
 changing 87
BBXS 90, 91
BFALTERS 63, 67
BFALTERS option 55, 60, 63, 65
BHTO parameter 48, 62, 65–66
BILLOVHD parameter 49, 61
BLKIMFW 89
BMP
 accounting 72
 activity data 50, 54
 checkpoint 58
 Event Collector option 61
 non-message-driven 72
 parameter 50, 61
 trace 67
BOTH option 96
buffer
 trace
 size 36
buffer handler timing
 Event Collector option 62

C

CB keyword 96
 parameters
 IMS 96
 LOCAL (default) 96
charge-out system 91
CICS
 activity data 51
 parameter 51
CICS parameter
 for the IRUF file 61
CIMLAR01 89
CIMPAR01 89
CIMTAR01 89
CIMTAR02 89
CIRUFR01 89
CIRUFR02 89
COBOL 89
command authorization 81
command processing 29
commands
 general command list 73
 command description 73
common service area 72
 usage 18
compatibility
 IMRUTRN3 exit 71
control
 CPU fields 53
 region overhead 54

- controlling
 - shared message queue data collection 44
- COUNT option 97
- CPI-C
 - event collector parameter 51, 52
- CPICDB2 parameter 51
- CPICDLI parameter 52
- CPU
 - Event Collector option 62
 - parameter 53, 62, 67
 - time data 53
 - usage
 - IMS Event Collector 17
 - MVDBC samplers 18
- CPUOVHD parameter 54
- CSA 72
 - See also* common service area 72
 - usage 18
- CSQ information views 44

D

- database
 - I/O
 - Event Collector option 63
- DBCTL
 - See also* CICS
 - specifying 69
- DBCTL threads
 - Event Collector option 61
- DBFP parameter 54
- DBIO parameter 55
 - BFALTERS 60, 63
- DBTNAME parameter 55
- DBTS parameter 56
- DBTS4BMP parameter 56
- DEDB 48
- defaults
 - trace 34
- DEFINE
 - default 95
 - option 95
 - REQUEST keyword 94
- deleting BBLINK modules 20
- DEP option 53
- DEPDB2 option 53
- dependent overhead CPU
 - Event Collector option 61
- DEPPGM option 53
- DEPPREC parameter 59, 64, 65–66
- detail trace buffers
 - default 36
 - size of 36
- DFSASOE00 29
- diagnostics
 - abend 85
- directory
 - trace log 33

- DL/I
 - CPU time 48, 53, 62
- DL/I-CALL-END user exit 72
 - caution statement 72
- DLISAS
 - IMS CPU incurred 63
- JCL
 - modifying 28
 - region overhead 54
- dumps
 - analysis 85
 - Event Collector 85
 - MAINVIEW AutoOPERATOR for IMS 85
 - AO exit 85
 - IMS 85
 - symptoms 85
- DUMPS parameter 59

E

- EILOAD load module
 - CSECT EIPOST 77
 - TRANMSW entry point 77
 - TRANTAB entry point 77
- Event Collector 47–68
 - CPU usage 47–59, 60, 61
 - debugging 90
 - dump analysis 85
 - options
 - evaluate 60
 - specify 47
 - parameter options 47–59, 61, 68
 - parameters 66
 - SAPEXIT 75
 - starting and stopping 47
 - user exit routine 91
 - using 47
- exit routines 29
 - SAPEXIT 75
- extended recovery
 - Event Collector option 64

F

- FA record
 - user exit 71
- FPORUN 89
- full option set 65
 - BFALTERS option 65

G

- general commands 73
- GTF 89, 90

H

HEX option 98
history traces
 printing 39, 92
 from a batch job (WATBTRAC) 39
 from an online application 40
 from batch job (WATBTRAC) 92

I

ICOPY 90
IDEL31\$5 20
IDEL32\$5 20
IDEL32\$6 20
IDEL33\$5 20
IDEL33\$6 20
IDEL33\$7 20
IELOAD 30
IM2103E
 services exceeded message 94
IMECSAUx load module
 CSECT IMEDL3Ex
 TELON1 76
 CSECT IMELTRNx
 TELON2 76
IMEDBT 90
IMEPGM 90
IMETMEQU 90
IMETRN 90
IMFACTIV 90
IMFARB 90
IMFASYDS 90
IMFBEX00 34
 options for controlling workload thresholds 43
 options for setting transaction trace limitations 42
 trace defaults 34
IMFCLNDR 90
IMFCOSTR 90
IMFECPO0 47
 evaluate data collection options 60
 initialization parameters 47
 standard option set 65
 TELON=YES 76
IMFFNSUM 90
IMFFPRPT 90
IMFLEDIT 76, 90
IMFLOGTR sample member 39
IMFMVSYA 90
IMFPLOT 90
IMFPROG 90
IMFREGUT 90
IMFRESP 90
IMFRPTS 90
IMFSELEC 90
IMFSETAM 91
IMFSTD00 93–94, 99
 creation rules 94
 service security codes 81
IMFSYS00 29, 30

IMFTARC 91
IMFTRAN 91
IMFTRLOD 91
IMFTRLOD sample member 39
IMFTRND2 91
IMFTRNFP 91
IMFVT 91
IMRUDLI3 91
 activate DL/I-CALL-END user exit 72
IMRUPGM
 program record user exit 71
IMRUPGM3 91
IMRUTRN3 71, 91
IMS 96
 dump analysis 85
 log file records
 creating trace data from 39
 MAINVIEW for IMS log record macros 90
IMS option 96
IMS PERFORMANCE REPORTER
 sample JCL 90
IMS Performance Reporter
 sample JCL 91
IMS RA
 See IMS Resource Analyzer
IMS Resource Analyzer
 Event Collector 47
 prototype modules 92
 service logging 99
IMS Resource Monitor
 Event Collector 47
 prototype modules 92
IMS RM
 See IMS Resource Monitor
IMSREL keyword 95
initialization exit 29
initialization parameters
 BHTO= 62
 BILLOVHD= 61
 BMP= 61
 CICS= 61
 CPU= 62
 DBIO= 63
 DEPREC= 64
 reviewing 61
input queue time
 for message switch transactions 77
 for selected transactions 77
INTEGER option 98
IOWAITS option 55, 63, 65
 CPU usage 63
IRUF layouts and batch JCL (BBSAMP) 92
ISQUERY parameter 44

J

JASEXIT 91
JCL
 DLISAS
 modifying 28
 samples 89
JRNLMMSG member 91
JXT001 37
JXT001 member 91
JXT003 38
JXT003 member 91
JXT011 member 91

K

keys, product 25
keywords 94–99
 examples 99
 parameters 94–99

L

LAST SEG LOST 31
LINKSAP 75, 91
LNKCEXIT 91
LNKLEXIT 91
LOCAL option 96
LOG keyword 96
 change default option 99
 parameters
 ASYNCR 96
 BOTH 96
 NO 96
 SYNC 96
LOGCPU13 91
logging
 service display 96
 traces 33
LOGREC 91
LSO
 CPU=ALL 63
 option 53

M

MAINVIEW
 architecture xi
MAINVIEW AutoOPERATOR for IMS
 dump 85
MAINVIEW for IMS
 log edit user exit routine 91
message switch 77
migrating
 from version 3.2 7
 from version 3.3.00 3
minimum option set 66
MODIFY option 95
modifying a service 93
MODULE keyword 98

MONITOR option 96
monitor services
 customizing 93
 security 81
MONTYPE keyword 97
 parameters
 AVERAGE 97
 COUNT (default) 97
 PERCENT 97
 STATUS 97
 WARNING 97
MSCCLOCK parameter 4, 57
MSCCLOCK TERM operand 5, 57
MSDB 48
MSGLVLI keyword
 BOTH 23
 MTO (IMS master terminal) 23
 NONE 23
 WTO (OS/390 system console) 23
MTO message capture 29
MVIMS
 log edit user exit routine 91

N

NAME option 98
NO option 96, 97
NOCPU option 50

O

OFFLINE option 51
OPT option 97
option sets
 full 65
 minimum 66
 online product data 67
 parameter 65
 standard 65

P

parameter option sets 65
parameters
 CPU= 61, 62
 Event Collector initialization
 ABCOUNT 59
 BACKOUT 59
 BHTO 48
 BILLOVHD 49
 BMP 50
 CICS 51
 CPICDB2 51
 CPICDLI 52
 CPU 53
 CPUOVHD 54
 DBFP 54
 DBIO 55
 DBTNAME 55

parameters, Event Collector initialization (continued)

- DBTS 56
- DBTS4BMP 56
- DEPREC 59
- DUMPS 59
- MSCCLOCK 57
- RGNIOPT 59
- SYSID 57
- TELON 58
- TRNSYNC 58

initialization

- BHTO= 62
- BILLOVHD= 61
- BMP= 61
- CICS= 61
- DBIO= 63
- DEPREC= 64

PARM keyword 97

parameters

- NO 97
- OPT 97
- REQ 97

PARMTYPE keyword 98

parameters

- HEX 98
- INTEGER 98
- NAME (default) 98
- QNAME 98

PERCENT option 97

PGM 76

PLOT service 97

PMACC 82

keyword 81

printing

- trace log data sets 39, 92
 - from a batch job (WATBTRAC) 39
 - from an online application 40

product authorization 3, 7

product components

activating 25

product keys 25

program record user exit 71

PRSEXIT 92

PRSEXITC 92

PRSLEXIT 92

PRSLEXTA 92

PRSLEXTC 92

PRSPRINT 92

Q

QNAME option 98

R

RARGEN 92

RARGFN 92

RAUSR00 92

REQ option 97

REQUEST keyword parameters

DEFINE (default) 95

MODIFY 95

response time

modifications 77

restoring

log data set 39

RGNIOPT parameter 59

RMUSR01 92

RTOPTCH 92

S

sample JCL 89

sample JCL and CLISTs (BBSAMP) 92

SAP program 75

SAPEXIT 75, 91

SASIRUF 92

SCHEDDLI option 49

security 82

trace 82

selective service lists

user group 81

SERVICE keyword 94, 95

service logging 99

service table 94–98

service utilities 73

services

adding 93

modifying 93

table definition 94

SERVLIST keyword 81

skeleton tailoring

defined 115

SRRCMIT 51, 52

standard option set 65

STATUS option 97

storage

trace 34

storage requirements 17

SYNC option 96

SYNCLOCK TERM operand 5, 57

syntax notation xiv

SYSID parameter 57

T

TACCOSTR 92

target system 69

TASCOSTR 90, 92

TASEXIT 92

TASxxxx 92

TELON

data collection parameter 58

support options 76

user activity data extraction 72

thresholds

options for controlling 43

- time
 - application 53
 - CPU 53
 - DB2 53
 - DL/I 53
- TITLE keyword 95
- trace
 - archiving 39
 - authorization 82
 - defaults 34
 - detail buffers 36
 - directory 33
 - log data sets
 - printing 39
 - logging 33
 - logging management 39
 - options for setting transaction limits 42
 - printing 92
 - from a batch job (WATBTRAC) 92
 - security 82
 - storage 34
- trace buffers
 - size 36
- trace log data sets
 - printing
 - from a batch job (WATBTRAC) 39
 - from an online application 40
- trace logging
 - from IMS log 39
- TRALLOC 82
- TRANMSW 77
- transaction
 - SAP 75
- transaction record user exit 71
- TRBUFF 36
 - default 36
- TRNSYNC parameter 58
- TRSIZE 36
 - default 36
- TRTIME parameter 35
- TS (terminal session) 69
 - image log 99
- TYPE keyword 96
 - parameters
 - ANALYZER (default) 96
 - MONITOR 96
- type-1 and type-2 AO exit routines 29

U

- user AO exit
 - DFSAOUE0 29
 - DFSAOUE1 29
- user authorization 81
- user exit routines
 - caution statement 72
 - DL/I-CALL-END 72
 - customizing 71
 - DL/I-CALL-END user exit 72

- user exit routines (continued)
 - program record 71
 - SAPEXIT 75
 - transaction record (IMRUTRN3) 71
- USERID
 - authorization 81
 - member of BBPARAM data set 81

W

- WARNING option 97
- WATBPRNT 40
- WATBTRAC 39, 92
- workload thresholds
 - options for controlling 43
- workload trace 67

Y

- YES option 51

STOP!

IMPORTANT INFORMATION - DO NOT INSTALL THIS PRODUCT UNLESS YOU HAVE READ ALL OF THE FOLLOWING MATERIAL

By clicking the YES or ACCEPT button below (when applicable), or by installing and using this Product or by having it installed and used on your behalf, You are taking affirmative action to signify that You are entering into a legal agreement and are agreeing to be bound by its terms, EVEN WITHOUT YOUR SIGNATURE. BMC is willing to license this Product to You ONLY if You are willing to accept all of these terms. CAREFULLY READ THIS AGREEMENT. If You DO NOT AGREE with its terms, DO NOT install or use this Product; press the NO or REJECT button below (when applicable) or promptly contact BMC or your BMC reseller and your money will be refunded if by such time You have already purchased a full-use License.

SOFTWARE LICENSE AGREEMENT FOR BMC PRODUCTS

SCOPE. This is a legally binding Software License Agreement ("**License**") between You (either an individual or an entity) and BMC pertaining to the original computer files (including all computer programs and data stored in such files) contained in the enclosed Media (as defined below) or made accessible to You for electronic delivery, if as a prerequisite to such accessibility You are required to indicate your acceptance of the terms of this License, and all whole or partial copies thereof, including modified copies and portions merged into other programs (collectively, the "**Software**"). "**Documentation**" means the related hard-copy or electronically reproducible technical documents furnished in association with the Software, "**Media**" means the original BMC-supplied physical materials (if any) containing the Software and/or Documentation, "**Product**" means collectively the Media, Software, and Documentation, and all Product updates subsequently provided to You, and "**You**" means the owner or lessee of the hardware on which the Software is installed and/or used. "**BMC**" means BMC Software Distribution, Inc. unless You are located in one of the following regions, in which case "BMC" refers to the following indicated BMC Software, Inc. subsidiary: (i) Europe, Middle East or Africa --BMC Software Distribution, B.V., (ii) Asia/Pacific -- BMC Software Asia Pacific Pte Ltd., (iii) Brazil -- BMC Software do Brazil, or (iv) Japan -- BMC Software K.K. If You enter into a separate, written software license agreement signed by both You and BMC or your authorized BMC reseller granting to you the rights to install and use this Product, then the terms of that separate, signed agreement will apply and this License is void.

FULL-USE LICENSE. Subject to these terms and payment of the applicable license fees, BMC grants You this non-exclusive License to install and use one copy of the Software for your internal use on the number(s) and type(s) of servers or workstations for which You have paid or agreed to pay to BMC or your BMC reseller the appropriate license fee. If your license fee entitles You only to a License having a limited term, then the duration of this License is limited to that term; otherwise this License is perpetual, subject to the termination provisions below.

TRIAL LICENSE. If You have not paid or agreed to pay to BMC or your BMC Reseller the appropriate license fees for a full use license, then, **NOTWITHSTANDING ANYTHING TO THE CONTRARY CONTAINED IN THIS LICENSE:** (i) this License consists of a non-exclusive evaluation license ("Trial License") to use the Product for a limited time ("Trial Period") only for evaluation; (ii) during the Trial Period, You may not use the Software for development, commercial, production, database management or other purposes than those expressly permitted in clause (i) immediately above; and (iii) your use of the Product is on an **AS IS** basis, and **BMC, ITS RESELLERS AND LICENSORS GRANT NO WARRANTIES OR CONDITIONS (INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) TO YOU AND ACCEPT NO LIABILITY WHATSOEVER RESULTING FROM THE USE OF THIS PRODUCT UNDER THIS TRIAL LICENSE.** If You use this Product for other than evaluation purposes or wish to continue using it after the Trial Period, you must purchase a full-use license. When the Trial Period ends, your right to use this Product automatically expires, though in certain cases You may be able to extend the term of the Trial Period by request. Contact BMC or your BMC reseller for details.

TERM AND TERMINATION. This License takes effect on the first to occur of the date of shipment or accessibility to You for electronic delivery, as applicable (the "**Product Effective Date**"). You may terminate this License at any time for any reason by written notice to BMC or your BMC reseller. This License and your right to use the Product will terminate automatically with or without notice by BMC if You fail to comply with any material term of this License. Upon termination, You must erase or destroy all components of the Product including all copies of the Software, and stop using or accessing the Software. Provisions concerning Title and Copyright, Restrictions (or Restricted Rights, if You are a U.S. Government entity) or limiting BMC's liability or responsibility shall survive any such termination.

TITLE AND COPYRIGHT; RESTRICTIONS. All title and copyrights in and to the Product, including but not limited to all modifications thereto, are owned by BMC and/or its affiliates and licensors, and are protected by both United States copyright law and applicable international copyright treaties. You will not claim or assert title to or ownership of the Product. To the extent expressly permitted by applicable law or treaty notwithstanding this limitation, You may copy the Software only for backup or archival purposes, or as an essential step in utilizing the Software, but for no other purpose. You will not remove or alter any copyright or proprietary notice from copies of the Product. You acknowledge that the Product contains valuable trade secrets of BMC and/or its affiliates and licensors. Except in accordance with the terms of this License, You agree (a) not to decompile, disassemble, reverse engineer or otherwise attempt to derive the Software's source code from object code except to the extent expressly permitted by applicable law or treaty despite this limitation; (b) not to sell, rent, lease, license, sublicense, display, modify, time share, outsource or otherwise transfer the Product to, or permit the use of this Product by, any third party; and (c) to use reasonable care and protection to prevent the unauthorized use, copying, publication or dissemination of the Product and BMC confidential information learned from your use of the Product. **You will not export or re-export any Product without both the written consent of BMC and the appropriate U.S. and/or foreign government license(s) or license exception(s).** Any programs, utilities, modules or other software or documentation created, developed, modified or enhanced by or for You using this Product shall likewise be subject to these restrictions. BMC has the right to obtain injunctive relief against any actual or threatened violation of these restrictions, in addition to any other available remedies. Additional restrictions may apply to certain files, programs or data supplied by third parties and embedded in the Product; consult the Product installation instructions or Release Notes for details.

LIMITED WARRANTY AND CONDITION. If You have purchased a Full-Use License, BMC warrants that (i) the Media will be, under normal use, free from physical defects, and (ii) for a period of ninety (90) days from the Product Effective Date, the Product will perform in substantial accordance with the operating specifications contained in the Documentation that is most current at the Product Effective Date. BMC's entire liability and your exclusive remedy under this provision will be for BMC to use reasonable best efforts to remedy defects covered by this warranty

and condition within a reasonable period of time or, at BMC's option, either to replace the defective Product or to refund the amount paid by You to license the use of the Product. BMC and its suppliers do not warrant that the Product will satisfy your requirements, that the operation of the Product will be uninterrupted or error free, or that all software defects can be corrected. This warranty and condition shall not apply if: (i) the Product is not used in accordance with BMC's instructions, (ii) a Product defect has been caused by any of your or a third party's malfunctioning equipment, (iii) any other cause within your control causes the Product to malfunction, or (iv) You have made modifications to the Product not expressly authorized in writing by BMC. No employee, agent or representative of BMC has authority to bind BMC to any oral representations, warranties or conditions concerning the Product. **THIS WARRANTY AND CONDITION IS IN LIEU OF ALL OTHER WARRANTIES AND CONDITIONS. THERE ARE NO OTHER EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS, INCLUDING THOSE OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, REGARDING THIS LICENSE OR ANY PRODUCT LICENSED HEREUNDER. THIS PARAGRAPH SHALL NOT APPLY TO A TRIAL LICENSE.** Additional support and maintenance may be available for an additional charge; contact BMC or your BMC reseller for details.

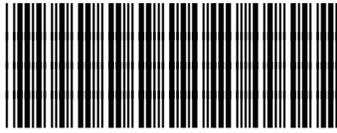
LIMITATION OF LIABILITY. Except as stated in the next succeeding paragraph, BMC's and your BMC reseller's total liability for all damages in connection with this License is limited to the price paid for the License. **IN NO EVENT SHALL BMC BE LIABLE FOR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, PUNITIVE OR INDIRECT DAMAGES OF ANY KIND ARISING OUT OF THE USE OF THIS PRODUCT (SUCH AS LOSS OF PROFITS, GOODWILL, BUSINESS, DATA OR COMPUTER TIME, OR THE COSTS OF RECREATING LOST DATA), EVEN IF BMC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Some jurisdictions do not permit the limitation of consequential damages so the above limitation may not apply.

INDEMNIFICATION FOR INFRINGEMENT. BMC will defend or settle, at its own expense, any claim against You by a third party asserting that your use of the Product within the scope of this License violates such third party's patent, copyright, trademark, trade secret or other proprietary rights, and will indemnify You against any damages finally awarded against You arising out of such claim. However, You must promptly notify BMC in writing after first receiving notice of any such claim, and BMC will have sole control of the defense of any action and all negotiations for its settlement or compromise, with your reasonable assistance. BMC will not be liable for any costs or expenditures incurred by You without BMC's prior written consent. If an order is obtained against your use of the Product by reason of any claimed infringement, or if in BMC's opinion the Product is likely to become the subject of such a claim, BMC will at its option and expense either (i) procure for You the right to continue using the product, or (ii) modify or replace the Product with a compatible, functionally equivalent, non-infringing Product, or (iii) if neither (i) nor (ii) is practicable, issue to You a pro-rata refund of your paid license fee(s) proportionate to the number of months remaining in the 36 month period following the Product Effective Date. This paragraph sets forth your only remedies and the total liability to You of BMC, its resellers and licensors arising out of such claims.

GENERAL. This License is the entire understanding between You and BMC concerning this License and may be modified only in a mutually signed writing between You and BMC. If any part of it is invalid or unenforceable, that part will be construed, limited, modified, or severed so as to eliminate its invalidity or unenforceability. This License will be governed by and interpreted under the laws of the jurisdiction named below, without regard to conflicts of law principles, depending on which BMC Software, Inc. subsidiary is the party to this License: (i) BMC Software Distribution, Inc. - the State of Texas, U.S.A., (ii) BMC Software Distribution, B.V. - The Netherlands, (iii) BMC Software Asia Pacific Pte Ltd. -- Singapore (iv) BMC Software do Brazil -- Brazil, or (v) BMC Software K.K. -- Japan. Any person who accepts or signs changes to the terms of this License promises that they have read and understood these terms, that they have the authority to accept on your behalf and legally obligate You to this License. Under local law and treaties, the restrictions and limitations of this License may not apply to You; You may have other rights and remedies, and be subject to other restrictions and limitations.

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b) or DFARS 227.7202, as amended from time to time. Contractor/Manufacturer is BMC Software, Inc., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

Notes



100042394