

MAINVIEW® SRM User Guide and Reference

Version 6.1

June 1, 2001



Copyright © 1998–2001 BMC Software, Inc., as an unpublished work.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.

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 UNPUBLISHED–RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES.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 Response Online™ (comprehensive information from the Web) or Response On Demand™. To expedite your inquiry, please see “Before Contacting BMC Software,” below.

Response Online

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

Response On Demand

In the USA and Canada, if you need technical support and do not have access to the Web, call 800 538 1872. Outside the USA and Canada, please contact your local support center or your local sales office for assistance.

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
 - messages from related software

Contents

About This Book	xv
------------------------------	-----------

Chapter 1	Introducing MAINVIEW SRM	
	Overview	1-2
	MAINVIEW Architecture	1-2
	MAINVIEW Address Spaces	1-3
	Communication Between Address Spaces	1-3
	MAINVIEW Window Interface	1-4
	Activities Monitored by MAINVIEW SRM	1-4
	Real-Time and Historical Space Information	1-4
	Historical Performance Information	1-5
	Budgeting and Quota Management for Logical Groupings	1-5
	System-Monitored Timeframes	1-5
	Historical Data	1-6
	Displayed Information	1-6
	Understanding the MAINVIEW Window Interface	1-7
	View Categories	1-8
	Tabular and Detail Views	1-8
	Summary Views	1-9
	Customizing Views and Help Text to Meet Your Needs	1-10
	Getting Help on Views	1-11
	Naming Views	1-12
	Moving Around in MAINVIEW SRM	1-13
	Using Hyperlinks	1-13
	Using Menus	1-14
	Using Commands	1-20
	Using MAINVIEW SRM on Multiple Systems	1-21
	Using Single System Image (SSI)	1-21
	Restrictions	1-26
	Accessing Another Target	1-26
	Using the TIME Command	1-27
	Examples of the TIME Command	1-29

Chapter 2	Defining the MAINVIEW SRM System	
	Overview	2-1
	System Configuration Members	2-2
	Parmlib Member Descriptions	2-3
	MAINVIEW SRM Parameters	2-5
	Action Parameters	2-6
	Selection Parameters	2-6
	Defining the Sysplex Environment	2-10
	Sysplex Parameters	2-11
	Syntax Considerations for the Shared Parmlib Environment	2-11
	Ordering of Selection Criteria	2-12
Chapter 3	Understanding General Syntax Rules	
	Statements and Parameters	3-1
	Parameter Relationships	3-3
	Comparison Operators	3-4
	Name Masking	3-5
	Variables in Name Masks	3-6
	Field-to-Field Comparisons	3-9
	Selection Criteria Ordering	3-9
Chapter 4	Using MAINVIEW SRM Operator Services	
	Overview	4-2
	Activating System Software in OS/390	4-2
	Communicating with MAINVIEW SRM Components	4-3
	Common Command Structure	4-3
	Controlling MAINVIEW SRM Components	4-3
	Managing the Storage Product Address Space	4-4
	Starting and Stopping MAINVIEW SRM	4-4
	Starting SVOS	4-4
	Starting MAINVIEW SRM Components	4-5
	Stopping MAINVIEW SRM Components	4-7
	Stopping SVOS	4-8
	Starting and Stopping MAINVIEW SRM with Other Products	4-8
	SVOS Command Syntax Conventions	4-9
	Entering Commands	4-9
	Entering Commands from the PARMLIB DD Statement	4-10
	SVOS Command Definitions	4-11
	DUMP Command	4-11
	FORCE Command	4-12
	SHUTDOWN Command	4-12
	START Command	4-13
	STATUS Command	4-14
	STOP Command	4-15
	SVOS Execution JCL Description	4-16
	Parameter Descriptions	4-16
	The SVOS Console Interface	4-18
	START/STOP Commands	4-19

DISPLAY Commands	4-19
REFRESH Commands	4-20
SET Commands	4-21
VSCAN command	4-22

Chapter 5 Using MAINVIEW SRM Easy Menu

Overview	5-2
SRM Real Time Monitor	5-4
Pools	5-4
SMS Storage Groups	5-5
SMS Pools	5-6
RAID Configurations	5-8
Storage Performance	5-9
SRM Historical Data	5-11
Historical Space	5-11
Historical Performance	5-13
EasyHSM	5-15
SG-Control	5-16
SMF Report Library	5-17
SRM Administration	5-18
Parmilb Members	5-18
Functions	5-21
SRM Component Status	5-22
SRM Tools and Menu	5-24
Workbench	5-24
MVS SRM View List	5-26
MVS SRM Batch Reports	5-27
MAINVIEW Messages	5-28

Chapter 6 Defining and Activating Functions

Overview	6-1
Using the SMFUNCxx Member	6-2
Active and Inactive Functions	6-2
Function Parameters	6-2
Function Parameter Quick Reference	6-3
Using the Functions Option	6-5
Defining the USERVARS Function	6-6

Chapter 7 Defining a Pool

Overview	7-1
Using the SMPOOLxx Member	7-2
Parameters	7-2
Defining and Monitoring Pools	7-5

Chapter 8 Using Variables

Overview	8-1
Using the SMVARSxx Member	8-1
Using the Variables Option	8-4

Chapter 9	Defining a Calendar	
	Overview	9-1
	Using the SMCALSxx Member	9-1
	Parameters.	9-2
	Using the Calendars Option	9-3
Chapter 10	Using Diagnostics	
	Overview	10-1
	Using the SMDIAGxx Member	10-1
	Parameters.	10-2
	Using the Diagnostics Option	10-3
	Using the Tracing Facility	10-4
	Activating Tracing	10-4
	How Function Tracing Works.	10-6
	Additional Trace Options	10-9
Chapter 11	Using Filter and Rule List Parameters	
	Overview	11-1
	Using SMFLSTxx - Filter List Parameters	11-2
	SET Statement Parameters	11-2
	INC/EXC Parameters	11-5
	Using SMRLSTxx - Rule List Parameters	11-6
	Using the Filter List and Rule List Options	11-9
Chapter 12	Monitoring Pools and SMS Storage Groups	
	Overview	12-2
	Monitor Pool Views.	12-3
	Monitor Volume Space Views.	12-4
	Monitor Volumes in a Pool View	12-5
	Monitor SMS Storage Group Views	12-6
	Monitor SMS Storage Group Volume Space View	12-6
	Monitor SMS Pool Views	12-7
	Monitor Performance Views	12-7
	Monitor Volume Performance Summary Views	12-9
	Monitor Volume Performance Statistics View	12-10
Chapter 13	Processing SMF Records	
	Overview	13-1
	Using the PRSMF000 Utility	13-2
	SMF Record Layout	13-2
Chapter 14	Generating and Managing Batch Reports	
	Overview	14-1
	Initiating Report JCL Generation	14-3
	Setting Up the MAINVIEW Batch Environment	14-5
	Generating MAINVIEW Batch Reports JCL	14-7
	Managing MAINVIEW Batch Report JCL Members.	14-9
	Batch Report Output Members.	14-10

Appendix A

Altering Messages with the Message Exit

Appendix B

Global Parameters

Master System Member Parameters B-1
Usage Notes B-14
Pool Member Parameters B-69
SMS Subpool Member Parameters B-73
Calendar Member Parameters B-74
Variable Member Parameters B-76
Function Member Parameters B-77
Diagnostic Member Parameters B-81
Event Member Parameters B-82
VTOC Scan Facility Parameters B-86

Glossary

Index

List of Figures

Figure 1-1	Communication Flow between SRM Address Spaces	1-3
Figure 1-2	MPOOLVOL View	1-7
Figure 1-3	Online Help Available for MAINVIEW Products	1-11
Figure 1-4	EZSRM Menu	1-15
Figure 1-5	EZSRMSGD Menu	1-16
Figure 1-6	Catalog Super Locate Data Filter Panel	1-17
Figure 1-7	EZSRM Menu, Storage Performance Pop-Up Menu	1-18
Figure 1-8	EZcmd Menu Example	1-19
Figure 1-9	MDEV View with CONtext ALL Specified	1-24
Figure 1-10	MDEVSP for Device 853C	1-25
Figure 1-11	Error Message if a System in Your Context Goes Down	1-26
Figure 1-12	SET TIME FRAME Panel	1-27
Figure 2-1	MAINVIEW SRM Definition Members	2-2
Figure 2-2	Connecting MAINVIEW SRM Definition Member Parameters	2-5
Figure 5-1	EZMSRM Menu	5-2
Figure 5-2	MPOOL View	5-4
Figure 5-3	MSMSG View	5-5
Figure 5-4	MSMSP View	5-6
Figure 5-5	EZSRM Menu, RAID Devices Menu	5-8
Figure 5-6	EZSRM Menu, Storage Performance Menu	5-9
Figure 5-7	EZSRMSGD Menu	5-11
Figure 5-8	EZSRMSGP Menu	5-13
Figure 5-9	EZSRMHSM Menu	5-15
Figure 5-10	EZSRMSGC Menu	5-16
Figure 5-11	SMFRPTD View	5-17
Figure 5-12	EZSRM Menu, Parmlib Members Menu	5-18
Figure 5-13	ADFUNC View	5-21
Figure 5-14	SRM Component Status	5-23
Figure 5-15	Workbench Menu	5-24
Figure 5-16	MVSRM View List	5-26
Figure 5-17	MVSRM Batch Reports	5-27
Figure 5-18	MAINVIEW Messages	5-28
Figure 7-1	Example Storage Configuration for Pools	7-4
Figure 10-1	Tracing Example	10-6

Figure 12-1	EZSRM Menu	12-2
Figure 13-1	SMF Record Layout (Part 1 of 5)	13-3
Figure 14-1	MAINVIEW Batch Reports Panel	14-4
Figure 14-2	MAINVIEW Batch Environment Setup Panel	14-5
Figure 14-3	Generate MAINVIEW Batch Reports JCL Panel	14-7
Figure 14-4	MAINVIEW Batch JCL Member List Panel	14-9
Figure 14-5	MAINVIEW Batch Report List Panel	14-10
Figure 14-6	MAINVIEW Batch Report, MCHAN	14-11

List of Tables

Table 1-1	Online Help Available for MAINVIEW Products	1-11
Table 3-1	Name Masking Symbols	3-5
Table 4-1	SVOS Command Table	4-11
Table 4-2	Command Syntax for the DISPLAY command	4-19
Table 4-3	Command Syntax for the REFRESH command	4-20
Table 4-4	Command Syntax for the SET command	4-22
Table 5-1	Parmlib Member Administration	5-18
Table 6-1	SET Statement Function Parameters	6-3
Table 6-2	SET Statement Parameters for USERVARS Function	6-6
Table 7-1	SET Statement Pool Parameter Quick Reference	7-3
Table 7-2	INC/EXC Statement Pool Parameter Quick Reference	7-3
Table 8-1	SET Statement Variable Parameters	8-2
Table 8-2	INC/EXC Statement Variable Parameters	8-2
Table 9-1	SET Statement Calendar Parameters	9-2
Table 9-2	INC/EXC Statement Calendar Parameters	9-2
Table 10-1	SET Statement Diagnostic Parameters	10-2
Table 10-2	INC/EXC Statement Diagnostic Parameters	10-3
Table 11-1	SET Statement Parameter Quick Reference	11-2
Table 2-1	SET Statement System Parameters	B-2
Table 2-2	INC/EXC Statement Parameter Quick Reference for SMMSYSxx	B-14
Table 2-3	SET Statement Pool Parameter Quick Reference	B-69
Table 2-4	INC/EXC Statement Pool Parameter Quick Reference	B-69
Table 2-5	SET Statement SMS Pool Parameters	B-73
Table 2-6	INC/EXC Statement SMS Pool Parameters	B-73
Table 2-7	SET Statement Calendar Parameters	B-74
Table 2-8	INC/EXC Statement Calendar Parameters	B-74
Table 2-9	SET Statement Variable Parameters	B-76
Table 2-10	INC/EXC Statement Variable Parameters	B-76
Table 2-11	SET Statement Function Parameters	B-77
Table 2-12	SET Statement Diagnostic Parameters	B-81
Table 2-13	INC/EXC Statement Diagnostic Parameters	B-81
Table 2-14	SET Statement Event Parameters	B-82
Table 2-15	INC/EXC Statement Event Parameters	B-82
Table 2-16	SET Statement VTOC Scan Facility Parameters	B-86

Table 2-17 INC/EXC Statement SMVSCFxx ParameterB-86

About This Book

This book contains information that is common to all MAINVIEW® Storage Resource Manager by BMC Software (formerly known as RESOLVE® SRM for OS/390) and is intended for storage administrators.

To use this book, you should be familiar with the OS/390 operating system, job control language (JCL), and the Interactive System Productivity Facility (ISPF)

How This Book Is Organized

This book is organized as follows. In addition, an index and glossary appear at the end of the book.

Chapter/Appendix	Description
"Chapter 1, Introducing MAINVIEW SRM"	helps you learn how to use MAINVIEW to monitor system performance
"Chapter 2, Defining the MAINVIEW SRM System"	provides an overview of global operational characteristics and system-wide functions of MAINVIEW SRM
"Chapter 3, Understanding General Syntax Rules"	provides an overview of the allowable structures of the statements and parameters used in MAINVIEW SRM
"Chapter 4, Using MAINVIEW SRM Operator Services"	describes the functions of SVOS, the started tasks that run all MAINVIEW SRM products
"Chapter 5, Using MAINVIEW SRM Easy Menu"	helps you learn how to use MAINVIEW SRM Easy Menu
"Chapter 6, Defining and Activating Functions"	describes how to use the SMFUNCxx parmlib member to define and activate functions; also describes in detail the USERVARS function, which is used by multiple components in MAINVIEW SRM
"Chapter 7, Defining a Pool"	describes how to use the SMPOOLxx parmlib member to define pools

Chapter/Appendix	Description
"Chapter 8, Using Variables"	describes how to use the SMVARSxx parmlib member to define variables SMVARSxx variables allow a user-specified name to be assigned any number of selection parameters.
"Chapter 9, Defining a Calendar"	describes how to use the SMCALSxx parmlib member to define non-working days in the MAINVIEW SRM system
"Chapter 10, Using Diagnostics"	describes how to use the SMDIAGxx parmlib member and provides an explanation and example of the tracing facility
"Chapter 11, Using Filter and Rule List Parameters"	describes how to use the FLSTxx and RLSTxx parmlib members to define filter and rule list parameters
"Chapter 12, Monitoring Pools and SMS Storage Groups"	describes how to monitor pool, SMS pool, and SMS storage groups
"Chapter 13, Processing SMF Records"	describes to use the logging facilities write activity messages to the SMF data set
"Chapter 14, Generating and Managing Batch Reports"	provides instructions for generating MAINVIEW SRM reports in batch
"Appendix A, Altering Messages with the Message Exit"	provides instructions for modifying messages using the message exit

Related Documentation

BMC Software products are supported by several types of documentation:

- online and printed books
- online Help
- release notes and other notices

In addition to this book and the online Help, you can find useful information in the publications listed in the following table. As "Online and Printed Books" on page xvii explains, these publications are available on request from BMC Software.

Category	Document	Description
MAINVIEW common documents	<i>OS/390 and z/OS Installer Guide</i> <i>MAINVIEW Installation Requirements Guide</i> <i>MAINVIEW Common Customization Guide</i> <i>Using MAINVIEW</i> <i>MAINVIEW Administration Guide</i> <i>Implementing Security for MAINVIEW</i>	provides instructions for installing, configuring, using, and administering MAINVIEW
MAINVIEW SRM customization documents	<i>MAINVIEW SRM Customization Guide</i>	provides instructions for configuring and customizing MAINVIEW SRM for OS/390

Category	Document	Description
reference documents	<i>MAINVIEW SRM Reference Summary</i>	provides a listing and explanation of global system parameters, FLST/RLST parameters, and functions for all MAINVIEW SRM products
product documents	<i>MAINVIEW SRM DMS2HSM User Guide and Reference</i> <i>MAINVIEW SRM EasyHSM User Guide and Reference</i> <i>MAINVIEW SRM EasyPOOL User Guide and Reference</i> <i>MAINVIEW SRM EasySMS User Guide and Reference</i> <i>MAINVIEW SRM Enterprise Storage Automation User Guide</i> <i>MAINVIEW SRM SG-Auto User Guide and Reference</i> <i>MAINVIEW SRM SG-Control User Guide and Reference</i> <i>MAINVIEW SRM StopX37/II User Guide and Reference</i> <i>MAINVIEW SRM StorageGUARD User Guide and Reference</i>	provide product-specific information for MAINVIEW SRM products
supplemental documents	release notes, flashes, technical bulletins	provides additional information about the product

Online and Printed Books

The books that accompany BMC Software products are available in online format and printed format. If you are a Windows or Unix user, you can view online books with Acrobat Reader from Adobe Systems. The reader is provided at no cost, as explained in “To Access Online Books.” You can also obtain additional printed books from BMC Software, as explained in “To Request Additional Printed Books.”

To Access Online Books

Online books are formatted as Portable Document Format (PDF) files. You can view them, print them, or copy them to your computer by using Acrobat Reader 3.0 or later. You can access online books from the documentation compact disc (CD) that accompanies your product or from the World Wide Web.

In some cases, installation of Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader from the Web, go to the Adobe Systems site at <http://www.adobe.com>.

To view any online book that BMC Software offers, visit the support page of the BMC Software Web site at <http://www.bmc.com/support.html>. Log on and select a product to access the related documentation. (To log on, first-time users can request a user name and password by registering at the support page or by contacting a BMC Software sales representative.)

To Request Additional Printed Books

BMC Software provides printed books with your product order. To request additional books, go to <http://www.bmc.com/support.html>.

Release Notes and Other Notices

Printed release notes accompany each BMC Software product. Release notes provide current information such as

- updates to the installation instructions
- last-minute product information

In addition, BMC Software sometimes provides updated product information between releases (in the form of a flash or a technical bulletin, for example). The latest versions of the release notes and other notices are available on the Web at <http://www.bmc.com/support.html>.

Conventions

This section provides examples of the conventions used in this book and explains how to read ISPF panel-flow diagrams and syntax statements.

General Conventions

This book uses the following general conventions:

Item	Example
information that you are instructed to type	Type SEARCH DB in the designated field.
specific (standard) keyboard key names	Press Enter .
field names, text on a panel	Type the appropriate entry in the Command field.

Item	Example
directories, file names, Web addresses	The BMC Software home page is at www.bmc.com .
nonspecific key names, option names	Use the HELP function key. KEEPDICTIONARY option
calls, commands, control statements, keywords, parameters, reserved words	Use the SEARCH command to find a particular object. The product generates the SQL TABLE statement next.
Unix commands, command options, database names	Use the sbacktrack program to create a backup script.
code examples, syntax statements, system messages, screen text	//STEPLIB DD The table <i>table_name</i> is not available.
emphasized words, new terms, variables	The instructions that you give to the software are called <i>commands</i> . In this message, the variable <i>file_name</i> represents the file that caused the error.
single-step procedures	»» To enable incremental backups, type y and press Enter at the next prompt.
GUI menu sequence	Choose File => Open .

This book uses the following types of special text:

Note: Notes contain important information that you should consider.

Warning! Warnings alert you to situations that could cause problems, such as loss of data, if you do not follow instructions carefully.

Tip: Tips contain useful information that may improve product performance or that may make procedures easier to follow.

Syntax Statements

Syntax statements appear in Courier. The following example shows a sample syntax statement:

```
COMMAND KEYWORD1 [KEYWORD2 | KEYWORD3] KEYWORD4={YES | NO}
      file_name...
```

The following table explains conventions for syntax statements and provides examples:

Item	Example
Items in italic type represent variables that you must replace with a name or value. Use an underscore for variables with more than one word.	<code>dtsbackup <i>control_directory</i></code>
Brackets indicate a group of options. You can choose at least one of the items in the group, but none of them is required. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option.	<code>[<i>table_name, column_name, field</i>]</code>
Braces enclose a list of required items. You must enter at least one of the items. Do not type the braces when you enter the item.	<code>{<i>DBD_name table_name</i>}</code>
A vertical bar means that you can choose only one of the listed items. In the example, you would choose either <i>commit</i> or <i>cancel</i> .	<code>{<i>commit cancel</i>}</code>
An ellipsis indicates that you can repeat the previous item or items as many times as necessary.	<code><i>column_name . . .</i></code>

Chapter 1 Introducing MAINVIEW SRM

This chapter provides a high-level overview of the BMC Software MAINVIEW Storage Resource Manager (SRM) suite of products. The following information is included:

Overview	1-2
MAINVIEW Architecture	1-2
MAINVIEW Address Spaces	1-3
Communication Between Address Spaces	1-3
MAINVIEW Window Interface	1-4
Activities Monitored by MAINVIEW SRM	1-4
Real-Time and Historical Space Information	1-4
Historical Performance Information	1-5
Budgeting and Quota Management for Logical Groupings	1-5
System-Monitored Timeframes	1-5
Historical Data	1-6
Displayed Information	1-6
Understanding the MAINVIEW Window Interface	1-7
View Categories	1-8
Tabular and Detail Views	1-8
Summary Views	1-9
Customizing Views and Help Text to Meet Your Needs	1-10
Getting Help on Views	1-11
Naming Views	1-12
Moving Around in MAINVIEW SRM	1-13
Using Hyperlinks	1-13
Using Menus	1-14
Using Commands	1-20
Using MAINVIEW SRM on Multiple Systems	1-21
Using Single System Image (SSI)	1-21
Restrictions	1-26
Accessing Another Target	1-26
Using the TIME Command	1-27

Examples of the TIME Command	1-29
------------------------------------	------

Overview

MAINVIEW SRM is a suite of storage products that provides a wide range of services and functions to help you manage storage in your entire system. Built on MAINVIEW architecture, MAINVIEW SRM employs a combination of the MAINVIEW standard ISPF panel interface and MAINVIEW window interface to provide easy, intuitive access to all of the storage data you need.

To use MAINVIEW SRM to its fullest advantage, you should have a good understanding of some of the key concepts and terms that pertain to all aspects of using the product.

This chapter provides background information on MAINVIEW SRM. You may also refer to *Using MAINVIEW* for more detailed information about the MAINVIEW architecture and interface.

While this chapter explains many of the key MAINVIEW window interface terms and concepts, the *MAINVIEW Command List* is the primary reference tool for understanding how to use the interface. The *MAINVIEW Command List* lists all of the topics pertaining to the window interface, along with each topic's ID.

To display information on a topic, on the **COMMAND** line type **HELP** *topic id*; where *topic id* is the name of the desired topic. This book refers you to specific topic IDs as appropriate.

MAINVIEW Architecture

The MAINVIEW architecture for systems management currently supports the following products:

- MAINVIEW for OS/390
- CMF MONITOR Online
- PLEX MANAGER
- VistaPoint
- MAINVIEW for CICS
- MAINVIEW for DB2
- MAINVIEW for IMS
- MAINVIEW for MQ Series
- MAINVIEW SRM

Before you examine MAINVIEW SRM capabilities, you should understand the MAINVIEW architecture.

MAINVIEW Address Spaces

All MAINVIEW products require three address spaces:

Coordinating Address Space

The coordinating address space (CAS), which runs as a subsystem, provides many services to all MAINVIEW products. There is one CAS per OS/390 system image.

Product Address Space

A product address space (PAS) provides special services to one or more related products. MAINVIEW SRM consists of three PASs that serve the MAINVIEW SRM product set.

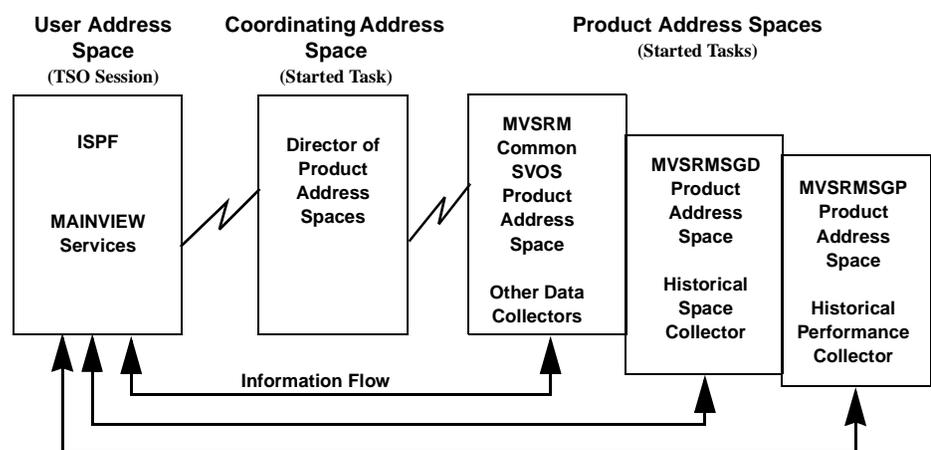
User Address Space

A user address space (UAS) is a TSO or MAINVIEW Alternate Access session.

Communication Between Address Spaces

The communication flow between the MAINVIEW SRM address spaces is illustrated in Figure 1-1.

Figure 1-1 Communication Flow between SRM Address Spaces



The MAINVIEW SRM coordinating address space and the product address spaces are set up by your system administrator. Under most circumstances, the common MVS SRM product address space is started automatically at each system IPL. The others are started as the associated product component is started.

Because the MAINVIEW SRM data collectors are isolated from the coordinating address space, you can control the two independently. This arrangement is especially useful if you want to run MAINVIEW SRM only during certain periods of the day—you can stop the product address space without stopping the coordinating address space, thus keeping other MAINVIEW products up and running.

MAINVIEW Window Interface

Each product in the MAINVIEW family takes full advantage of the MAINVIEW window interface, a robust, easy-to-use extension of the standard ISPF interface. With the MAINVIEW window interface, you can use multiple products to control and monitor multiple resources on multiple systems—all on just one screen.

Activities Monitored by MAINVIEW SRM

Storage administrators can use the MAINVIEW SRM solution provides to manage performance and space consumption for various storage resources and processes. The centralized storage management of MAINVIEW SRM allows you to manage storage availability and performance in the context of your total system management strategy.

MAINVIEW SRM provides powerful and flexible features, including

- real-time and historical space information
- historical performance information to the data-set level
- budgeting and quota management for logical groupings (applications)

Real-Time and Historical Space Information

Real-time and historical space information enables proactive management of storage subsystem performance. Information about activity for devices, I/O queuing, channels, and contention by enqueue and reserves is available in real time.

Historical space views assist you in determining the current use and growth of DASD in your data center. DASD usage can be reported from several different views, including overall summary by time, by storage pools, by RAID volumes, by applications, and by volumes.

Historical Performance Information

Historical performance databases collect detailed historical information for storage performance and capacity. Performance statistics for data sets include response time, I/O rates, and cache activity. Performance and storage-use statistics are collected and summarized at several different levels. For example, you may want to use monthly summaries to forecast storage usage or use daily and interval information to pinpoint a problem. Performance reports are updated automatically at user-defined intervals (called snapshots).

Budgeting and Quota Management for Logical Groupings

The MAINVIEW SRM SG-Control product, in combination with the MAINVIEW SRM StorageGUARD product

- allows you to analyze trends in the data for planning purposes
- obtains the true minimum and maximum occupancy information; when you are budgeting by logical groups, it can warn you about or even deny allocations when budgets are exceeded.
- displays application and functional views of the storage usage, as well as other information needed to manage the DASD

System-Monitored Timeframes

Although MAINVIEW SRM continually gathers and stores information on workloads, address spaces, system resources, delays, and thresholds, you control when and how often that information is displayed.

MAINVIEW SRM allows you to display the same data in two different timeframes:

Timeframe	Description
Real Time	MAINVIEW SRM can capture up-to-the-second performance, configuration, and utilization information to provide you with the most timely information as it exists at the moment of inquiry.
Historical	You can use interval data to pinpoint and analyze other occurrences of the problem in the past.

Historical Data

MAINVIEW SRM allows you to effectively recreate the operating environment as it existed during a particular time period in the past. The historical space and historical performance databases store information about your storage environment at the end of each interval, allowing you to compare today's storage usage and performance to yesterday's, last week's, or last month's—all on the same screen. You can use this information to determine whether current behavior is an anomaly or part of a trend.

For information on using the historical space and historical performance databases, see the *MAINVIEW SRM StorageGUARD User Guide and Reference*.

Displayed Information

MAINVIEW SRM displays the information that it gathers in a view. You see a set of rows and columns that presents data on a particular topic in tabular form. When you select a view for display, a structured query is executed against the MAINVIEW SRM collection of data to retrieve the relevant information. The data is formatted according to the associated set of instructions for the selected view.

All but real-time information depends on the date and time you request before selecting the view. For information on how to set the date and time for displayed information, see "Using the TIME Command" on page 1-25.

With MAINVIEW SRM, you can change a view's format—or form—without affecting its underlying query. For information on how to accomplish this action, on any MAINVIEW **COMMAND** line type **HELP FORM**.

Figure 1-2 shows an example of view output.

Figure 1-2 MPOOLVOL View

```

03JAN2001 13:01:24 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =MPOOLVOL=====SJS=====*=====03JAN2001==13:01:24====MVSRM=====578
Volume UCB  Status/ Use      SMS      Pool/SG
----- Addr Use      Attrib  Status  Name
--83B5 83B5 ONLINE PRIVATE      MRWPOOL
--833C 833C ONLINE PRIVATE      MRWPOOL
ACFWK1 8599 ONLINE PRIVATE      MRWPOOL
ACFWK2 859A ONLINE PRIVATE      MRWPOOL
ACFWK3 859B ONLINE PRIVATE      MRWPOOL
ACFWRK 85A7 ONLINE PRIVATE      MRWPOOL
APDPUB 855E ONLINE PRIVATE      MRWPOOL
APDS01 8594 ONLINE PRIVATE      MRWPOOL
APD26D 85A1 ONLINE PRIVATE      MRWPOOL
APD26E 859F ONLINE PRIVATE      MRWPOOL
APD26F 8558 ONLINE PRIVATE      MRWPOOL
APD260 854A ONLINE PRIVATE      MRWPOOL
APD261 854B ONLINE PRIVATE      MRWPOOL
APD262 854C ONLINE PRIVATE      MRWPOOL
APD27D 85A0 ONLINE PRIVATE      MRWPOOL
APD27E 859E ONLINE PRIVATE      MRWPOOL

```

This view, called MPOOLVOL, displays information about volumes in a pool. MAINVIEW SRM provides more than 75 views, each focusing on a different aspect of system storage.

Understanding the MAINVIEW Window Interface

All MAINVIEW products use either the MAINVIEW window interface or the MAINVIEW standard ISPF panel interface (or a combination of both). MAINVIEW SRM products operate in the window environment with only two exceptions, Edit and Browse, which use ISPF.

In the MAINVIEW window environment, each view is displayed in its own window. The top row of each window, called the window information line, displays (among other things) the number and status of the window; the name of the view; the system, date, and time that are reflected by the view; and the name of the MAINVIEW product you are currently using. In Figure 1-2, the window information line looks like this:

```
W1 =MPOOLVOL=====SYS=====*=====03JAN2001==13:01:24====MVSRM=====578
```

Everything below the window information line is called the display area. In Figure 1-2, the display area contains the MPOOLVOL data.

The top three lines of the MAINVIEW window interface are called the window control area. The window control area consists of the Information Display line (which contains the current date and time), as well as the **COMMAND** line, the **SCROLL** field, the **CURR WIN** (current window) field, and the **ALT WIN** (alternate window) field.

The window information line displays a lot of information; however, its exact appearance depends on many factors. For information on any of these fields, place the cursor on the field and press **PF1** (HELP). The window information line and its fields also are covered in detail in *Using MAINVIEW*.

View Categories

The following kinds of views are available in MAINVIEW products:

- *Menu views* allow you to hyperlink to other views. Some menus hyperlink to views that display information about your system; some menus hyperlink to more specific views or menus that allow you to focus on the information you need.
- *Tabular views* are rows and columns of data. Each field in a given row addresses the same pool, volume, or device. Most views are tabular.
- *Detail views* in MAINVIEW SRM provide the same information that is on a row in vertical format. For example, if you select the detail view for a row on the SPSNAP tabular view, the SPSNAPDT detail view displays all of the fields from that row in a vertical format.
- *Summary views* provide compressed and summarized data.
- *User views* are views you create yourself to help you to focus on storage performance aspects critical to your applications. These views are stored in the user view category.

Tabular and Detail Views

MAINVIEW SRM provides tabular and detail views to help you monitor discrete areas of storage activity and perform various administrative tasks.

MAINVIEW SRM tabular and detail views are divided into the following categories:

- Real-time monitor views monitor current storage activity of pools, storage groups, RAID configurations, and storage performance.

- Historical data views report on historical space, historical performance, DFHSM, and application activity. They depend on the date and time you request before selecting the views. For information on how to set the date and time for displayed information, see “Using the TIME Command” on page 1-25.
- Tools and menus views provide a list of all MAINVIEW SRM views and set of real-time data set-level and VTOC-level reports to simplify daily DASD housekeeping functions, including
 - HLQ (High-Level Qualifier)
 - Catalog Super Locate
 - VTOC Reporting
- Administrative views, which help you manage various aspects of the MAINVIEW SRM components.

Summary Views

Summary views can be used for the following tasks:

- You can compress several rows of data into a single row based on certain criteria. For example, a summary view focusing on LCU performance might compress the Channel Path field so that each channel path is represented by a single row of data. You can create summary views from tabular views by using the GROUP BY customization option. For more information, on the **COMMAND** line type **HELP CUSTOM**, and then select the GROUP BY option.
- You can summarize a resource across multiple local and remote systems, which allows you to compare and contrast data from different systems. For example, MDEVSUM summarizes device activity across all (CON ALL) or selected systems in the sysplex using the CONtext command. For more information on the CONtext command, on the **COMMAND** line type **HELP CON**.
- You can summarize historical space and historical performance snapshots by using the TIME command range. For more information about the TIME command, on the **COMMAND** line type **HELP TIME**.

Customizing Views and Help Text to Meet Your Needs

One of the primary advantages of the MAINVIEW SRM window interface is that you can customize most views and help text to meet the particular needs of your installation.

View Customization

With the MAINVIEW view customization facility, you can perform a wide variety of activities, including the following:

- sort on multiple columns
- rearrange columns
- graph the data
- modify the view so that certain columns are completely hidden, thus displaying only the data you need

You can enter the view customization facility by typing **CUSTOM** on the **COMMAND** line.

For explicit instructions on how to customize MAINVIEW SRM views, on the **COMMAND** line type **HELP CUSTOM**.

Help Text Customization

To create your own help text, see the *MAINVIEW Administration Guide*. You can store this help text in your own private help text library or make it accessible to all MAINVIEW SRM users at your site. Your MAINVIEW SRM product administrator can show you how.

Getting Help on Views

MAINVIEW views are virtually self-documenting; no matter how you customize a view by using the CUSTom command, the online help always draws from the most current information and is thus always accurate.

Figure 1-3 depicts where and how you can get help on views.

Figure 1-3 Online Help Available for MAINVIEW Products

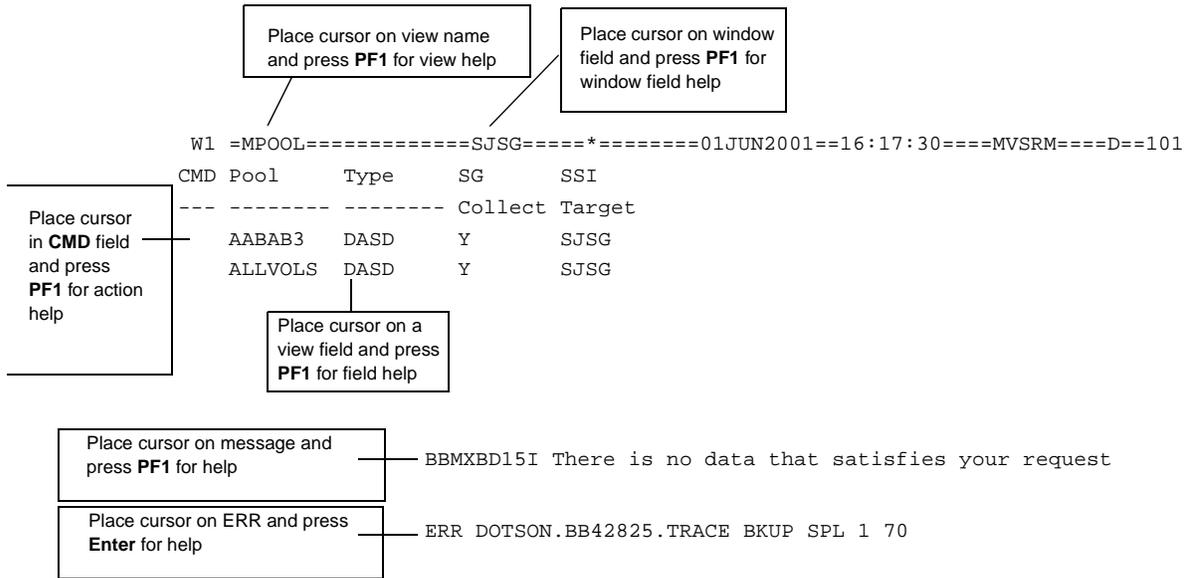


Table 1-1 describes the different types of available online help.

Table 1-1 Online Help Available for MAINVIEW Products (Part 1 of 2)

To display this	Do this
Help on a view	Place the cursor on the view name on the window information line and press PF1 . Alternatively, on the COMMAND line, type HELP <i>viewname</i> . View help displays other topics that tell you which parameters are currently in effect; which fields are included and excluded within the view; which fields have hyperlinks and to where; primary action commands, line action commands; and so on.
Help on actions	Place the cursor in the CMD field and press PF1 . Action help displays primary action commands and line action commands available for the view
Help on a field that appears in a view	Place the cursor on the field and press PF1 .

Table 1-1 Online Help Available for MAINVIEW Products (Part 2 of 2)

To display this	Do this
Help on a field on the window information line	Place the cursor on the field and press PF1 .
Help on message	Place the cursor on the message ID and press PF1 .
Help on errors	Place the cursor on the ERR and press Enter .
Help on a command or topic pertaining to the MAINVIEW window interface itself	On the COMMAND line, type HELP topic id , where topic id is the ID of the topic as listed in the <i>MAINVIEW Command List</i> . (For example, HELP ASU gives you help on the ASU command.) Alternatively, place the cursor on the COMMAND line and press PF1 to display the MAINVIEW help tutorial. Select either Beginning or Advanced topics, or type INDEX to display all the available topics.

Naming Views

To help you understand the aspect of your system that a particular view monitors, and in which timeframe, view names use following naming convention:

Vtxxxxxx

Vt Identifies the view type

- AD** SRM Parmlib and Function views
- AP** SG-Control application views
- CO** component status views
- EZ** Easy Menus
- Mt** real time monitor views; where *t* is the monitor type
- PR** historical performance views
- RE** RAID EMC Symmetrix views
- RI** RAID IBM views
- SP** historical space views
- WB** workbench views

xxxxx Identifies the area of interest. For example, POOL in the view name MPOOL indicates that the view monitors pool activity; VOL in the view name MVOLPAV indicates that the view monitors volumes.

Moving Around in MAINVIEW SRM

Thus far, you have learned that MAINVIEW SRM displays the information it collects in the form of views—one view for each type of activity, area of interest, and timeframe.

You can use the following methods to display these views and the rest of the services that are provided by MAINVIEW SRM:

- hyperlinks
- menus
- commands

After you become comfortable with each method, you will most likely find that using them in combination affords you the greatest degree of flexibility and control.

You can use hyperlinks, menus, parameters, and commands interchangeably.

If a hyperlink does not exist for a desired view and you do not want to establish one, you can always display the view you want by typing its name (and parameters) on the command line. You can also retrace up to 20 steps in a window by pressing **PF3** repeatedly, or get back to the MAIN or EZSRM views at any time by typing the view name on the **COMMAND** line.

Using Hyperlinks

A *hyperlink* is a way of executing a command without explicitly entering it. You can think of a hyperlink as a fast path to another view or command. When you place the cursor on a field for which a hyperlink exists and press **Enter**, the underlying command is executed and its output is displayed. In most cases, this output is another view.

The field names for which a hyperlink exists appear on your terminal in a different color. On monochrome terminals, hyperlinked fields highlighted.

All MAINVIEW window interface products allow you to establish your own hyperlinks. After you start using MAINVIEW SRM, you may find that you habitually follow a path through the product that is not supported by the default hyperlinks. To find out how to override these defaults and create your own hyperlinks, on the **COMMAND** line type **HELP HYPERLINK**.

Using Menus

This section describes the use of Easy Menus, pop-up menus, and EZcmd Menus.

Easy Menus

An Easy Menu consists of a series of options, all of which hyperlink to either data views or to other menus that are specific to that particular option. The names of all Easy Menus are prefixed with the letters EZ.

Options that are preceded by a period (.) display system data; options that are preceded by a *greater than* sign (>) display either pop-up menus in the center of the window or other menu views from which you can access a broad variety of information.

The following example illustrates how you might use an Easy Menu.

Example

The EZSRM Menu, (Figure 1-4) is presented as the initial display when you access MAINVIEW SRM. All options on this menu hyperlink to high-level views or submenus and have been given succinct, descriptive names, allowing you to quickly access the data you need.

Figure 1-4 EZSRM Menu

```

14MAY2001 09:43:29 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =EZSRM=====SJS=====14MAY2001==09:43:29==MVSRM====D====1
                                EZSRM  Menu

SRM Real Time Monitor  +-----+ SRM Historical Data
                        |         |
. Pools                 | Place cursor on |
. SMS Storage Groups   | menu item and | > Historical Space
. SMS Pools            | press ENTER  | > Historical Performance
> RAID Configurations  +-----+ > EasyHSM
> Storage Performance  > SGControl Applications
                        > SMF Report Library

SRM Administration    SRM Tools and Menus

> Parmlib Members     > Workbench
. Functions            . MVSRM View List
. SRM Component Status . MVSRM Batch Reports
                       . MainView Messages
                       . Return....

```

When you select Historical Space and press **Enter**, the EZSRMSGD menu is displayed (Figure 1-5).

Figure 1-5 EZSRMSGD Menu

```

14MAY2001 09:20:36 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRMSGD=====SJSG=====*=====14MAY2001==09:20:36===MVSMSGD=D====1
          EZSRMSGD Menu

Historical Space      +-----+
> Summary by Time    | Place cursor on |
. Pool Utilization   | menu item and  |
. RAID Physicals     | press ENTER    |
. SGControl Applications
. Volumes

SRM Administration          SRM Tools and Menus

> Parmlib Members        > Workbench
. Functions              . MVSMSGD View List
. SRM Component Status   . MVSMSGD Batch Reports
                          . Return....
    
```

You can tab to any of the EZSRMSGD Historical Space options and press **Enter** to display information that is specific to the device or snapshot selected.

Easy Menu options have been given descriptive, intuitive names that correspond to some aspect of system storage. This convention allows you to use MAINVIEW SRM quickly and easily without having to learn the names and functions of specific views.

Data Input Filtering from Menus

When you select some menu options, a data entry panel is displayed in which you can filter the data you want to see. Figure 1-6 shows an example of a data entry panel.

Figure 1-6 Catalog Super Locate Data Filter Panel

```

13APR2001 09:08:37 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =WBSLDET=====SJSG=====13APR2001==09:08:37===MVS RM====D====1
<== Type S to process request.

Catalog Super Locate.....
Data Set Name.....
Data Set Type e.g. All, VSAM..
Migrated Data Sets Yes/No/Only

```

» To filter data, complete the fields provided and type S to the left of the <== symbol to process the request.

Pop-Up Menus

Some menu options that are preceded by a greater than sign (>) display pop-up menus in the center of the window. For example, when you select Storage Performance from the EZSRM menu, the Storage Performance pop-up menu is displayed, as shown in Figure 1-7. While the pop-up menu is displayed, you can only access the menu items in the pop-up menu. Press **PF3** to return to the EZSRM Menu.

Figure 1-7 EZSRM Menu, Storage Performance Pop-Up Menu

```

14MAY2001 09:18:38 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRM====EZSRMS====SJSG====*====14MAY2001==09:16:35====MVSRM====D====1
                                EZSRM Menu

SRM Real Time Monitor                SRM Historical Data
+ Storage Performance =+
. Pools                               . Device Activity      . > Historical Space
. SMS Storage Groups                  . Channel Activity     . > Historical Performance
. SMS Pools                           . I/O Queueing         . > EasyHSM
> RAID Configurations                 . ENQ/Reserve Activity. > SGControl Applications
> Storage Performance                 . ENQ Activity         . > SMF Report Library
. Return...                           .                      .

SRM Administration                   +-----+ SRM Tools and Menu
> Parmlib Members                     > Workbench
. Functions                            . MVSRM View List
. SRM Component Status                 . MVSRM Batch Reports
                                        . MainView Messages
                                        . Return...
    
```

EZcmd Menus

For any view with action line commands that hyperlink to another view, you can use the EZcmd menu to hyperlink rather than the action line command. To use an EZcmd menu, perform the following steps:

- Step 1** On the view you select, place the cursor on the first data field in the view.
- Step 2** Press **Enter**. The **EZcmd** menu is displayed (Figure 1-8)

Figure 1-8 EZcmd Menu Example

```

20APR2001 10:44:00 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
W1 =MPOOL===MPOOLZ===SJSJG=====*=====20APR2001==10:43:56===MVS SRM===D===1
                                     MPOOL EZcmd Menu

Selected Pool..... ALLVOLS

      Commands      +-----+
. Display Pool Definition | Place cursor on |
. Volumes Performance   | menu item and  |
. Volumes Space         |   press ENTER  |
. Pool Volumes          +-----+
. Detail View

```

- Step 3** On the EZcmd menu, place the cursor on the view you want to see and press **Enter**. The view you requested is displayed.

Using Commands

As an alternative to using hyperlinks or menus, you can display a view—or issue a MAINVIEW window interface command—by typing the view name or command on the **COMMAND** line.

Note: MAINVIEW window interface commands are available to all products that use the MAINVIEW window interface. For a list of the MAINVIEW window interface commands, see the *MAINVIEW Command List*.

Tip: MAINVIEW SRM provides additional commands that are described in the online help for each view. Many of these commands provide power users with functionality otherwise unavailable.

You can filter data by specifying parameters with your view commands. For example, to display high-level qualifiers for data sets that begin with **BACKUP**, type

WBHLQ BACKUP*

Note: Some views have *required* positional parameters that must be specified before any available filter parameters. Failure to specify the parameters causes error messages and the view not to be displayed. See “Using Positional Parameters” on page 1-20.

Using View Parameters

Most views have been defined with a set of parameters. Use the view help information to discover the parameters that are defined for a specific view.

Using Positional Parameters

When you use *positional parameters*, you supply values for the parameters in a predetermined order. To identify the parameters and their order for a given view, display the view’s online help, place the cursor on the highlighted term **positional parameters**, and then press **Enter**.

Note: If a column is not defined as a parameter by default, you can make it so by typing **CUST** on the **COMMAND** line and then choosing the **L** (Filter) and **P** (Parameters) options.

For more information, on the **COMMAND** line type **HELP CUSTOM**

Using MAINVIEW SRM on Multiple Systems

One of the greatest benefits of the MAINVIEW window interface is that it allows you to control multiple local and remote systems, access different products on those systems, and compare and contrast data from different time periods—all on the same screen, *all at the same time*.

You can monitor multiple systems from MAINVIEW SRM in the following ways:

- Using the MAINVIEW Single System Image (SSI) function, you can retrieve data from multiple systems in a single view.
- Using the MAINVIEW window interface, you can open up to 20 windows¹ and control a different aspect of system performance in each window, looking at various systems and using various views.

With MAINVIEW architecture, you can do all these activities from a single user session, rather than initiating multiple sessions under the control of a session manager.

To monitor other systems, you can specify a *target* (the name of an OS/390 image, such as SYSA or SYSB) or an *SSI context* (a group of targets across multiple systems or sysplexes that looks and acts like a single target).

Using Single System Image (SSI)

MAINVIEW offers you the ability to combine data from many target systems into a single view and work with the information as if it were from a single system—a *Single System Image (SSI)*. In any multisystem environment where coordinating address spaces (CASs) are configured to communicate with each other, views are enabled for SSI mode. You can define an SSI context outside of sysplex boundaries in any configuration that best suits your purposes.

A summary view can combine data from multiple OS/390 images into a single row.

To use SSI, perform the following steps:

- Step 1** Make sure your CASs are configured to communicate with each other and are running on BBI version 3.3 and above.
- Step 2** Define an SSI context.

1. Twenty windows can be opened concurrently.

These steps are discussed in the *MAINVIEW Common Customization Guide*.

Note: The SSI context ALL is predefined to include all active SSI contexts at your installation. However, context ALL can be customized to include only certain SSI contexts. See the *MAINVIEW Common Customization Guide* for details.

Step 3 Use the CONtext command with the SSI context name to see the data. See “Using the CONtext Command” on page 1-22 for more information.

You do not need a sysplex configuration to use the MAINVIEW SSI support. The data from any MAINVIEW SRM product that connects to a MAINVIEW CAS, where cross-system communication is established, can be included in a view in SSI mode.

Note: Be sure to keep the names of your sysplexes unique; otherwise you might assume you are viewing one sysplex when in fact you are viewing a different one by the same name.

Dynamic Fields

Two fields, SSI Target and SSI System, can be set to appear dynamically in tabular views when you access an SSI context. SSI Target is a single OS/390 image (as defined in TGTDEF, a Plex Manager ADMIN view). SSI System is the CAS system being monitored (as defined in CASDEF, a Plex Manager ADMIN view). Figure 1-9 on page 1-24 shows how these fields appear in a view.

Whether one, both, or neither of these fields stays in a view is up to you. You can use the INclude and EXclude commands; for example, typing **INclude TARGET** on the **COMMAND** line causes the SSI Target field to be displayed, regardless of the default setting at your installation. Likewise, **EXclude SYSTEM** on the **COMMAND** line hides the SSI System field (of course, both commands work with either field). You can also enter View Customization mode and use the exclude toggle command to display or hide the fields.

Using the CONtext Command

Use the CONtext command to enter into SSI mode. This section provides an overview of the CONtext command.

- To access an SSI context, type

CONtext *ssi name product id*

ssiname

Is the name of the SSI context (one or more OS/390 images). Specify ALL for all active contexts. This is a required parameter.

productid

Is the product identifier (CMF, MVMVS, PLEXMGR). This parameter is optional. The default value is the current product you are using. (The values * and = also default to the current product.)

When you access an SSI context, the window information line reflects the product and location(s) accessed.

To view data in all active targets, type

CON ALL

- If you want to access a single target (whether or not your installation has an SSI context defined), type

CONtext *targetname* *productid*

targetname

Is the name of the OS/390 target you want to access. This parameter is required. An asterisk (*) in this field specifies the local system to which you originally logged on. An equal sign (=) retains the current target you are using.

productid

Is the product identifier (MVSRM, MVSRMSGD, MVSRMSGP, MVSRMHSM, and MVSRMSGC). This parameter is optional. The default value is the current product you are using. (The values * and = also default to the current product.)

For more information on the CONtext command and its parameters, on the **COMMAND** line, type **HELP CONtext**.

Example

Assume that your installation has defined an SSI context called PRODUCTN to include your company's production systems.

- To view your device data from all your production systems, on the **COMMAND** line type the command

CON PRODUCTN M/SRM; MDEV

- To view a subset of the data fetched by **PRODUCTN**—to limit the data to devices running on **SYSB**, for example:
 - On the **COMMAND** line, type the command **CON PRODUCTN**, if you have not already done so.
 - Type the **SCOpe** command

SCO SYSB

The MDEV view is displayed.

Using SSI to Check on Devices

Suppose you want to check on devices that are being shared by systems in your sysplex to see if any devices are experiencing high levels of contention.

- On the **COMMAND** line, type **MDEV**.
- Type **CONtext ALL**.

In the example shown in Figure 1-9, **CONtext ALL** includes OS/390 images SYSG and SYSH.

Figure 1-9 MDEV View with CONtext ALL Specified

```

23FEB2001 09:17:54 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> PAGE
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =MDEV===== (ALL=====*) 23FEB2001==09:17:38====MVS RM====D=3468
Volser Dev  SSI      LCU Actv Resp IOSQ Pend Disc Conn  %Dev %Resv Open Alloc
----- Num  Target  --- Rate Time Time Time Time Time Util ----- Cnt  Cnt
BSDW09 853A SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.1 0.0 0.0 2 90
BSDW09 853A SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.1 0.0 0.0 2 90
BSDW10 853B SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.2 0.0 0.0 0 0
BSDW10 853B SJSH     BD 0.2 11.4 0.0 0.1 0.0 11.1 0.2 0.0 1 120
BSDW10 853B SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.2 0.0 0.0 0 0
BSDW10 853B SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.2 0.0 0.0 0 0
BSDW10 853B SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.2 0.0 0.0 0 0
BSDW10 853B SJSG     BD 0.0 1.5 0.0 0.1 0.0 1.2 0.0 0.0 0 0
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW11 853C SJSH     BD 0.1 23.4 0.0 0.2 0.0 23.0 0.4 0.0 0 1
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW11 853C SJSG     BD 0.0 1.5 0.0 0.2 0.0 1.1 0.0 0.0 0 0
BSDW12 853D SJSG     BD 0.0 1.4 0.0 0.1 0.0 1.1 0.0 0.0 1 90
BSDW12 853D SJSH     BD 0.4 11.2 0.0 0.1 0.0 10.9 0.5 0.0 1 36

```

Suppose that you are concerned about the high response time shown for device number 853C. Hyperlink on one of the **Volser** field lines on which the **Dev Num** is 853C.

The resulting MDEVSP view (Figure 1-10) shows you how device 853C is being used.

For more examples, type **HELP CONtext** or **HELP SCOpe** on the **COMMAND** line.

Figure 1-10 MDEVSP for Device 853C

```

23FEB2001 09:19:19 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =MDEVSP===== (ALL=====*)=====)23FEB2001==09:19:18====MVS RM====D====6
Time      SSI      Intvl Storage  Volser Dev Dev  LCU Actv Resp IOSQ CUB  DPB
----- Target  ----- Group  ----- Cls Num  --- Rate Time Time Dly  Dly
09:19:18 SJSH      0          BSDW11 DAS 853C  BD 16.7 16.7  0.0  0.0  0.0
09:19:19 SJSG      0          BSDW11 DAS 853C  BD  6.7  6.7  0.0  0.0  0.0
09:19:19 SJSG      0          BSDW11 DAS 853C  BD  6.7  6.7  0.0  0.0  0.0
09:19:19 SJSG      0          BSDW11 DAS 853C  BD  6.7  6.7  0.0  0.0  0.0
09:19:19 SJSG      0          BSDW11 DAS 853C  BD  6.7  6.7  0.0  0.0  0.0
09:19:19 SJSG      0          BSDW11 DAS 853C  BD  6.7  6.7  0.0  0.0  0.0

```

What Happens if a System in Your Context Goes Down?

If one of the systems in your sysplex goes down while you are using MAINVIEW SRM with an SSI context, you are notified immediately with a message similar to the one shown in Figure 1-11.

Figure 1-11 Error Message if a System in Your Context Goes Down

```
09NOV1998 11:58:30 ----- INFORMATION DISPLAY -----  
COMMAND ==>                                     SCROLL ==> PAGE  
CURR WIN ==> 1           ALT WIN ==>  
W1 =JOVER===== (ALL=====*)=====)09NOV1998 ==11:58:30====MVMVS=====  
BBMXV331I Target SYSC      left the SSI context
```

Restrictions

You cannot use the SYSPROG, CSMON, or CONSOLE utilities in SSI mode, nor can you use the WKLIST view.

Accessing Another Target

If you want to access a single target (whether or not your installation has an SSI context defined), on the **COMMAND** line, type

CONtext *targetname productid*

targetname

Is the name of the OS/390 system you want to access. This parameter is required. An asterisk (*) in this field specifies the local system to which you originally logged on. An equal sign (=) retains the current target you are using.

productid

Is the product identifier (CMF, MVMVS, PLEXMGR). This parameter is optional. The default value is the current product you are using. (The values * and = also default to the current product.)

Using the TIME Command

The ability to look at data from the past (called *historical data*) is an extremely powerful tool. When you access historical data, MAINVIEW SRM presents data from the most recent interval specified and the preceding intervals. Using the TIME command, you can specify intervals from any timeframe for which data exists on your system.

Tip: Most views you will retrieve from MAINVIEW SRM depend on date and time. If the TIME command is *not* set, the default is the current date and time and requests for views result in “no data” errors.

When you issue the TIME command with no parameters, MAINVIEW SRM prompts you for the parameters on a pop-up panel, as shown in Figure 1-12.

Figure 1-12 SET TIME FRAME Panel

```

----- SET TIME FRAME -----
COMMAND  ===>

Requested Time Frame:
End Date  ===> *                (*, =, or ddmmmyyyy)
End Time  ===> *                (*, =, or hh:mm)
Duration  ===> 1I              (*, =, NEXT, PREV, TODAY, MONTH,
                               nnnnI, nnnnM, nnnnH, nnnD, or nnW)
DOW Mask  ===> EVERYDAY        (EVERYDAY, WEEKDAYS, WEEKENDS)
TOD Mask  ===> ALLDAY          (ALLDAY, PRIMESHIFT, SWINGSHIFT,
                               GRAVEYARDSHIFT)

Data in the Requested Time Frame:
Interval  ===> 1M              (Length, in minutes, of one interval)
End Date  ===> 20DEC2000       (End date of data)
End Time  ===> 09:38          (End time of data)
Duration  ===> 1M              (Minutes spanned by data)
DOW Mask  ===> EVERYDAY        (Day-of-week mask)
TOD Mask  ===> ALLDAY          (Time-of-day mask)

Type END to set the window's requested time frame

```

The syntax for the TIME command is

TIME [*date time* [*duration* | **NEXT** | **PREV**]] [**DOW** *mask* **TOD** *mask*]

Press PF3 to set the time.

date is the ending date of the data you want to see

This parameter is required. An asterisk (*) gives you the default value, which is the current date.

Specify the date in the same format as the current date, which always appears in the upper left corner of the screen.

You can change the format of the date by selecting Option 0 on the MAINVIEW Selection Menu, and then selecting Option 4 on the MAINVIEW Parameter Editors screen.

time is the ending time of the data you want to see

This parameter is required. An asterisk (*) gives you the default value, the current time.

Specify the time in the format hh:mm.

duration

is the time period over which you want your data summarized

This parameter is optional. The default is one recording interval (usually 15 or 30 minutes).

Specify the duration in the format *nnnnu*

nnnn indicates the number of hours, minutes, or intervals in the duration

u indicates the unit of time: I (intervals), M (minutes), H (hours), D (up to 416 days), or W (up to 59 weeks)

TODAY or **TDAY**

specifies today's intervals since midnight

MONTH

specifies one month

NEXT can be specified instead of the duration parameter

NEXT uses the duration value currently in effect to cycle forward by the duration amount.

PREV can be specified instead of the duration parameter

PREV uses the duration value currently in effect to cycle backward by the duration amount.

DOW mask

limits the selected intervals to end on specific days of the week

TOD mask

limits the selected intervals to end within a specific time of the day

Tip: In place of the date, time, or duration parameters, you can use

- | | |
|--------------------------|---|
| An asterisk (*) | to specify the default value
The default values are: the current date, the current time, and one recording interval. |
| An equal sign (=) | to specify the currently requested date, time, or duration |

Examples of the TIME Command

The following examples demonstrate several different uses of the TIME command.

Example

Assume that today is June 10, 2000. To retrieve data from one week ago at 9:25am, type

TIME 06/03/00 09:25

This displays data from the end of the interval that contains 9:25; that is, the interval between 9:15 and 9:30.

Example

To display data from the next interval starting on the same date and time as the last interval specified, type

TIME == NEXT

The NEXT parameter steps forward one Extractor interval (the default) from the date and time last specified. Specifically, data from June 3 during the interval 9:30-9:45 is displayed.

Note: You may find it useful to set one PF key to issue **TIME == NEXT** and another PF key to issue **TIME == PREV**. This allows you to cycle quickly through recording intervals without having to manually type the TIME command and all its parameters.

Example

To display data from the 3-hour period ending on June 6, 2000 at 12 noon, type

TIME 06/06/00 12:00 3h

Assuming 15-minute intervals, the duration field contains 180M—four intervals per hour.

Example

To display data from the next day during the same time period, type

TIME 06/07/00 ==

The equal sign in this position retains the time you specified last, 12:00, and the duration you last specified, 3h.

Example

To display data that includes the 30-minute interval ending at 8:00 on June 16, type

TIME 06/16/00 08:00 30M

Example

To display data from earlier today at 9:00, type

```
TIME * 9:00
```

The asterisk in this position indicates the current date.

Example

To reestablish the current timeframe, type

```
TIME * * *
```

Example

To display data that includes all intervals ending on prime shifts for weekdays from last month, type

```
TIME ENDOFMONTH 23:59 MONTH WEEKDAYS PRIMESH
```

Chapter 2 Defining the MAINVIEW SRM System

This chapter provides an overview of global operational characteristics and system-wide functions of MAINVIEW SRM. The following information is included:

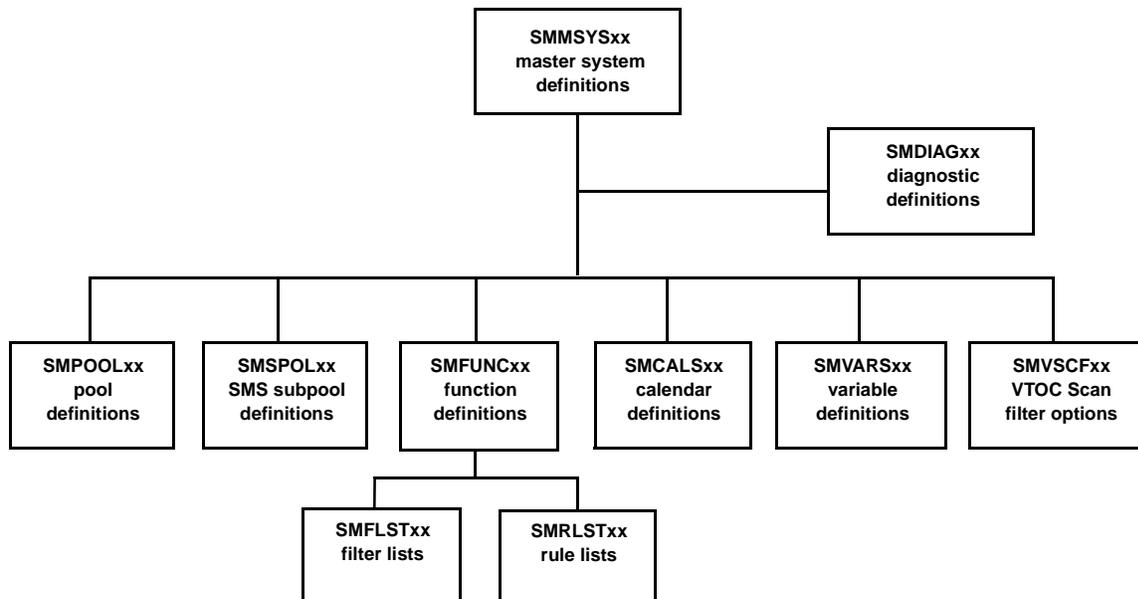
Overview	2-1
System Configuration Members	2-2
Parmlib Member Descriptions	2-3
MAINVIEW SRM Parameters	2-5
Action Parameters	2-6
Selection Parameters	2-6
Defining the Sysplex Environment	2-10
Sysplex Parameters	2-11
Syntax Considerations for the Shared Parmlib Environment	2-11
Ordering of Selection Criteria	2-12

Overview

You define the MAINVIEW SRM system by coding global Partitioned Data Set (PDS) members and storing them in the parameter library. The global PDS members shown Figure 2-1 on page 2-2 are stored in the MAINVIEW SRM parameter library (BBPARM). They contain the specifications that define the overall characteristics of MAINVIEW SRM. For example, pool members define the composition of DASD pools, and calendar members define company holidays in the calendar year. You specify definitions in text format, and the PDS members interrelate in a hierarchy.

MAINVIEW SRM supports the sysplex environment, which means you can configure your system for a single-system environment or a shared parmlib environment.

Figure 2-1 MAINVIEW SRM Definition Members



System Configuration Members

A system configuration consists of a SMMSYS master system member, which includes global definition members. Multiple MAINVIEW SRM system configurations can reside in BBPARM. A fully functional MAINVIEW SRM system contains at least a single configuration of the global members, which are read during MAINVIEW SRM start up:

- The master system member, SMMSYS, is read first. SMMSYS $_{xx}$ specifies the versions of the other global definition members to include in the particular configuration of MAINVIEW SRM that is being started:
 - Pool (SMPOOL $_{xx}$)
 - SMS subpool (SMSPOL $_{xx}$)
 - Function (SMFUNC $_{xx}$)
 - Calendar (SMCAL $_{sxx}$)
 - Variable (SMVAR $_{sxx}$)
 - Diagnostic (SMDIAG $_{xx}$)
- Additional configuration members reside in BBPARM.SMMSYS $_{xx}$. These members specify the versions of the members to include in the instance of MAINVIEW SRM that is being started:

- Command (SMACMDxx) for SG-Auto
- Event (SMEVNTxx) for Enterprise Storage Automation
- VTOC scan (SMVSCFxx) for StorageGUARD

Multiple versions of the parmlib members, each containing different specifications, can reside in BBPARM. However, only one set is in use by MAINVIEW SRM at any one time.

In a shared parmlib environment, you can implement and maintain MAINVIEW SRM systems by coding statements in a shared OS/390 PDS. For more information, see “Defining the Sysplex Environment” on page 2-10.

Parmlib Member Descriptions

The following list describes each type of MAINVIEW SRM definition member.

Note: The first six characters of each member name are reserved by MAINVIEW SRM, and the last two are specified by the user.

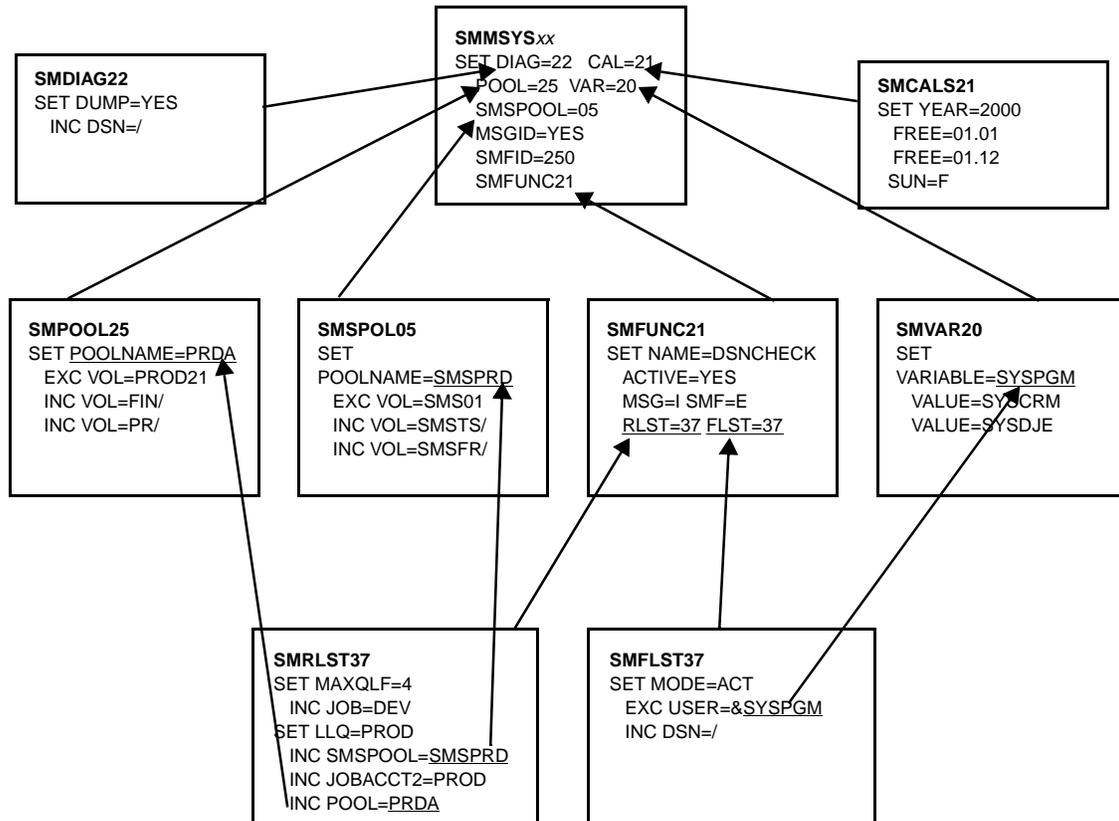
Member	Description
SMMSYSxx	contains <i>all global process options</i> that apply to the overall functionality of MAINVIEW SRM SMMSYSxx specifies default device type information, message control information, passwords, and SMF IDs. During startup, the SMMSYSxx member is read before other configuration members. SMMSYSxx specifies the other definition members (see Figure 2-1 on page 2-2) to include in the MAINVIEW SRM configuration that is being started.
SMDIAGxx	specifies various diagnostic options, such as tracing of filter and rule list processing or other debugging options, that may be requested by BMC Software Customer Support
SMPOOLxx	defines pools of DASD volumes and pools of tape devices
SMSPOLxx	defines subpools of SMS-managed DASD volumes SMS subpools are only used by EasyPOOL.
SMCALsxx	defines non-working days (weekends, public holidays, company-specific holidays, and so on) to assist in DFHSM migration
SMVARsxx	defines MAINVIEW SRM variables and assigns values to those variables MAINVIEW SRM variables can be substituted later for MAINVIEW SRM parameter values.
SMFUNCxx	activates MAINVIEW SRM functions and controls message and tracing activity MAINVIEW SRM space management services are divided into <i>functions</i> . Functions provide all runtime services of MAINVIEW SRM. SMFUNCxx points to members SMFLSTxx and SMRLSTxx, which select resources and control the operation of the functions. With the exception of the USERVARS function, functions and the filter lists and rule lists that are used by a function are specific to individual components.

Member	Description
SMFLSTxx	selects the resources on which a MAINVIEW SRM function operates Resources can be selected by data set name, volume name, unit name, data set type, job name, account number, or a number of other characteristics. For example, all data sets with names that begin with GL225, or all jobs with a name that ends with TST may be selected. A powerful name-masking facility allows flexibility in identifying the resources to select.
SMRLSTxx	defines rules that control the action of the MAINVIEW SRM function on the selected resources Rule parameters allow modification of properties such as blocksize, pool name, and data class, and they allow further identification of affected resources based on data set name, volume name, job name, data set size, and so on.
SMACMDxx	specifies matrix and command members that are used for storage automation SG-Auto uses a command member or a combination of a command member and matrix members to perform the functions of monitoring system usage and automating procedures for corrective action. Matrix members are called by command members.
SMEVNTxx	defines how storage event notices are managed in Enterprise Storage Automation Event member parameters include mode, destination, severity, and so on.
SMVSCFxx	contains information used in processing requests to collect VTOC scan data

MAINVIEW SRM Parameters

Specifications in the parmlib members are in the form of statements containing parameters. Parameters are either *action parameters* or *selection parameters*. Each parameter has a keyword followed by a value.

Figure 2-2 Connecting MAINVIEW SRM Definition Member Parameters



You can use selection parameters (when they are available) for any function, but action parameters are generally unique to a function. For instance, **LIMIT** (used to set a size limit on a new data set allocation) can only be set by the **SPACLIMI** function and **STOGROUP** (storage group) can only be set by the **SMSACSSG** function.

Action Parameters

Action parameters specify values that either control MAINVIEW SRM operation or set a value that is subsequently used by OS/390 or DFSMS. For example:

Example

```
VOL=(PRGL01, PRAP11)
```

tells MAINVIEW SRM which volumes comprise a pool

```
DATACLAS=VSAMLINR
```

sets the DFSMS data class for a new data set

```
EXPDT=96001
```

sets the expiration date for a new data set

```
MSG=E
```

tells MAINVIEW SRM to generate only error messages

You specify action parameters on SET statements (statements that begin with the keyword SET).

Selection Parameters

Resource selection is defined by selection parameters in filter list and rule list members. You also use selection parameters in pool definitions to specify the components of a pool.

You specify selection parameters on INC (include) or EXC (exclude) statements. Selection parameter values are compared to OS/390 properties; if the selection parameter is equal to the OS/390 value, the resource is selected for processing by the MAINVIEW SRM function. For example:

Example

```
DD=SORTOUT
```

selects data sets with a DD name of SORTOUT

```
JOBACCT2=TEST
```

selects data sets in jobs with a second job account field that contains TEST

```
MAXSIZE>100MB
```

selects data sets with a maximum possible size exceeding 100 megabytes

```
SMSMANAGED=NO
```

selects data sets that are not under DFSMS control

```
VOL=PROD23
```

selects data sets that reside, or will reside, on volume PROD23

The flexibility of resource selection allows a function to operate on every data set in the data center, on a single data set for a single user, or on any group of data sets in between these two extremes.

Filter Lists

A filter list selects resources for one or more functions and defines the processing mode of those resources to be active, inactive, or simulate. You can use multiple selection criteria in a filter list. The function definition identifies the filter list to be used. A filter list for the DSNCHECK function might look like this:

Example

```
SET MODE=SIM
```

```
INC XMODE=JOB
```

selects all data sets in jobs that are running in batch mode

The DSNCHECK function for these resources is to run in simulate mode.

```
SET MODE=ACT
```

```
INC XMODE=TSO
```

selects all data sets in jobs that are running from TSO

The DSNCHECK function for these resources is to run in active mode.

Rule Lists

A rule list selects resources that are modified with specific values in action parameters. A rule list is invoked for the group of resources already selected by the filter list; individual rules in the list can then select all or some resources from that group for processing. You can specify multiple rules in a rule list. The function definition identifies the rule list to use. A rule list for the SPACLIMI function might look like this:

Example

```
SET LIMIT=10MB  
INC DISP2=KEEP
```

sets an upper size limit of 10 megabytes on data sets with a normal termination disposition of KEEP

```
SET LIMIT=100MB  
INC JOBACCT2=PR
```

sets an upper size limit of 100 megabytes on data sets with a second job account field that is PR

Resource selection statements can include or exclude resources. INC statements select a resource for processing if the resource meets the criteria in the selection parameters. EXC statements exclude a resource from processing if it meets the criteria in the selection parameters.

Name Masking

The value you specify for a parameter can be a number (LIMIT=100MB), a literal YES or NO (REPLACE=YES), a single character (MSG=I), or a character field (data set names, volume names, job names, data classes, job account fields, program names, pool names, RACF groups, and so on).

Character field parameters, or name parameters, can be specified in full (JOB=PGLYTD25) or can use *name masking* to specify a partial name (JOB=PGL/). Name masking allows resource selection parameters to select large or small groups of resources; for example:

Example

```
INC DSN=HR25.MSTR.YTD
```

selects the single data set HR25.MSTR.YTD

```
INC DSN=*.MSTR.YTD
```

selects all data sets with the final two name qualifiers MSTR.YTD, regardless of the first name qualifier

```
INC DSN=**.MSTR.YTD
```

selects all data sets with the final two name qualifiers MASTER.YTD, regardless of any beginning qualifiers

Name masking cannot be used with parameters requiring a numeric value or one of a predefined set of values.

Variables

The variables definition facility further extends the capabilities of resource selection. A variable can be defined with multiple values and substituted in any selection parameter; for example:

Example

```
SET VARIABLE=SYSTLIB  
VALUE=ISP.V*.*LIB  
VALUE=ISR.V*.*LIB  
VALUE=SYS?.*LIB
```

defines the variable SYSTLIB to contain the three data set name specifications

(in a filter list)

```
INC DSN=&SYSTLIB
```

selects data sets as if these selection parameters were specified:

```
INC DSN=ISP.V*.*LIB  
INC DSN=ISR.V*.*LIB  
INC DSN=SYS?.*LIB
```

Defining the Sysplex Environment

MAINVIEW SRM uses BBI-3 architecture and supports the sysplex environment. The information in this section helps you understand how to define a shared parmlib environment.

In a shared parmlib environment, you can implement and maintain MAINVIEW SRM systems by coding statements in a shared OS/390 partitioned data set (PDS). MAINVIEW SRM parmlib members support new INC/EXC parameters for the SET statement and a new override capability to certain SET statements.

You may choose to share SMMSYS_{xx} but not SMPOOL_{xx} or other member(s). Sharing parmlib members requires a well planned naming convention so that pools do not have the same name but different characteristics on different systems.

Sysplex Parameters

The same parameters that are used in the non-shared environment are supported in the shared environment. To accommodate a shared parmlib environment, three INC/EXC keywords, FORSYSID, FORSMFID, and FORPLEXNAME are valid in the following members:

- SMMSYS_{xx}
- SMFUNC_{xx}
- SMPOOL_{xx}
- SMFLST_{xx}
- SMRLST_{xx}
- SMDIAG_{xx}
- SMEVNT_{xx}
- SMCRT_{xx}
- SMCALS_{xx}
- SMSPOL_{xxx}
- SMVARS_{xx}

The SET statement can be overridden, which means that you can code an initial SET statement in the shared SMMSYS_{xx}, SMFUNC_{xx}, SMPOOL_{xx}, SMDIAG_{xx}, SMEVNT_{xx} and SMCRT_{xx} members, followed by INC/EXC parameters for each image in the sysplex.

The last INC/EXC statement that is coded in a SET statement is accepted as the override value. Previously coded INC/EXC statements are ignored.

Syntax Considerations for the Shared Parmlib Environment

If no INC FORSYSID is coded on a SET statement, the SET statement applies to all systems in the sysplex.

If multiple SET POOLNAME parameters are found in a shared member for the same POOLNAME, the last value that is coded for the parameter overrides all other values. INC/EXCs cannot be overridden; that is, they are not allowed on multiple SET statements.

Example

```
SET POOLNAME=DOUG SGDCOLLECT=NO
INC VOL=BAB/
INC VOL=SHK/
INC SYSID=*
SET POOLNAME=DOUG SGDCOLLECT=YES
INC SYSID=SJSG
```

This rule allows you to override the SET parameters only, assuming that all volumes in a given pool should be the same on all images in a sysplex. You cannot define pools with the same name but containing different volumes on different images.

Ordering of Selection Criteria

There is no change to selection criteria ordering logic; however, INC FORSYSID, FORSMFID, and FORPLEXNAME default to the current image unless an include statement for one of the three is coded. In that case, the resulting selection will prevail.

Example

```
INC FORSYSID=SYSG
```

results in the selection of system SYSG only

```
EXC FORSYSID=SYSG
```

results in the selection of all images in the sysplex, except SYSG.

Chapter 3 Understanding General Syntax Rules

This chapter provides an overview of the allowable structures of the statements and parameters used in MAINVIEW SRM. The following information is included:

Statements and Parameters	3-1
Parameter Relationships	3-3
Comparison Operators	3-4
Name Masking	3-5
Field-to-Field Comparisons	3-9
Selection Criteria Ordering.	3-9

Statements and Parameters

You code the various specifications that control MAINVIEW SRM in SET statements in the library members. A SET statement is composed of the keyword SET and one or more parameter specifications. A SET statement begins with the SET keyword and continues until the next SET keyword is found or the end of the member is reached.

The following rules apply:

- All text must be uppercase.
- The SET keyword must precede all parameters. There are no column restrictions. For example, the following statements are all valid:

```
column 1
|
SET POOL=TEST
    SET POOL=TEST
SET POOL = TEST
```

- Parameters are composed of keywords and associated values. There are two types of parameters: action and selection. Action parameters indicate how the MAINVIEW SRM function operates. Selection parameters indicate which resources (volumes, jobs, data sets, and so on) are affected by the MAINVIEW SRM function.
- Action parameters must be placed immediately after the SET operation keyword. For members SMPOOLxx, SMFLSTxx, and SMRLSTxx, all action parameters must be on the same line or a continued line. For other members (SMMSYSxx, SMCALSxx, SMVARSxx, and SMFUNCxx), action parameters can be listed on separate lines.

In all members the first line *must* be preceded with SET. In SMMSYS and SMEVNT, lines that are *not* preceded by SET, INC, or EXC default to SET (except continuations). In all other members, lines that are *not* preceded by SET, INC, or EXC default to INC (except continuations).

- Selection parameters must follow the action parameters and start on a new line. Selection parameters can be preceded by the keyword INC or EXC to indicate inclusion or exclusion (the default is INC).
- The parameter value is separated from the parameter keyword by an equal sign (=) and in some cases, a greater than (>) or less than (<) sign; spaces are optional.
- Most parameters have a single value specification: a keyword, a name or a name mask, or a number; for example:

```
MODE=ACT
EXC PGM=GL0125A
EXTENT>5
```

- All selection parameters and some action parameters may have a list of values, in which case the individual values are enclosed in parentheses and separated by commas or blanks:

```
POOL= ( TEST , WORK , PROD )
```

- Parameter specifications within a single line are separated by commas or spaces; for example:

```
SET POOL=WORK , USEVOL=ALL
SET POOL=WORK USEVOL=ALL
```

- Variables are concatenated by placing them together, without the use of any special connective characters. Blanks or commas cannot be placed between concatenated variables; for example:

```
DSN=&TESTHLQ&DIVCODE&TESTLLQ
```

- A comment line is indicated by an asterisk in column 1. Comments can be embedded in a line by starting the comment with /* and ending it with */.
- Blank lines are allowed.
- In SMPOOL_{xx} or a filter or rule list, a SET statement action parameter or selection parameter can be continued onto the next line by placing any character in column 72.

Parameter Relationships

Action parameters are independent of each other. They can be specified in any order without affecting their operation. For members SMFLST_{xx} and SMRLST_{xx}, all action parameters for a single SET statement must be on the same line (or on a continued line). For other members, action parameters can be on separate lines.

Selection parameters are cumulative; each selection parameter within a statement is modified by subsequent selection parameters. Selection parameters on a single line are related by an AND condition, whereas selection parameters on separate lines are related to preceding lines by an OR condition. For example:

```
VOL=TEST01 , DSN=APPROD . VNDR . MASTER
```

indicates that the volume ID must be TEST01 *and* the data set name must be APPROD.VNDR.MASTER.

```
VOL=TEST01
DSN=APPROD . VNDR . MASTER
```

indicates that the MAINVIEW SRM function is applied if *either* the volume ID is TEST01 *or* the data set name is APPROD.VNDR.MASTER.

Multiple values for a single selection parameter are related by an OR condition; that is, when several values are specified for a single selection parameter by enclosing the list of values in parentheses, any of the specified values will satisfy the selection. The following examples show how this works:

DSN2 = (DBYTD, DBQTR, DBDLY)

indicates that the second name qualifier must be DBYTD or DBQTR or DBDLY.

When multiple selection parameters with value lists are specified, the selection will be satisfied by a combination of OR and AND conditions.

DSN2 = (DBYTD, DBQTR) DSN3 = (RECEIPT, ORDER)

indicates that the MAINVIEW SRM function is applied if the second name qualifier is either DBYTD or DBQTR and if the third name qualifier is either RECEIPT or ORDER.

Selection parameter specifications are usually contained within a single line; however, they can be continued onto multiple lines. Continuation is indicated by a non-blank character in column 72. Therefore, parameter specifications continued on the next line are treated as being connected by AND logic. For example:

	column 72
DSN=APPROD.VNDR.MASTER	X
VOL=WORK	

indicates that the data set name must be APPROD.VNDR.MASTER and the volume ID must be WORK for the MAINVIEW SRM function to be applied. If the second line was not a continuation line, the two parameters would be related by OR logic.

Comparison Operators

Selection parameters may use any of the following comparison operators:

= equal
value of the parameter at run time must equal the specified value

- < less than
value of the parameter at run time must be less than the specified value
- > greater than
value of the parameter at run time must be greater than the specified value
- ≠ not equal to
value of the parameter at run time must be a value other than the specified value (the logical NOT symbol preceding the equal symbol is EBCDIC X'5F')

Operators for “less than or equal to” and “greater than or equal to” are not supported.

Name Masking

Specification of names, or any character-field value, in any MAINVIEW SRM parameter can use the complete name, or a partial name with special mask symbols to build flexible criteria.

Table 3-1 lists the mask symbols that are supported.

Table 3-1 Name Masking Symbols (Part 1 of 2)

Symbol	Explanation	Examples
%	specifies a single numeric character Multiple mask characters can be specified and the mask can be embedded within text.	<ul style="list-style-type: none"> VOL=PRODV% condition that would be satisfied by PRODV1 or PRODV9 (but not PRODVX). VOL=TST%V% condition that would be satisfied by TST5V1 or TST2V9 (but not TSTAV5 or TST44V). VOL=WRK%%% condition that would be satisfied by WRK104 or WRK582 (but not WRK33).
?	specifies any character except a period (.), including numeric, alphabetic, or special (such as #) characters Multiple mask characters can be specified, and the mask can be embedded within text.	<ul style="list-style-type: none"> VOL=PR???? condition that would be satisfied by PRODV1 or PR45B6 (but not PR64). VOL=PR??9 condition that would be satisfied by PR4B9 or PR#39 (but not PROD4)

Table 3-1 Name Masking Symbols (Part 2 of 2)

Symbol	Explanation	Examples
/	<p>specifies zero or more characters of any type at the end of a name</p> <p>This mask character terminates a name mask and any characters that follow are ignored.</p> <p>Note: A period is significant in a DSN name mask. The parameter DSN=TEST./ differs from DSN=TEST/</p>	<ul style="list-style-type: none"> • VOL=TST/ condition that would be satisfied by TST, TSTA, TSTB14, or TST40V (but not TSXX44) • VOL=TST%/ condition that would be satisfied by TST4BX or TST477 (but not TSTA64) • VOL=/ condition that would be satisfied by any combination of characters • DSN=TEST/ condition that would be satisfied by TEST.CICS.PPT (but not PROD.CICS.PPT) • DSN=TEST./ condition that would be satisfied by TEST.CICS.PPT (but not TESTX.CICS.PPT)
*	<p>specifies any character string in a single name node</p> <p>This mask character can be embedded within other text.</p> <p>Note: Leading asterisks can be confusing. Name matching stops after the name mask has been exhausted, even if more characters remain in the name. LLQ=*T selects LAST but not TEST. The asterisk requires a non-zero number of characters. LAST has three characters (matching the asterisk) before the T (matching the T in the mask). TEST, however, has zero characters before a T is encountered; the initial T in TEST matches the T in the mask *T, but because no characters precede the T, the match fails.</p>	<ul style="list-style-type: none"> • DSN=AP1177A.*.MASTER condition that would be satisfied by AP1177A.VNDR.MASTER or AP1177A.VOUCHER.MASTER (but not AP1177A.VNDR.TEST.MASTER or AP1177A.MASTER) • DSN=AP117A.PER*.MASTER condition that would be satisfied by AP117A.PER11.MASTER or AP117A.PER07A.MASTER (but not AP117A.MON12.MASTER) • DSN=AP117A.*TST.MASTER condition that would be satisfied by AP117A.YEARTST.MASTER or AP117A.TST.MASTER (but not AP117A.DAILY.MASTER)
**	<p>specifies any non-zero number of name nodes</p> <p>This mask character can be embedded within other text.</p> <p>Note: Using ** at the end of a name string is equivalent to using /. For example, AP.TEST.MAS** is equivalent to AP.TEST.MAS/. The / mask, however, cannot be used with following characters, as it terminates the mask. The value of the ** mask is that it can be followed by more name nodes.</p>	<ul style="list-style-type: none"> • DSN=AP**.MASTER condition that would be satisfied by AP.VNDR.MASTER or AP.VNDR.TEST.MASTER or AP.VNDR.TEST.YTD.MASTER (but not AP.VNDR.MASTER.TEST or AP.MASTER) • DSN=**.TEST.CICS condition that would be satisfied by GLYTD5.TEST.CICS or GLYTD.PER04.TEST.CICS (but not TEST.CICS) • DSN=AP.TST**D.MASTER condition that would be satisfied by AP.TST002.YTD.MASTER (but not AP.PR002.YTD.MASTER)

Variables in Name Masks

The name of a variable can be used in any non-numeric MAINVIEW SRM selection parameter and in action parameters for the DSNCHECK function. The values of the variable are substituted into the parameter before the parameter is evaluated.

Variables are defined in the SMVARS $_{xx}$ parmlib member. A variable can contain multiple values. For example, the variable APTEST is defined as follows:

```
SET VARIABLE = APTEST
  VALUE = APTEST.*.PERIOD12
  VALUE = APTEST.MASTER./
```

A selection parameter using the APTEST variable might be used as follows:

```
INC DSN = &APTEST
```

This include statement is equivalent to the following statements:

```
INC DSN = APTEST.*.PERIOD12
INC DSN = APTEST.MASTER./
```

Variables may be concatenated together. For example, the following variables are defined in SMVARS $_{xx}$:

```
SET VARIABLE = APTESTHLQ
  VALUE = APTEST
  VALUE = APTST
  VALUE = TESTAP*
SET VARIABLE = DIVISIONCODE
  VALUE = RBK
  VALUE = CRM
  VALUE = DSR
SET VARIABLE = APLLQ
  VALUE = TRAN/
  VALUE = UPD/
  VALUE = MASTER*
```

The variables from the previous example may be concatenated as follows in an FLST or RLST statement:

```
INC DSN = &APTESTHLQ&DIVISIONCODE&APLLQ
```

Blanks or commas are not allowed between concatenated variable names.

The following rules about using variables apply:

- Variables cannot be used with numeric parameters such as BLKSIZE or SECSPACE.
- Variables can be used alone or concatenated with other variables. Variables *cannot* be mixed with text or name mask characters. For example:

```
INC DSN = &PRODHLQ                is valid
INC DSN = &PRODHLQ&ACCT            is valid
INC DSN = &PRODHLQ/                is invalid
INC DSN = &PRODHLQ.*.&PRODLLQ      is invalid
INC DSN = &PRODHLQ*.GLYTD.TRAN     is invalid
```

- Variables can contain any text and any mask characters. To use mask characters with variables, the mask characters must be defined within the variable. For example:

```
SET VARIABLE = PRODHLQ
VALUE = ACCT*.*.*.
SET VARIABLE = PRODLLQ
VALUE = DAILY/
VALUE = MONTHLY/
INC DSN = &PRODHLQ&PRODLLQ
```

is the equivalent of:

```
INC DSN = ACCT*.*.*.DAILY/
INC DSN = ACCT*.*.*.MONTHLY/
```

- Variable names can be from 1 to 30 characters long, plus a preceding ampersand character (&). For example:

```
SET VARIABLE = FULLDATASETNAME
VALUE = . . .
INC DSN = &FULLDATASETNAME
```

- The ampersand character can only be used preceding a variable name in a selection or action parameter. It cannot be used as part of a text string since it flags the beginning of a variable name.
- Variables are concatenated by placing them together, without the use of any special connective characters. Blanks or commas cannot be placed between concatenated variables. For example:

```
DSN=&TESTHLQ&DIVCODE&TESTLLQ
```

- Variable definitions cannot be nested; that is, a variable definition that contains a variable is invalid.

Field-to-Field Comparisons

Selection parameters can be compared to values in other selection parameters by preceding the target operand (on the right side of the equal sign) with an exclamation point. For example, comparisons allow you to select data sets where the high-level qualifier is equal to the user name or where the job name is equal to the user name. For example:

INC HLQ = !DSN2	includes all data sets where the high-level qualifier is the same as the second data set name node
INC HLQ = !USER	includes all data sets where the high-level qualifier is the same as the user name
EXC JOB = !USER	excludes all data sets where the job name is the same as the user name

The exclamation point is required to recognize the field-to-field comparison. If the exclamation point is omitted, the information to the right of the equal sign is interpreted as text:

INC HLQ = !USER	compares the high-level data set name qualifier to the user ID
INC HLQ = USER	compares the high-level data set name qualifier to the literal USER

Field-to-field comparisons work for all selection parameters (INC and EXC statements) in all function FLSTs and RLSTs. Field-to-field comparisons do not work with action parameters (SET statements).

Selection Criteria Ordering

To use MAINVIEW SRM successfully, you must understand the processing flow of selection parameters fully. Resources to be affected by MAINVIEW SRM functions are selected or excluded based on the selection parameters, including the order and interrelationships (OR, AND) of the selection parameters.

Selection parameters are processed from top to bottom; as soon as a match is made to the selection parameters, the selection process stops. MAINVIEW SRM takes the first match to selection parameters. Therefore, selection parameters should be listed from specific to general. For example, the following specification selects all data sets with PROD1 as the first qualifier, except for those that begin with PROD1.TEST:

```
EXC DSN=PROD1.TEST/  
INC DSN=PROD1./
```

The separation onto two lines defines an OR relationship between the two criteria. A data set named PROD1.MASTER.FINAL is selected by the INC parameter, since it does not match the criteria in the EXC parameter but it does match the INC parameter. However, a data set named PROD1.TEST.MASTER would not be selected; the EXC parameter would be satisfied first and the data set excluded.

If the sequence were reversed, as in:

```
INC DSN=PROD1 . /  
EXC DSN=PROD1 . TEST/
```

PROD1.TEST.MASTER would not be excluded. The INC selection parameter would be satisfied (because PROD1.TST.MASTER begins with PROD1.), and processing would be halted before the EXC parameter was processed.

Definitions can vary and can be complicated. Some key points to remember are

- Selection parameters on the same line (or on continued lines) are related with an AND condition.
- Selection parameters on separate lines (without line continuation) are related with an OR condition.
- Selection parameters should be listed from specific to general, because selection processing *stops* as soon as any INC condition is fully satisfied. A resource that has met the criteria in an INclude statement is selected for processing and is not considered by any other filters or rules in the list. However, a resource that has met the criteria in an EXclude statement is excluded from the current SET statement only and is passed to the next SET statement in the filter/rule list for consideration.

Consider the following points in defining resource selection:

- An exclude without an include will not select any resources. For example:

```
SET SPACVOLA=50  
EXC DD=TRANSP/
```

will exclude all data sets with a DDname beginning with TRANSP. However, since there is no include, no resources will be selected for this rule; that is, there is no default include following an exclude.

If the intent is to select all resources except TRANSP DD names, an include such as INC DD=/ should follow the exclude.

This principle applies to both filter and rule list entries.

- A SET statement without an include or exclude will be ignored. For example:

```
SET VIO=YES  
SET VIO=NO  
INCLUDE SIZE<500KB
```

The first SET (VIO=YES) will never be executed, because no resources are included. All resources are passed to the second SET statement (VIO=NO) for consideration.

This principle applies to both filter and rule list entries.

Chapter 4 Using MAINVIEW SRM Operator Services

This chapter provides an overview of the operator services used in MAINVIEW SRM. The following information is included:

Overview	4-2
Activating System Software in OS/390	4-2
Communicating with MAINVIEW SRM Components	4-3
Common Command Structure	4-3
Controlling MAINVIEW SRM Components	4-3
Managing the Storage Product Address Space	4-4
Starting and Stopping MAINVIEW SRM	4-4
Starting SVOS	4-4
Starting MAINVIEW SRM Components	4-5
Stopping MAINVIEW SRM Components	4-7
Stopping SVOS	4-8
Starting and Stopping MAINVIEW SRM with Other Products	4-8
SVOS Command Syntax Conventions	4-9
Entering Commands	4-9
Entering Commands from the PARMLIB DD Statement	4-10
SVOS Command Definitions	4-11
DUMP Command	4-11
FORCE Command	4-12
SHUTDOWN Command	4-12
START Command	4-13
STATUS Command	4-14
STOP Command	4-15
SVOS Execution JCL Description	4-16
Parameter Descriptions	4-16
The SVOS Console Interface	4-18
START/STOP Commands	4-19
DISPLAY Commands	4-19
REFRESH Commands	4-20
SET Commands	4-21

Overview

MAINVIEW SRM operator services (SVOS) is a started task that runs all MAINVIEW SRM products. SVOS is the PAS that manages the connection and communication with the CAS. The SVOS PAS provides

- connection for real-time monitor, EasyHSM, administration, SG-Control, and common views
- dynamic activation and deactivation of MAINVIEW SRM components
- communication with MAINVIEW SRM components

In addition to explaining these functions in detail, this chapter includes information about

- SVOS commands
- SVOS JCL
- console interface

Note: With the following exceptions for SG-Control, SVOS must remain active while any of the MAINVIEW SRM products are running:

- SGCMAINT INIT function
- SGCMAINT REPORT function
- SGCMAINT COMPARE function
- BUDGET command

Activating System Software in OS/390

There are two methods of activating system software products:

Static Activation

Usually, this method uses SMP to link a USERMOD into the operating system. The modification is actually applied to system libraries and an IPL usually is required to activate the product. While this is a safe method of activating system software products, there are serious system maintenance drawbacks. Manual intervention is required by system programmers whenever maintenance is applied to modified CSECTS. Just scheduling the SMP work, and an IPL, often can be a problem.

Dynamic Activation

This method requires a sophisticated activation process. SVOS is a started task that performs this process for MAINVIEW SRM components. SVOS loads and controls the various component program modules in the OS390 operating system. This started task eliminates the system maintenance problems that are involved with static modification.

Communicating with MAINVIEW SRM Components

Communication with the MAINVIEW SRM components is another function of SVOS. SVOS manages common storage, controls any required subtasks, and performs console command processing. This function of SVOS offers

- a common command structure
- control of MAINVIEW SRM components

Common Command Structure

SVOS uses an easily understood common command structure so that the MAINVIEW SRM components all respond to one set of commands. The base command is the same regardless of the component; only the component identifier code is specific to a component.

Note: The global system parameter, BBI3_SSID specifies the CAS subsystem name to which the SVOS PAS should connect. Since the BBI3 connection occurs during SVOS startup, SVOS will not start if BBI3_SSID is not specified. To update the value of BBI3_SSID, SVOS must be stopped and restarted; it cannot be refreshed. The CAS subsystem name is specified in the SSID parameter on the PARM keyword for the CAS JCL EXEC statement.

Controlling MAINVIEW SRM Components

Using SVOS commands, MAINVIEW SRM components can be activated or deactivated easily. Also, dynamically changing component options is easily accomplished using SVOS.

The following list identifies each component or set of components for MAINVIEW SRM:

Component	Identifier
SVOS	SVOS
StorageGUARD (historical space)	SVSGD
StorageGUARD (historical performance)	SVSGP
SG-Auto	SVSGA
EasySMS, EasyPOOL, StopX37/II	SVALLOC
EasyHSM	SVHSM
Enterprise Storage Automation	SVESA

Managing the Storage Product Address Space

There are three product address spaces (PAS) for the MAINVIEW SRM suite of storage products. The MAINVIEW SRM common PAS, SVOS, must be started first. After which, the SVSGD and SVSGP can be started, provided your site is licensed for StorageGUARD. Only one SVOS started task can be active on a system at any time.

This chapter provides instructions for starting and stopping the SVOS PAS.

Starting and Stopping MAINVIEW SRM

The following sections describe how to start and stop the SVOS PAS.

Starting SVOS

Sample JCL for SVOS is in the SVOS member of the *?prefix.BBSAMP* data set. See “SVOS Execution JCL Description” on page 4-16 for descriptions of start-up statements and descriptions of the parameters that are used to pass installation-dependent options to the program.

SVOS is started with the following OS/390 command:

```
S SVOS
```

After a successful start up of SVOS, the following message is displayed on the console:

```
SVO0106 RELEASE v.r.m OF SVOS HAS BEEN STARTED
```

The SVOS component must be the *first* component started. At SVOS startup, the parmlib members are read and processed and made available to all other components.

Note: When the SVOS component is started, the connection to BBI3 is made. If the BBI3_SSID parm in SMMSYSxx is not specified, the connection cannot be made and the ADMEM view is not available. You must edit the SMMSYSxx member and add the BBI3_SSID parameter. Then, either stop and start SVOS or perform a system refresh.

SVOS is designed to start successfully, even if parameter errors are present. If there are parameter errors, access the Parmlib Members option (ADMEM view), correct the errors in the appropriate member(s), and perform a refresh. The other components of MAINVIEW SRM may be started once the corrected member(s) are refreshed.

If there is attempt to start another component after SVOS has had a parameter error, the startup attempt will fail and messages will point to the first SVOS that was started.

Starting MAINVIEW SRM Components

After SVOS startup, the MAINVIEW SRM components can be started. There are eight separate start up commands for MAINVIEW SRM, one for each of the following components or set of components:

Component	Command
SVOS	/S SVOS,SUF=00
SG-Auto	/S SVSGA
EasySMS, EasyPOOL, StopX37/II	/S SVALLOC
EasyHSM	/S SVHSM
Enterprise Storage Automation	/S SVESA
StorageGUARD Historical Performance	/S SVSGP
StorageGUARD Historical Space	/S SVSGD
StorageGUARD Historical Space Alternate Data Collector	/S SVSGDn (n is a number, in the range 1–8, that identifies the alternate data collector)

To start MAINVIEW SRM components, you can use any of the following methods:

- The MAINVIEW SRM component startup commands can be entered from the console by preceding the command with the SVOS command character.
- SVOS can be configured to start MAINVIEW SRM components automatically at start up.
- You can use the COMPSTAT view to manually activate components.

After a successful start up of the MAINVIEW SRM component(s), the following message(s) are displayed on the console indicating that MAINVIEW SRM is active:

```
SVO0106 RELEASE v.r.m OF SVOS HAS BEEN STARTED
```

```
SVO0610 MAINVIEW SRM/SGD REL v.r.m HAS BEEN STARTED  
ON sybsys
```

```
SVO0610 MAINVIEW SRM/HSM REL v.r.m HAS BEEN STARTED  
ON sybsys
```

```
SVO0610 MAINVIEW SRM/SGP REL v.r.m HAS BEEN STARTED  
ON sybsys
```

```
SVO0610 MAINVIEW SRM/SGA REL v.r.m HAS BEEN STARTED  
ON sybsys
```

```
SVO0610 MAINVIEW SRM/SGC REL v.r.m HAS BEEN STARTED  
ON sybsys
```

```
SVO0610 MAINVIEW SRM/SVESA REL v.r.m HAS BEEN  
STARTED ON sybsys
```

```
SVO0610 MAINVIEW SRM/ALLOC REL v.r.m HAS BEEN  
STARTED ON sybsys
```

In these messages, *v. r. m* is the version, release, and modification number and *sybsys* is the sub-system name that is used in SUBSYS parameter in SVOSJCL (the default is ETIS).

Stopping MAINVIEW SRM Components

All MAINVIEW SRM components must be stopped *before* SVOS is stopped. Type the appropriate command(s) on the console to stop the component(s):

Component	Command
StorageGUARD (historical space)	/P SVSGD
StorageGUARD (historical performance)	/P SVSGP
SG-Auto	/P SVSGA
EasySMS, EasyPOOL, StopX37/II	/P SVALLOC
EasyHSM	P SVHSM
Enterprise Storage Automation	/P SVESA
SVOS	/P SVOS

In these commands, / is the SVOS command character that is specified in SUBCHAR parameter in SVOS JCL.

After a successful shutdown of the MAINVIEW SRM components, the following message(s) are displayed on the console:

```
SVO0620 MAINVIEW SRM/SGD REL v.r.m HAS BEEN STOPPED
ON subsys
```

```
SVO0620 MAINVIEW SRM/SGP REL v.r.m HAS BEEN STOPPED
ON subsys
```

```
SVO0620 MAINVIEW SRM/SGA REL v.r.m HAS BEEN STOPPED
ON subsys
```

```
SVO0610 MAINVIEW SRM/SGC REL v.r.m HAS BEEN STARTED
ON subsys
```

```
SVO0610 MAINVIEW SRM/SVESA REL v.r.m HAS BEEN
STARTED ON subsys
```

```
SVO0620 MAINVIEW SRM/ALLOC REL v.r.m HAS BEEN
STOPPED ON subsys
```

In these messages, *v. r. m* is the version, release, and modification number and *subsys* is the sub-system name that is used in SUBSYS= parameter in SVOSJCL (the default is ETIS).

Instead of shutting down the components individually, you may shut them down at one time by using the SVOS shutdown command (explained in the next section).

Stopping SVOS

SVOS is stopped *after* MAINVIEW SRM components are stopped. To stop SVOS, type the following SVOS command on the console:

/SHUTDOWN

In this command, / is the SVOS command character that is specified in SUBCHAR parameter in SVOS JCL.

The SHUTDOWN command stops SVOS and any BMC Software products that are running under SVOS. No SVOS messages are written to the console on a successful shutdown.

Starting and Stopping MAINVIEW SRM with Other Products

MAINVIEW SRM dynamically hooks into the operating system when it starts; when shut down, MAINVIEW SRM unhooks itself in an orderly manner. If other system management products are started that hook into the operating system in a similar manner, problems can occur on shutdown of those products or MAINVIEW SRM, depending on the order of startup and shutdown.

In general, MAINVIEW SRM should be the last product started and the first stopped. In all cases, products should be stopped in the reverse order of startup; that is, if product A is started first, followed by MAINVIEW SRM, MAINVIEW SRM should be stopped first, followed by product A.

PSP Releases 115 and earlier must be started after MAINVIEW SRM. Softworks has zaps to fix this problem; the zap IDs are SWAG011.200 and SWAG012.200. (These zaps have been added to PSP 116.) After these zaps are applied, or if you are using PSP 116 or earlier, MAINVIEW SRM can be started after PSP is initialized.

Contact the BMC Software Customer Support for the most current information on compatibility with other system management products.

SVOS Command Syntax Conventions

The following syntax conventions apply to SVOS commands. They are used throughout this chapter to explain SVOS commands and the optional parameters.

- Square brackets [] indicate that the parameters within the brackets are optional. Type only information that is shown within the brackets. Do not type the brackets.
- Braces { } indicate that you must choose among items found within the braces. Use only one item. Do not type the braces.
- A vertical bar | indicates that you must choose among the items separated by the bar(s). The bar is used in conjunction with the brackets and braces. Use only one item. Do not type the bar.
- Uppercase indicates that the field must be typed as shown. You can type it in either uppercase or lowercase.
- Lowercase indicates that the operand must be supplied to the command. You can type it in either uppercase or lowercase.
- Commands and operands can be separated with either blanks or commas.

Entering Commands

SVOS can obtain commands in three ways:

- When SVOS is being started, commands are read from the PARMLIB DD statement.
- After SVOS has been started, a MODIFY command can be used to issue commands.
- Commands can be processed with the SVOS subsystem command character.

Note: The selected subsystem character (slash, /, by default) is used in the same way as JES2 uses the dollar (\$) character to denote commands.

Entering Commands from the PARMLIB DD Statement

When the commands are read from the PARMLIB DD statement, no MODIFY information or subsystem character (/) should precede the SVOS command. Only one command per line is allowed. Any line containing an asterisk in column 1 is considered to be a comment statement. The following commands start all of the MAINVIEW SRM components:

```
*  
* START THE PRODUCTS AND THEN DISPLAY THEIR STATUS  
*  
START SVOS, SUF=xx  
START SVSGD  
START SVSGP  
START SVSGA  
START SVESA  
START SVHSM  
START SVALLOC  
STATUS
```

In these commands, the SMMSYS xx member that is used at startup.

Using MODIFY Commands

After SVOS has been started, MODIFY commands can be used to enter commands from the console. The following example shows how to use the MODIFY command:

```
F SVOS, STATUS SVSGD  
F SVOS, STOP SVALLOC
```

Using a Subsystem Character

The SVOS subsystem character can be used to enter commands from the console. The slash (/) is the default character, but the SUBCHAR parameter can be used to override the default subsystem character. To disable the SVOS subsystem character and allow the use of the MODIFY command only, specify SUBCHAR=NONE in the parameter field in the SVOS started task.

By using the subsystem character, you do not need to remember the started task name that is used by the MODIFY commands. To get the same results as shown in the MODIFY command example, type the following commands on the console:

```
/STATUS SVALLOC  
/START SVSGP  
/STOP SVSGD
```

SVOS Command Definitions

The following table lists the SVOS commands. Additional commands that are used to support a specific component of MAINVIEW SRM are documented in the user manual for the component (see “Related Documentation” on page xvi).

Table 4-1 SVOS Command Table

Command	Synonym	Purpose
DUMP		terminates SVOS with an abend
FORCE		forces SVOS to perform requested procedure
SHUTDOWN		terminates SVOS and all products running under it
START	S	initiates the requested product
STATUS	STAT	lists status information for one or all products
STOP	P	deactivates the requested product

The following sections provide more information about SVOS commands.

DUMP Command

Purpose: The DUMP command forces SVOS to terminate with a user 1000 abend.

Syntax: DUMP

Comments: This command should never be required under normal circumstances and should be used only at the request of BMC Software Customer Support.

Examples: At the request of BMC Software Customer Support, you can enter either of the following console commands to force SVOS to abend:

/DUMP

or

F SVOS,DUMP

FORCE Command

Purpose: The FORCE command forces SVOS to go through activation or deactivation processing for a product. It also can be used to reinitialize the SVOS subsystem tables.

Syntax: FORCE { *SSVT | *SSCVT | *product_name* [START | STOP] }

Comments: This command should never be required under normal circumstances. The *SSVT and *SSCVT operands should be used only at the request of BMC Software Customer Support. The *product_name* operand should be used only when a START or STOP command cannot complete. This unusual condition should occur only when a product has been partially activated. When processing the *product_name* operand, if neither START nor STOP has been requested, SVOS simply marks the product as inactive.

Examples: If SVOS had been cancelled during SVALLOC initialization, either of the following console commands will force SVOS to restart SVALLOC:

```
/FORCE SVALLOC,START
```

or

```
F SVOS,FORCE SVALLOC,START
```

At the request of BMC Software Customer Support, enter either of the following console commands to force SVOS to reinitialize the SSVT for the ETIS subsystem. This subsystem is used by SVOS to keep track of the status of the BMC Software products under its control.

```
/FORCE *SSVT
```

or

```
F SVOS,FORCE *SSVT
```

SHUTDOWN Command

Purpose: The SHUTDOWN command forces SVOS to deactivate all BMC Software products that were activated by SVOS, and then SVOS terminates.

Syntax: SHUTDOWN

Comments: If the SHUTDOWN command is placed in the parmlib member that is allocated to the PARMLIB DD statement, SVOS deactivates any products that were previously activated and no other commands in the parmlib member are processed.

Examples: To deactivate all products that are running under SVOS and terminate SVOS, use one of the following console commands:

/SHUTDOWN

or

F SVOS,SHUTDOWN

START Command

Purpose: The START command informs SVOS to perform activation for the requested component.

Syntax: START *component_name* [NOVER]]

Synonym: S

Comments: The START command normally would be placed in the parmlib member that is allocated to the PARMLIB DD statement.

The NOVER operand should never be required under normal circumstances and should be used only at the request of BMC Software Customer Support. The NOVER operand bypasses the verify procedure that is normally performed before storage is modified. If specified, STOP processing defaults to NOVER as well, and all enqueue protection on the REFRESH and STOP command for the product is bypassed. This operand can be disabled by specifying NOVERIFY=NO on the EXEC statement in the SVOS JCL.

Examples: If you are running multiple MAINVIEW SRM components, all START commands for the products should be placed in the data set that is allocated to the PARMLIB DD statement in the ELD started task:

```
START SVOS, SUF=xx
START SVALLOC
START SVSGP
START SVSGD
START SVSGA
START SVHSM
START SVESA
STATUS
```

In these commands, *xx* is the SMMSYS*xx* member that is used at startup.

STATUS Command

Purpose: The STATUS command instructs SVOS to list information about MAINVIEW SRM components.

Syntax: STATUS [*component_name* | * | VMAP] [LIST | VER] [ALL]

Synonym: STAT

Comments: A STATUS command with no parameters lists the BMC Software products that are currently active.

If a *product_name* is requested, only information for the specified product is listed.

If VER is specified, the current product version is displayed in place of the selection module information.

If the LIST operand is specified, information that is used by SVOS to control a product is displayed. This information normally is used only by BMC Software Customer Support.

If ALL is specified, the status for all SVOS subsystems is displayed for each product, using the LIST or VER option as supplied.

Note: If VMAP is specified, *product_name* cannot be specified. ALL can be specified to produce a VMAP of all SVOS subsystems. LIST or VER is ignored if specified.

Examples: To start all MAINVIEW SRM components during SVOS initialization and then display the current status for each product, the data set that is allocated to the PARMLIB DD statement could contain the following commands:

```
START SVOS, SUF=xx
START SVALLOC
START SVSGP
START SVSGD
START SVSGA
START SVHSM
STATUS
```

In these commands, *xx* is the SMMSYS*xx* member that is used at startup

To display a virtual storage map of all SVOS subsystems, issue the following console command:

```
/STATUS VMAP ALL
```

To display a virtual storage map of all SVOS subsystems and to see the version number of SVOS, issue the following console command:

```
/STATUS VMAP ALL VER
```

To display the status of all products across all SVOS subsystems, issue the following console command:

```
/STATUS ALL
```

STOP Command

Purpose: The STOP command instructs SVOS to deactivate the requested product.

Syntax: STOP *component_name* [VER | NOVER]

Synonym: P

Comments: The STOP command should never be placed in the parmlib member that allocated to the PARMLIB DD statement.

To deactivate all MAINVIEW SRM components and terminate SVOS, the SHUTDOWN command should be used.

The VER and NOVER operands should never be required under normal circumstances and should be used only at the request of BMC Software Customer Support. Unless the associated START command requested NOVER, STOP commands will perform a verify by default before changing memory. This operand can be disabled by specifying NOVERIFY=NO on the EXEC statement in the SVOS JCL.

Examples: If you want to deactivate SVSGD, one of the following console commands can be used:

```
/STOP SVSGD
```

or

```
F SVOS,STOP SVSGD
```

SVOS Execution JCL Description

This section describes the execution JCL that is used by SVOS.

Statement	Use
PROC	procedure statement for the started task
EXEC	specifies the program name for SVOS (PGM=ETILOADR)
STEPLIB DD	<p>defines one or more input partitioned data sets</p> <p>These DD statements allocate various load libraries that are required for executing SVOS and BMC Software products. All libraries that are allocated to STEPLIB must be authorized. SVOS is divided into four load modules: LODR410\$ (ETILOADR), LODR411\$, LODR412\$, and DISR410\$ (ETIDYNAM). The BMC Software product(s) also can be found in this library.</p>
PARMLIB DD	<p>defines an input sequential data set or a member of a partitioned data set</p> <p>After SVOS has completed initialization, this file is examined for loader commands. The file normally is used to issue START commands for the BMC Software products.</p>
SYSPRINT DD	<p>defines a sequential output data set that is used for listing actions that were taken during dynamic activation processing</p> <p>You can use a nonexistent SYSOUT class can be used to purge the SYSPRINT file after SVOS terminates. The DCB information is DCB=(LRECL=80,RECFM=FB).</p>
SYSLIB DD	<p>specifies a cataloged data set name for the LPALIB library concatenations that are to be allocated at SVOS startup as a default</p> <p>LPALIB data sets must be the same as they were when the system was last IPLd with a CLPA and/or an MPLA. There is a limit of three data sets that can be concatenated. A SYSLIB DD statement in the JCL overrides a SYSLIB global parameter.</p> <p>Note: You must specify either the SYSLIB global parameter or include the LPALIB in the execution JCL.</p>

Parameter Descriptions

During initialization, SVOS processes the parameter information that is passed with the PARM field on the EXEC statement. These parameters are used to pass installation-dependent options to the program. A list of the parameters and their defaults follows. All parameters are separated from their values with an equal sign (=).

FORCE Parameter

Function: The FORCE parameter determines whether the SVOS FORCE command can be issued.

Warning! If used improperly, the FORCE command can corrupt system control blocks and load modules.

If FORCE=NO is specified, the FORCE command is deactivated and cannot be used.

Default: FORCE=YES

Format: YES or NO

NOVERIFY Parameter

Function: The NOVERIFY parameter determines whether the SVOS START and STOP commands can use the NOVERIFY operand. The activation or deactivation procedure normally validates the pointers that are being changed in system control blocks and load modules. If NOVERIFY=YES is specified and the NOVER operand is requested on START/STOP, SVOS modifies storage without validating the current pointers.

Default: NOVERIFY=YES

Format: YES or NO

SUBCHAR Parameter

Function: The SUBCHAR parameter can be used in addition to MODIFY commands to enter SVOS commands from the console. The slash (/) is the default character, but this operand can be used to select any character as the subsystem character. If SUBCHAR=NONE is specified, only the MODIFY form of the loader commands are processed.

Default: SUBCHAR=/

Format: Any valid subsystem command character or NONE

Note: When you are using the multiple subsystem feature of SVOS, the subsystem character should be unique for each SVOS task that is started.

SUBSYS Parameter

Function: The SUBSYS parameter defines the subsystem name that identifies SVOS to the system. If the default subsystem name needs to be overridden, specify the new four-character subsystem name by using the SUBSYS parameter.

Default: SUBSYS=ETIS

Format: Any valid four character subsystem name.

CANCEL Parameter

Function: The CANCEL parameter allows SVOS to run as CANCELABLE or NON-CANCELABLE. The system may be exposed to error conditions if SVOS is eligible to be cancelled. OS/390 will not honor a cancel request by the operator if SVOS is running as NON-CANCELABLE. This parameter allows SVOS to be shut down without error and resolve all changes made to the system. If YES is coded during SVOS initialization, a warning message is sent to the operator to respond either YES (allow) or NO (prevent) further initialization. If NO is specified, no message is sent and SVOS continues initialization as normal and SVOS runs as NON-CANCELABLE.

Default: CANCEL=NO

Format: YES or NO

The SVOS Console Interface

In addition to controlling MAINVIEW SRM through the ISPF interface, you can control MAINVIEW SRM by operator commands that you enter through the SVOS console interface. Commands are directed to SVOS, which relays the commands to the MAINVIEW SRM component for execution.

Commands may be directed to SVOS in two ways:

- by using the component identifier that is defined to SVOS with the SUBCHAR parameter (see “Using a Subsystem Character” on page 4-10). For example:

```
/SVOS R,SYS=00
```

In this command, / is the subsystem character.

- by using the OS/390 MODIFY command. For example:

F SVOS,SVOS R,SYS=00

In this command, SVOS is the name of the MAINVIEW SRM Operator Services job.

The following types of SVOS console commands are available:

- START/STOP commands
- DISPLAY commands
- REFRESH commands
- SET commands
- VSCAN command

START/STOP Commands

Use standard OS/390 console commands to start and stop MAINVIEW SRM. MAINVIEW SRM runs under SVOS. MAINVIEW SRM is started and stopped with SVOS commands.

Warning! MAINVIEW SRM *should never be stopped with the OS/390 STOP command.*

DISPLAY Commands

Use DISPLAY commands to show the current values of MAINVIEW SRM parameters and members in the executing MAINVIEW SRM component.

The following abbreviations are valid for the DISPLAY console command:

- **D**
- **DISP**
- **DISPL**
- **DISPLAY**

Table 4-2 Command Syntax for the DISPLAY command

Command Syntax	Command Function
D,CAL,ALL	displays non-working days for all years in the calendar member
D,CAL= <i>year</i>	displays non-working days for a specific year
D,DIAG,ALL	displays all user-defined diagnostic members
D,DIAG= <i>diagnostic name</i>	displays a single user-defined diagnostic member
D,EVENT,ALL	displays information about all event IDs
D,EVENT= <i>eventid</i>	displays information about a specific event ID

Table 4-2 Command Syntax for the DISPLAY command

Command Syntax	Command Function
D,FUNC,ACT	displays information about all active functions
D,FUNC,ALL	displays information about all functions
D,FUNC= <i>function name</i>	displays information about a specific function
D,FUNC= <i>function name</i> ,FLST	displays information about a specific function's filter list
D,FUNC= <i>function name</i> ,RLST	displays information about a specific function's rule list
D,POOL,ALL	displays information about all pools
D,POOL= <i>pool name</i>	displays information about a specific pool
D,SPOL,ALL	displays information about all DFSMS-managed subpools
D,SPOL= <i>subpool</i>	displays information about a specific DFSMS-managed subpool
D,SYS	displays the current MAINVIEW SRM system definition For example: F SVOS,SVOS D,SYS
D,VAR,ALL	displays all user-defined variables
D,VAR= <i>variable name</i>	displays a single user-defined variable

REFRESH Commands

Use REFRESH commands to rebuild MAINVIEW SRM parameters from the members in the parameter data set.

The following abbreviations are valid for the REFRESH console command:

- R
- REF
- REFL
- REFRESH

Table 4-3 Command Syntax for the REFRESH command

Command Syntax	Command Function
R,CAL= <i>xx</i>	causes the SMCALS <i>xx</i> member to be read and all calendar parameters to be rebuilt
R,DIAG= <i>xx</i>	causes the SMDIAG <i>xx</i> member to be read and all diagnostic parameters to be rebuilt
R,EVNT= <i>xx</i>	causes the SMEVNT <i>xx</i> member to be read and all event parameters to be rebuilt
R,FUNC= <i>xx</i>	causes the SMFUNC <i>xx</i> member to be read and all function parameters to be rebuilt, including SMFLST <i>xx</i> and SMRLST <i>xx</i>
R,FUNC= <i>function name</i> ,FLST <i>xx</i>	causes the specified filter list (SMFLST <i>xx</i>) for the named function to be rebuilt from the parameter library member

Table 4-3 Command Syntax for the REFRESH command

Command Syntax	Command Function
R,FUNC= <i>function name</i> ,RLST <i>xx</i>	causes the specified rule list (SMRLST <i>xx</i>) for the named function to be rebuilt from the parameter library member
R,POOL= <i>xx</i>	causes the SMPPOOL <i>xx</i> member to be read and all pools parameters to be rebuilt
R,SPOL= <i>xx</i>	causes the SMSPOL <i>xx</i> member to be read and all SMS sub-pool parameters to be rebuilt
R,SYS= <i>xx</i>	causes the complete MAINVIEW SRM component to be rebuilt and all parameters refreshed, based on the SMMSYS <i>xx</i> member that is identified by specified suffix
R,VAR= <i>xx</i>	causes the SMVAR <i>xx</i> member to be read and all variables to be rebuilt

SET Commands

Use SET commands to *temporarily* alter the values of certain MAINVIEW SRM parameters.

Warning! Changes you make with the SET command are temporary. The changes are reset to the defined values in the MAINVIEW SRM PARMLIB members after MAINVIEW SRM is stopped and restarted.

The following abbreviations are valid for the SET console command:

- **T**
- **SET**

Table 4-4 Command Syntax for the SET command

Command Syntax	Command Function
T,FUNC= <i>function name</i> ,ACTIVE=YES/NO	temporarily changes the status of the specified function
T,FUNC= <i>function name</i> ,MSG=I/W/E/S/N	temporarily changes the MSG parameter of the specified function to information (I), warning (W), error (E), severe (S), or none (N)
T,FUNC= <i>function name</i> ,SMF=E/I/N	temporarily changes the SMF message parameter for the specified function to information (I), warning (W), error (E), severe (S), or none (N)
T,MSGID,YES/NO/DEB	temporarily changes the message ID to allow or suppress the message ID from being issued with MAINVIEW SRM messages (The debug option is used at the direction of BMC Software MAINVIEW SRM Customer Support to add diagnostic information to messages that are written to the job log.)
T,MSGPREF, <i>xxx</i>	temporarily changes the message prefix characters
T,SMFID, <i>nnn</i>	temporarily changes the SMFID The new value should be in the range from 129 to 255, or 0 (zero) to disable recording. For example, F SVOS,SVALLOC T SMFID,255
T,TRKCYL, <i>nnnnn</i>	temporarily changes the default cylinder size (tracks per cylinder)
T,TRKLEN, <i>nnnnn</i>	temporarily changes the default track size (bytes per track)

VSCAN command

Use the VSCAN command to start a VTOC scan. For example:

```
/SVOS VSCAN,SUF=xx
```

In this command, *xx* is the suffix of the SMVSCF*xx* member to use in the scan.

The data set name that contains the scan output is indicated in a message appearing in the SVOS job log in response to this command. The output data set is also available in the WBVTOC view.

Chapter 5 Using MAINVIEW SRM Easy Menus

This chapter contains an overview of each menu option and tells where to find more information.

Overview	5-2
SRM Real Time Monitor	5-4
Pools	5-4
SMS Storage Groups	5-5
SMS Pools	5-6
RAID Configurations	5-8
Storage Performance	5-9
SRM Historical Data	5-11
Historical Space	5-11
Historical Performance	5-13
EasyHSM	5-15
SG-Control	5-16
SMF Report Library	5-17
SRM Administration	5-18
Parmilb Members	5-18
Functions	5-21
SRM Component Status	5-22
SRM Tools and Menus	5-24
Workbench	5-24
MVSRM View List	5-26
MVSRM Batch Reports	5-27
MAINVIEW Messages	5-28

Overview

MAINVIEW SRM offers a quick, convenient way to use the product with little introduction and without having to remember view names. This interface consists of views with one primary menu, the EZSRM Menu.

The EZSRM Menu, shown in Figure 5-1, is presented as the initial display when you access MAINVIEW SRM. All options on this menu hyperlink to high-level views or submenus and have been given succinct, descriptive names, allowing you to quickly access the data you need.

Figure 5-1 EZMSRM Menu

```

14MAY2001 09:16:35 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> PAGE
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =EZSRM=====SJSJG=====*=====14MAY2001==09:16:35===MVSRM====D====1
          EZSRM  Menu

SRM Real Time Monitor  +-----+ SRM Historical Data
                        |         |
. Pools                 | Place cursor on | > Historical Space
. SMS Storage Groups   | menu item and  | > Historical Performance
. SMS Pools            | press ENTER   | > EasyHSM
> RAID Configurations +-----+ > SGControl Applications
> Storage Performance > SMF Report Library

SRM Administration    SRM Tools and Menus

> Parmlib Members     > Workbench
. Functions            . MVSRM View List
. SRM Component Status . MVSRM Batch Reports
                       . MainView Messages
                       . Return....
    
```

Options that are preceded by a period (.) display system data; options that are preceded by a *greater than* sign (>) display either pop-up menus in the center of the window or other menu views from which you can access a broad variety of information. Options on this menu are grouped into the following categories:

Category	Description
SRM Real Time Monitor	displays views and submenus that give you real time storage information
SRM Historical Data	displays submenus from which you can access historical space, performance, DFHSM, applications, and SMF libraries
SRM Administration	displays views and submenus that allow you to define the MAINVIEW SRM components
SRM Tools and Menus	provides access to various useful listings, such as HLQs, VTOCs, view names, batch reports, and messages

The EZSRM Menu options are described in the following pages.

Tip: You can use the EZcmd menu to hyperlink to another view rather than the action line command. See “EZcmd Menus” on page 1-19 for details.

SRM Real Time Monitor

The Real Time Monitor options display views and submenus that give you real time storage information.

Pools

The Pools option displays the MPOOL view, as shown in Figure 5-2, which lists all the pools that are defined in the system.

Figure 5-2 MPOOL View

```

12FEB2001 10:33:23 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =MPOOL=====SJSG=====*=====12FEB2001==10:33:20===MVSRM===D===99
C Pool      Type      SG      SSI
- - - - - Collect Target
AABAB3     DASD      Y       SJSG
ALLVOLS    DASD      N       SJSG
ANDY       DASD      N       SJSG
BELL       DASD      N       SJSG
BJPLX1     DASD      N       SJSG
BJPLX2     DASD      N       SJSG
BJPLX3     DASD      N       SJSG
BJPLX4     DASD      N       SJSG
BJPLX5     DASD      N       SJSG
BJPLX6     DASD      N       SJSG
BJPLX7     DASD      N       SJSG
BJPL01     DASD      N       SJSG
BJPL02     DASD      N       SJSG
BJPL03     DASD      N       SJSG
BJPL04     DASD      N       SJSG
BJPL05     DASD      N       SJSG

```

The following actions are available from the MPOOL view:

Action	Takes you to	View Name
D	contents of storage that correspond to the definition of the selected pool	
P	performance data (over the last 5 minute interval, maintained by the DASD performance option (DPO)) for the pool	MVOLPER
S	space information for volumes in the selected pool	MVOLSPC
V	a list of all volumes and unit addresses that belong to the selected pool	MPOOLVOL

The following actions are available on the MVOLPER view:

Action	Takes you to	View Name
P	performance statistics that are gathered by DPO for the base and all alias UCBs of a parallel access volume (PAV)	MVOLPAV

See “Chapter 12, Monitoring Pools and SMS Storage Groups” for more information.

SMS Storage Groups

The SMS Storage Groups option displays the MSMSG view, as shown in Figure 5-3, which lists all SMS groups that are defined in the system.

Figure 5-3 MSMSG View

```

09FEB2001 15:46:55 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1           ALT WIN ==>
W1 =MSMSG=====SJSG=====*=====09FEB2001==15:46:55====MVS RM====D====
C Storage                SSI
- Group                  Target
MIKEL                    SJSG
SGENG                    SJSG
STEVEJ                   SJSG
STG1                     SJSG
STG2                     SJSG
TEDG                     SJSG

```

The following actions are available from the MSMSG view:

Action	Takes you to	View Name
P	performance data for the SMS group (this data is shown over the last 5-minute interval and is maintained by DPO)	MVOLPER
S	space information for volumes in the selected SMS group	MVOLSSPC
V	a list all volumes and unit addresses that belong to the selected SMS group	MPOOLVOL

The following actions are available on the MVOLPER view:

Action	Takes you to	View Name
P	performance statistics gathered by DPO for the base and all alias UCBs of a parallel access volume (PAV)	MVOLPAV

SMS Pools

The SMS Pools option displays the MSMSG view, as shown in Figure 5-4, which lists all the SMS pools defined in the system.

Figure 5-4 MSMSG View

```

09FEB2001 15:42:58 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =MSMSG=====SMSG=====*=====09FEB2001==15:42:58====MMSG====D====3
C Pool      SSI
- - - - - Target
  CHERRY1  SMSG
  RAYSMS1  SMSG
  RAYSMS2  SMSG

```

The following actions are available from the MSMSP view:

Action	Takes you to	View Name
D	contents of storage that correspond to the definition of the selected SMS pool	
P	performance data for the SMS group (this data is shown over the last 5-minute INTERVAL and is maintained by DPO)	MVOLPER
S	space information for volumes in the selected SMS group	MVOLSPC
V	a list of all volumes and unit addresses that belong to the selected SMS group	MPOOLVOL

The following actions are available on the MVOLPER view:

Action	Takes you to	Hyperlinks
P	performance statistics gathered by DPO for the base and all alias UCBs of a parallel access volume (PAV)	MVOLPAV

See the *MAINVIEW SRM EasyPOOL User Guide and Reference* for more information.

RAID Configurations

The RAID Configurations option displays the RAID Devices pop-up menu in the center of the window, as shown in Figure 5-5. This option requires the StorageGUARD component of MAINVIEW SRM.

Figure 5-5 EZSRM Menu, RAID Devices Menu

```

14MAY2001 09:18:13 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRM====EZSRMR====SJSJG====*=====14MAY2001==09:16:35====MVSRRM====D====1
          EZSRM Menu

SRM Real Time Monitor                      SRM Historical Data
+ RAID Devices =====+
. Pools                                  . EMC Symmetrix          . > Historical Space
. SMS Storage Groups                    . IBM RVA/Shark/RAMAC  . > Historical Performance
. SMS Pools                              . Return...            . > EasyHSM
> RAID Configurations                  +-----+ > SGControl Applications
> Storage Performance                    > SMF Report Library

SRM Administration                        SRM Tools and Menus
> Parmlib Members                       > Workbench
. Functions                               . MVSRRM View List
. SRM Component Status                   . MVSRRM Batch Reports
                                          . MainView Messages
                                          . Return....

```

Options on this menu are divided into the following categories:

Option	Description	View Name
EMC Symmetrix	displays all EMC Symmetrix devices that are accessed by the current MVS system and provides hyperlinks to configuration information, directors, physical disks, SRDF, subsystem, and volume information	REBOX
IBM RVA/Shark/RAMAC	displays all RAMAC devices, 2105 subsystem devices, and RVA devices that are attached to the current MVS system This view provides hyperlinks to drawer, subsystem, and logical volume information.	RIBOX

See the *MAINVIEW SRM StorageGUARD User Guide and Reference* for further view information.

Storage Performance

The Storage Performance option displays the Storage Performance pop-up menu in the center of the window, as shown in Figure 5-6. StorageGUARD provides a unique perspective on DASD-related performance information. This option requires the StorageGUARD component of MAINVIEW SRM.

Figure 5-6 EZSRM Menu, Storage Performance Menu

```

14MAY2001 09:18:38 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRM====EZSRMS===SJSG====*=====14MAY2001==09:16:35====MVSRM====D====1
          EZSRM  Menu

SRM Real Time Monitor          SRM Historical Data
+ Storage Performance =+
. Pools                        . Device Activity      . > Historical Space
. SMS Storage Groups          . Channel Activity     . > Historical Performance
. SMS Pools                   . I/O Queueing        . > EasyHSM
> RAID Configurations         . ENQ/Reserve Activity. > SGControl Applications
> Storage Performance         . ENQ Activity         . > SMF Report Library
                              . Return...           .
SRM Administration           +-----+ SRM Tools and Menus
                              |
> Parmlib Members            > Workbench
. Functions                   . MVSRM View List
. SRM Component Status        . MVSRM Batch Reports
                              . MainView Messages
                              . Return....

```

This menu includes the following options:

Option	Description	View Names
Device Activity	provides a listing of device activity and performance on a volume-by-volume basis View information includes volume status, SMS status, mount status, and paging indicators.	MDEV MDEV D MDEV SUM MDEV SP MDEV SD
Channel Activity	provides a listing of channel activity and performance for all online channels The view values are generated using the delta between pressing Enter.	MCHAN MCHAN D MCHAN SUM
I/O Queueing	provides a listing of all online LCUs and associated channels that have had activity during the current CMF/RMF recording interval This allows you to determine if there are performance problems because of bottlenecks at the channel, controller, or LCU level.	MIOQ MIOQ D MIOQ S
ENQ/Reserve Activity	displays all currently outstanding RESERVE requests that have been made against the serially reusable resources in your system This view can be used to show resource control contention.	MRES MRES D
ENQ Activity	displays information on the contention that exists for all serially reusable resources in your system This view can be used to show resource control contention.	MENQ MENQ D

See the *MAINVIEW SRM StorageGUARD User Guide and Reference* for further view information.

SRM Historical Data

The historical data options display submenus from which you can access historical space, performance, DFHSM, and applications.

Historical Space

The Historical Space option displays the EZSRMSGD Menu, as shown in Figure 5-7. Historical space views provide space usage information by time, pool, RAID device, application account, and volume. This option requires the StorageGUARD component of MAINVIEW SRM.

Figure 5-7 EZSRMSGD Menu

```

14MAY2001 09:20:36 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> PAGE
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =EZSRMSGD=====SJSG=====14MAY2001==09:20:36===MVSMSGD=D===1
          EZSRMSGD Menu

  Historical Space      +-----+
                        |         |
> Summary by Time     | Place cursor on |
. Pool Utilization    | menu item and  |
. RAID Physicals      | press ENTER   |
. SGControl Applications
. Volumes

  SRM Administration          SRM Tools and Menus

> Parmlib Members          > Workbench
. Functions                 . MVSMSGD View List
. SRM Component Status     . MVSMSGD Batch Reports
                           . Return....

```

This menu includes the following options:

Option	Description	View Name
Summary by Time	provides a list of interval snapshots and links to summarized and other views for DASD devices	SPSNAP SPSNAPD SPSNAPW
Pool Utilization	provides pool-level space usage information	SPPOOL
RAID Physicals	provides RAID device usage information	SPRAID SPRAIDVO
SG-Control Applications	provides application-level space usage information	SPAPPL
Volumes	provides volume-level space usage information	SPVOL

See the *MAINVIEW SRM StorageGUARD User Guide and Reference* for more information.

Historical Performance

The Historical Performance option displays the EZSRMSGP Menu, as shown in Figure 5-8. This option requires the StorageGUARD component of MAINVIEW SRM.

Figure 5-8 EZSRMSGP Menu

```

14MAY2001 09:22:26 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRMSGP=====SJSH=====14MAY2001==09:22:26====MVSMSGP=D====
          EZSRMSGP  Menu

Historical Performance +-----+ SRM Administration
                       | Place cursor on |
. System Summary by Time | menu item and | > Parmlib Members
. Interval Data         | press ENTER  | . Functions
. Channel Path          +-----+ . SRM Component Status
. Cache Control Unit    |                     | . MVSMSGP View List
. Logical Control Unit  |                     | . MVSMSGP Batch Reports
. Pools                 | RAID Performance | . Return....
. Volumes               |                     |
. Storage Class         | . RAID Director  |
. Data Set              | . RAID Physical Volume
. Job                   | . RVA Subsystem Frame
                       | . 2105 Ranks    |

```

This menu includes the following options:

Option	Description	View Names
System Summary by Time	provides summarized performance history information by time for historical data	PRSSUM
Interval Data	displays the interval reports that are stored in the database	PRINTV
Channel Path	displays channel path data from the most recent interval	PRCHP
Cache Controller	displays cache controller data from the most recent interval	PRCCU
Logical Control Unit	displays logical control data from the most recent interval	PRLCU
Pool	displays pool data from the most recent interval	PRPOOL
Volume	displays volume data from the most recent interval	PRVOL
Storage Class	displays storage class data from the most recent interval	PRSCL
Data Set	displays data set data from the most recent interval	PRDS
Job	displays job data from the most recent interval	PRJOB
RAID director	displays RAID director data from the most recent interval	PRRDIR
RAID Physical Volume	displays RAID physical volume data from the most recent interval	PRPVOL
RVA Subsystem Frame Resource	displays RVA data from the most recent interval	PRRSF
RAID Rank	displays RAID Rank performance data from the most recent interval	PRRRK

See the *MAINVIEW SRM StorageGUARD User Guide and Reference* for further view information.

SG-Control

The SG-Control Applications option displays the EZSRMSGC menu, as shown in Figure 5-10. This component requires the StorageGUARD and SG-Control components of MAINVIEW SRM.

Figure 5-10 EZSRMSGC Menu

```

14MAY2001 09:24:29 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> PAGE
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =EZSRMSGC=====SJSH=====*=====14MAY2001==09:24:29===MVSMSGC=D====1
                                EZSRMSGC  Menu

SG-Control          +-----+          SRM Administration
                    |         |          |
. Application Maint | Place cursor on | > Parmlib Members
. Application Mass Upd | menu item and | . Functions
. Application List   | press ENTER   | . SRM Component Status
. Data Set Information +-----+ . MVSMSGC View List
                                   . MVSMSGC Batch Reports
                                   . Return....

```

This menu includes the following options:

Option	Description	View Name
Application Maint	allows you to add, change, or mark for deletion <i>individual</i> applications	
Application Mass Update	allows you to change <i>multiple</i> applications	
Application List	lists all applications	APPLTAB
Data Set Info	allows you to query applications by data set name	APPLDSNT

See the *MAINVIEW SRM SG-Control User Guide and Reference* for more information.

SMF Report Library

The SMF report library option is used to specify an SMF output data set. In addition to displaying messages on the job log, all MAINVIEW SRM activity messages can be written to the SMF data set. The messages written to SMF contain the same information as those written to the job log and can be accumulated for any length of time. Messages are generated by function.

The SMS report library option requests the SMF report library data set name and then displays the SMFRPTD, as shown in Figure 5-11.

Figure 5-11 SMFRPTD View

```

22MAR2001 14:16:30 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =SMFRPT=====SJSJG=====*=====22MAR2001==14:16:06====MVSIRM====D===23
CMD Member      Description                               Data Set Name
-----
$SSI0000        MAINVIEW SRM SUBSYSTEM PROCESSING                    DOTSON.SMFLIB
DASDPOOL        ALLOCATE DATA SETS TO DASD VOLUMES                  DOTSON.SMFLIB
FORCECAT        FORCE CATALOG DISPOSITION ON NEW DSNS                 DOTSON.SMFLIB
HSMBACKP        CONTROL DATASET SELECTION FOR HSM BACKUP             DOTSON.SMFLIB
HSMCCNV         APPLY CALENDAR CONVERSION TO MIGRATION              DOTSON.SMFLIB
HSMIGRT         CONTROL DFHSM MIGRATION CHARACTERISTICS              DOTSON.SMFLIB
NOCATLG2        PREVENT OCCURRENCES OF NOT CATLG2                   DOTSON.SMFLIB
OPENEMPT        OPEN EMPTY DATA SET TO SET END-OF-FILE             DOTSON.SMFLIB
OPTBLKSZ        ALLOCATE DATASETS WITH OPTIMUM BLOCKSIZE            DOTSON.SMFLIB
SETEXPDT        SET EXPIRATION DATE FOR NEW DATASETS                 DOTSON.SMFLIB
SMSACSDC        ASSIGN A DFSMS DATA CLASS                           DOTSON.SMFLIB
SMSACSMC        ASSIGN A DFSMS MANAGEMENT CLASS                     DOTSON.SMFLIB
SMSACSSC        ASSIGN A DFSMS STORAGE CLASS                         DOTSON.SMFLIB
SMSACSSG        ASSIGN A DFSMS STORAGE GROUP                         DOTSON.SMFLIB
SMSSELCT        SELECT VOLUME FROM STORAGE GROUP                     DOTSON.SMFLIB
SPACPRIM        REDUCE PRIMARY SPACE ALLOCATION SIZE                  DOTSON.SMFLIB

```

The following actions are available from this view:

Action	Used to
B	invoke ISPF browse for the specified member
E	invoke ISPF edit for the specified member

SRM Administration

The selections under SRM Administration provide views and submenus that allow you to define the MAINVIEW SRM components

Parmlib Members

The Parmlib Members option, which allows you to define system parameters, displays the EZSRMP Parmlib Members menu, as shown in Figure 5-12.

Figure 5-12 EZSRM Menu, Parmlib Members Menu

```

14MAY2001 09:25:43 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> PAGE
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRM====EZSRMP===SJSJ=====14MAY2001==09:25:40===MVSRM====D====1
          EZSRMP Parmlib Members

SRM Parmlib Members      +-----+      SRM Administration
                          |         |
                          | Place cursor on |
                          | menu item and   |
                          | press ENTER    |
. All Active Members     |         | . Functions
. All Members           |         | . SRM Component Status
. System                +-----+ . Return....
. Functions
. Pools
. Events
. SMS Pools             . Filter List
. Diagnostics           . Rule List
. VtocScan              . Edit Reorg (StopX37)
. Variables             . CritList
. Calendars             . SG-Auto

```

This menu includes the following options:

Table 5-1 Parmlib Member Administration

Option	Member	Description
All Active Members	N/A	invokes the ADMEM * Y command and displays all <i>active</i> members
All Members	N/A	invokes the ADMEM * * command and displays <i>all</i> members

Table 5-1 Parmlib Member Administration

Option	Member	Description
System	SMMSYSxx	contains master specifications that apply to the overall functionality of MAINVIEW SRM SMMSYSxx specifies default device type information, message control information, passwords, and SMF IDs. At startup, the suffix of SMMSYSxx is read first. SMMSYSxx specifies the other definition members are to be included in the MAINVIEW SRM configuration to be started. See “Chapter 2, Defining the MAINVIEW SRM System” for more information about parmliib members.
Functions	SMFUNCxx	activates MAINVIEW SRM functions and controls message and tracing activity Functions provide all the runtime services of MAINVIEW SRM. SMFUNCxx points to members SMFLSTxx and SMRLSTxx, which select resources and control the operation of the functions. With the exception of the USERVARS function, functions and the filter lists and rule lists that are used by a function are component specific. See “Chapter 6, Defining and Activating Functions.”
Pools	SMPOOLxx	defines pools of DASD volumes and pools of tape devices See “Chapter 7, Defining a Pool.”
Events	SMEVNTxx	define how storage event notices are managed The Enterprise Storage Automation event member parameters include mode, destination, severity, and so on. For input instructions, see the <i>Enterprise Storage Automation User Guide</i> .
SMS Pools	SMSPOLxx	defines subpools of SMS-managed DASD volumes SMS subpools are used by EasyPOOL only. For input instructions, see the <i>EasyPOOL User Guide and Reference</i> .
Diagnostics	SMDIAGxx	specifies various diagnostic options, such as tracing of filter and rule list processing or other debugging options, that may be requested by BMC Software Customer Support See “Chapter 10, Using Diagnostics.”
VTOC Scan	SMVSCF	contains information that is used in processing requests to collect VTOC scan data
Variables	SMVARSxx	defines MAINVIEW SRM variables and assigns values to those variables, which can be substituted for MAINVIEW SRM parameter values See “Chapter 8, Using Variables.”
Calendars	SMCALsxx	defines non-working days (weekends, public holidays, company-specific holidays, and so on) to assist in DFHSM migration See “Chapter 9, Defining a Calendar.”
Filter List	SMFLSTxx	selects the resources on which a MAINVIEW SRM function operates Filter list parameters select resources by data set name, volume name, unit name, data set type, job name, account number, or a number of other characteristics. “Chapter 11, Using Filter and Rule List Parameters.”

Table 5-1 Parmlib Member Administration

Option	Member	Description
Rule List	SMRLSTxx	<p>defines rules that control the action of the MAINVIEW SRM function on the resources selected</p> <p>Rule list parameters allow modification of such properties as blocksize, pool name, data class, and allow further identification of affected resources based on data set name, volume name, job name, data set size, and so on.</p> <p>See "Chapter 11, Using Filter and Rule List Parameters."</p>
Edit Reorg (StopX37)	SMRORGxx	<p>defines rules that control data set reorganization</p> <p>For input instructions, see the <i>StopX37/II User Guide and Reference</i>.</p>
CritList	SMCRITxx	<p>allows you to identify in your environment a list of critical data sets that should never be placed on the same volume as any other data set in the list</p> <p>It is important to consider, however, that CRITDSN is resource intensive and should be used only for a small list of critical data sets. It should not be used without considering the system impact.</p> <p>For input instructions, see the <i>EasyPOOL User Guide and Reference</i>.</p>
SG-Auto	SMACMDxx	<p>defines SG-Auto command members or a combination of a command and matrix members to perform the functions of monitoring system usage and automating procedures for corrective action</p> <p>Matrix members are called by command members.</p> <p>For input instructions, see the <i>SG-Auto User Guide and Reference</i>.</p>

Functions

The Functions option displays the ADFUNC view, as shown in Figure 5-12, which lists the status and information for all functions defined in the system.

Figure 5-13 ADFUNC View

```

26MAR2001 13:51:46 ----- INFORMATION DISPLAY -----
COMMAND ====>                                SCROLL ====> PAGE
CURR WIN ====> 1          ALT WIN ====>
>W1 =ADFUNC=====SJSG=====*=====26MAR2001==13:51:46====MVSRM====D===41
CM Function Status FLST RLST MSG SMF Description                                     L
-----
DASDPOOL Y      99  99  I  W  CONTROL DASD ALLOCATIONS                                     Y
DMS2HSM  Y      CV           I  I  DMS TO HSM CONVERSION                                       Y
DSNCHECK N      51  R3  I  I  ENFORCE NAMING STANDARDS                                     Y
FDRASIST Y      5D  5D  I  I  FDR ASSIST                                                 Y
FORCECAT Y      52           I  I  FORCE CATALOGING OF DATASETS                               Y
HSMBACKP Y      55  01  I  I  DFHSM BACKUP ENHANCEMENT                                   Y
HSMDELET Y      ML           I  I  POOL SUMMARY RECORD PROCESSING                             Y
HSMCCNV  Y      55  I7  I  I  DFHSM MIGRATION DAYS BY CALENDAR                           Y
HSMIGRT  Y      55  ML  I  I  DFHSM MIGRATION ENHANCEMENT                               Y
HSMRECAL Y      55  E6  I  I  DFHSM RECALL ENHANCEMENT                                   Y
MODDELET Y      56           I  I  DISP=(MOD,DELETE,DELETE)                                  Y
NOCATLG2 Y      IV  I8  I  I  HANDLE DUPLICATE DATASET NAMES                             Y
SGD      Y      BJ  BJ  I  I  SGD ADABAS ACCOUNT                                         Y
SGDACCT  Y      S1           I  I  SGD ACCOUNT UTILIZATION PROCESSING                         Y
SGDPOOL  Y      S2           I  I  SGD POOL UTILIZATION PROCESSING                             Y
SGDVOL  Y      S3           I  I  SGD VOLUME UTILIZATION PROCESSING                           Y

```

The following actions are available from this view:

Action	Used to
A	activate the function
I	inactivate the function
C	change the filter/rule list suffix and/or the message or SMF level
RF	refresh the filter list
RR	refresh the rule list
R	refresh both the filter and rule list
BF	browse the filter list
BR	browse the rule list
EF	edit the filter list
ER	edit the rule list

Action	Used to
DF	display the filter list in core storage
DR	display the rule list in core storage

SRM Component Status

The SRM Component Status view displays a list of all MAINVIEW SRM components on your system, along with the active/not active status and product license information for each component.

All components except SVOS may be activated and deactivated from this view:

- Use the A or S line commands to activate components
- Use the D or P line commands to deactivate components

Multiple components may exist in the same startable component. The Started Component field lists the actual component that is activated or deactivated for the listed component. When a component is activated, all components that have the same Started Component value are activated. The same is true for deactivating a component.

The term ACTIVE in the status field does not necessarily mean that the component is performing its inherent function. Some components require that a function or functions to be activated before they begin to perform their function. The user guide for each component should be reviewed to determine whether a component is performing its processing.

The view can be invoked from the EZSRM menu or by invoking the view name, COMPSTAT, as shown in Figure 5-14. To filter the view to active components only, invoke COMPSTAT * Y. To filter the view to licensed components only, invoke COMPSTATLIC.

Figure 5-14 SRM Component Status

```

21FEB2001 06:16:38 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =COMPSTAT=====SJSG=====*=====21FEB2001==06:16:38====MVS RM====D====10
C Component      Component      Component  SSI      Started  Lic
- Name           Description   Status     System   Component Ind
Automation       Storage automation  NOT ACTIVE SJSG     SVESA    Y
EasyHSM          HSM Collection    ACTIVE     SJSG     SVHSM    Y
EasyPOOL         Pool Allocation   ACTIVE     SJSG     SVALLOC  Y
EasySMS          EasySMS           ACTIVE     SJSG     SVALLOC  Y
StopX37/II       StopX37/II        ACTIVE     SJSG     SVALLOC  Y
SG-Auto          SGD Automation    NOT ACTIVE SJSG     SVSGA    Y
SG-Control       Historical application space  ACTIVE     SJSG     SVSGC    Y
SGD HIST         SGD historical space  ACTIVE     SJSG     SVSGD    Y
SGP HIST         SGD historical performance  ACTIVE     SJSG     SVSGP    Y
SVOS             SVOS Operator Services  ACTIVE     SJSG     SVOS     Y

```

Tip: Since the SVOS component must be licensed and started to be connected to MAINVIEW, it will always appear as LICENSED and ACTIVE. The LICENSED COMPONENT field contains the value COMMON, indicating that this product is licensed with any other component. Stopping SVOS causes all other components in the selected SVOS server to be stopped. The SVOS component itself then stops, which causes disconnection from MAINVIEW. After this occurs, the MVS RM product is no longer accessible and appears as INACTIVE on the PLEX view of the PlexManager.

SRM Tools and Menus

The SRM Tools and Menus options provide access to various useful listings, such as HLQs, VTOCs, and view names.

Workbench

The Workbench option displays the Workbench pop-up menu in the center of the window, as shown in Figure 5-15.

The Workbench option provides several real time data set-level reporting utilities to assist you with daily housekeeping of your DASD environment. With these reports and utilities, you can locate problem data sets and take action if necessary. You can inspect data sets from the catalog and VTOC viewpoints. This option requires the StorageGUARD component of MAINVIEW SRM.

Figure 5-15 Workbench Menu

```

14MAY2001 09:26:44 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =EZSRM===EZSRMW===SJSJG=====14MAY2001==09:26:40===MVSRM===D===1
                                EZSRM  Menu

SRM Real Time Monitor                SRM Historical Data
+ Workbench =====+
. Pools                             . HLQ                               . > Historical Space
. SMS Storage Groups                 . Catalog Super Locate.          . > Historical Performance
. SMS Pools                           . VTOC Reporting                 . > EasyHSM
> RAID Configurations                . Return...                       . > SGControl Applications
> Storage Performance                +-----+                          . > SMF Report Library

SRM Administration                    SRM Tools and Menus
> Parmlib Members                    . > Workbench
. Functions                           . MVSRM View List
. SRM Component Status                . MVSRM Batch Reports
                                       . MainView Messages
                                       . Return....

```

Options on this menu are divided into the following categories:

Option	Description	View Names
HLQ	lists all the high-level qualifiers in the catalog The master catalog is read and a list of all high-level qualifiers is compiled. This option can be used as a utility to search for entries in a top-down manner.	WBHLQ
Catalog Super Locate	provides an enhanced version of ISPF 3.4 and ISMF Data Set Services, comprising the best of both worlds with increased ease of use and flexibility	WBSL
VTOC Reporting	displays information on the DASD volume, such as volume serial number, mount type, percentage full, number of free DSCBs, free space, and so on	WVTOC

See the *MAINVIEW SRM StorageGUARD User Guide and Reference* for more information.

MVSRM View List

The MVSRM View List option displays a list of all views that are available for MAINVIEW SRM, as shown in Figure 5-16. View lists for specific MAINVIEW SRM products are available from the product menu.

Figure 5-16 MVSRM View List

```

28DEC2000 16:44:41 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1           ALT WIN ==>
W1 =MAIN=====SJS=====*=====28DEC2000==16:44:40====MVSRM=====71
C View Name Product Area Description
-----
ADFUNC MVSRM Admin PARMLIB FUNCTION DISPLAY
ADMEM MVSRM Admin PARMLIB MEMBER DISPLAY
APPLDTL MVSRM SGContro Temp testing CER ***
APPLTAB MVSRM SGContro Temp testing CER ***
COMPSTAT MVSRM Admin Component Config/Status (temp)
COMPSTRT MVSRM Admin Component Config/Status (temp)
EZSRM MVSRM EZMenu Main SRM Menu
MAIN MVSRM Main Main View
MCHAN MVSRM Monitor CHANNEL ACTIVITY
MCHAND MVSRM Monitor CHANNEL ACTIVITY
MDEV MVSRM Monitor DEVICE ACTIVITY
MDEV D MVSRM Monitor DEVICE ACTIVITY
MDEVSP MVSRM Monitor DEVICE ACTIVITY SPECIFIC
MDEVSPD MVSRM Monitor DEVICE ACTIVITY SPECIFIC
MENQ MVSRM Monitor ENQUEUE ACTIVITY
MENQD MVSRM Monitor ENQUEUE ACTIVITY

```

MVSRM Batch Reports

The MVSRM Batch Reports option allows you to print the information on MAINVIEW SRM common views by submitting a batch job. Batch reports for product-specific views can be selected from the product-specific menu. The MBSRM Batch Reports menu is shown in Figure 5-17.

Figure 5-17 MVSRM Batch Reports

```
----- MainView Batch Reports -----
Option ==>

0 Setup          Set up MainView Batch Environment
1 Generate       Generate MainView Batch Reports JCL
2 Edit/Submit    Edit/Submit existing MainView Batch Reports JCL
3 Browse         Browse MainView Reports

X Exit           Terminate

                (C) Copyright 2000, BMC Software, Inc.
```

See “Chapter 14, Generating and Managing Batch Reports” for details.

MAINVIEW Messages

The MAINVIEW Messages option displays a list of MAINVIEW messages, as shown in Figure 5-18.

Figure 5-18 **MAINVIEW Messages**

```

BMC Software ----- Messages & Co Row 7,614 to 7,627 of 8,864
COMMAND ==>>                                     SCROLL ==>> PAGE

Primary commands: S string - selects a message, L string - locates a message
Line commands: S - Select

LC Msg ID      Message Text
-----
__ SS9012E     UNABLE TO OPEN SYSUT4
__ SVM0002E     INVALID SUBSYSTEM NAME, SVOS TERMINATED
__ SVM0003S     CONFLICTING COMPONENT ALREADY ACTIVE, START ABORTED
__ SVM0004S     SUBSYSTEM ERROR, START ABORTED
__ SVM0005S     STORAGE OBTAIN FAILURE RC=@@@, START ABORTED
__ SVM0006S     ABEND INTIALIZING SUBSYSTEM, START ABORTED
__ SVM0007E     THERE WAS NO PASSWORD FOR ANY AVALLOC COMPONENT
__ SVM0008S     SVALLOC IS ALREADY ACTIVE, SAVLLOC TERMINATED
__ SVM0009E     INVALID SUBSYSTEM NAME, SVALLOC TERMINATED
__ SVM0010E     COMPONENT SVOS MUST BE STARTED
__ SVM0014E     module TERMINATED, ABEND CODE SYSTEM=abendcode
__ SVM0016E     COMMAND IS INVALID
__ SVM0019S     module NOT FOUND, SVOS TERMINATED
__ SVM0020E     ENF LISTEN REQUEST FAILED WITH RC (return code)

```

See the messages section in *Using MAINVIEW* for more information.

Chapter 6 Defining and Activating Functions

This chapter provides instructions for defining and activating functions in MAINVIEW SRM. The following information is included:

Overview	6-1
Using the SMFUNCxx Member	6-2
Active and Inactive Functions	6-2
Function Parameters	6-2
Function Parameter Quick Reference	6-3
Using the Functions Option	6-5
Defining the USERVARS Function	6-6

Overview

The information in this chapter pertains to the following components of MAINVIEW SRM.

- DMS2HSM
- EasyHSM
- EasyPOOL
- EasySMS
- SG-Control
- StorageGUARD
- StopX37/II

Functions must be defined and activated for one of these components to operate. This chapter explains how to define and activate functions.

Using the SMFUNCxx Member

SMFUNCxx is the function definition parameter member. It is read by MAINVIEW SRM during system startup. The suffix specification in SMMSYSxx indicates the version of SMFUNCxx that contains function specifications for the particular configuration of MAINVIEW SRM that is being executed. SMFUNCxx also identifies the filter list and rule list members used by a function. Functions provide all the runtime services of MAINVIEW SRM. SMFUNCxx is a required member, and MAINVIEW SRM does nothing without defined function parameters.

Active and Inactive Functions

A function may be defined in SMFUNCxx but be kept inactive. Functions in the distributed SMFUNCxx member are defined as inactive and must be activated (see “Using the Functions Option” on page 6-5). Function definitions that are not required may be removed from the SMFUNCxx member.

Note: Filter and rule lists for inactive functions are validated for proper syntax during refresh operations and at system start up.

Function Quick Reference

MAINVIEW SRM function member parameters are described in the “Appendix B, Global Parameters.” With the exception of the USERVARS function, functions and the filter lists and rule lists used by a function are specific to particular components. See “Defining the USERVARS Function” on page 6-6 for a complete description of the USERVARS function. See component reference manuals for complete descriptions of the other functions in the chart and the filter list and rule list parameters available for them.

Function Parameters

Functions are defined in SET statements. You can change parameters

- by editing the member directly
- by using the MAINVIEW SRM subsystem SET command
- through the **Functions** option on the EZSRMP Parmlib Members menu (see “Using the Functions Option” on page 6-5)

Function Parameter Quick Reference

SET statement function parameters are described in the following table. Detailed descriptions can be found in the “Appendix B, Global Parameters.”

Table 6-1 SET Statement Function Parameters

Parameter	Required	Description
NAME=xxxxxxx	Yes	MAINVIEW SRM-assigned name of the function
FLST=xx	No	suffix of filter list member SMFLSTxx
RLST=xx	No	suffix of rule list member SMRLSTxx
ACTIVE=YES/NO	Yes	status of the function
MSG=I/W/E/S/N	Yes	level of messages to be generated
SMF=I/W/E/S/N	Yes	level of messages to be written to SMF
TRACE=xxxxxxx	No	jobname of traced MAINVIEW SRM actions
DESC='xxxxxxxxxxxxxxxxxxx'	No	description of function

Examples

The following examples show how functions are defined in MAINVIEW SRM.

Example

The following statement defines the DSNCHECK function and makes it active, and defines the SPACLIMI function and makes it inactive.

SET	NAME=DSNCHECK FLST=56 ACTIVE=YES MSG=I SMF=N DESC='DSN CHECKING'	The MAINVIEW SRM DSNCHECK function is defined and made active. Filter list parameters are in SMFLST56; no rule list parameters are provided. Information and error messages are to be generated. No information or error messages are to go to SMF.
SET	NAME=SPACLIMI ACTIVE=NO FLST=GG RLST=G1 MSG=I SMF=E DESC='SPACE LIMIT' TRACE=TESTJOB	The MAINVIEW SRM SPACLIMI function (space limit) is defined but inactive. Filter list parameters are in SMFLSTGG; rule list parameters are in SMRLSTG1. Information and error messages are generated, but only error messages are sent to SMF. For jobs named TESTJOB, trace messages are generated to detail filter and rule list processing.

Example

The following statement shows the relationship between the SMFUNCxx and the SMFLSTxx members.

SMFUNCxx contains		
SET	NAME=SPACLIMI ACTIVE=YES FLST=56 RLST=56 MSG=I SMF=E DESC='SPACE LIMIT'	
SMFLST56 contains		
SET	MODE=SIM MSG=I INC SIZE>10M	Sets the function to simulation mode for data sets exceeding 10 megabytes.
SET	MODE=ACT MSG=E INC DISP1=NEW LLQ=TEST/	Sets the function active for data sets that have a first disposition parameter of NEW and a low-level data set name qualifier that begins with TEST.

Using the Functions Option

All functions in the distributed SMFUNCxx member are defined as inactive. To use MAINVIEW SRM, the functions must be activated and the resources on which the functions act must have been defined. MAINVIEW SRM functions may be activated all at once or may be activated a single function at a time. To use a function you must first perform the following actions:

- Define a filter list (in an SMFLSTxx member), with MODE=ACTIVE, and with INCLUDE parameters to apply the function to a set of resources (data sets, volumes, pools, jobs, and so on). (See “Using SMFLSTxx - Filter List Parameters” on page 11-2 for instructions on how to define a filter list.)
- Define a rule list (in an SMRLSTxx member), specifying how resources are affected. (See “Using SMRLSTxx - Rule List Parameters” on page 11-6 for instructions on how to define a rule list.)

After defining a filter list and a rule list, use the Functions option to define the parameters for each function and activate the function definition in SMFUNCxx using the following steps:

- Step 1** Click **Parmlib Members** on the EZSRM Menu. A list of members is displayed.
- Step 2** Select **Functions** from the selection list.
- Step 3** Type **E** in the line command field of the **CMD** column next to an active SMFUNCxx member; then press **Enter**.
- Step 4** Edit the member to specify the data center function definitions.
- Step 5** Press **PF3** to save and exit the member.
- Step 6** Type **R** in the line command of the **CMD** column next to the SMFUNCxx member you edited; then press **Enter**.

The member is refreshed. By refreshing SMFUNCxx, you have made it the active function list. It will remain active until SVOS is stopped or until you refresh another function member.

Defining the USERVARS Function

The USERVARS function allows you to set variables that are not included in the set of variables provided by MAINVIEW SRM. The variables that are defined in USERVARS may be subsequently tested and used in filter lists and rule lists for other functions.

SET statement parameters for the USERVARS function are located in member SMRLSTxx. Table 6-2 describes them.

Table 6-2 SET Statement Parameters for USERVARS Function

Parameter	Description
USRCn=xxxxxxxx	character field for a user-specified variable (1–8 characters) The value of <i>n</i> can be 1–10 (for example USRC1, USRC2, and so on).
USRNy=nnnnnnnn	numeric field for a user-specified variable The value of <i>y</i> can be 1–10 (for example USRN1, USRN2, and so on). The value of <i>n</i> cannot exceed 214783647.

Example

The following example shows how USERVARS can be used to set variables across functions.

SMFUNCxx member		
SET	NAME=USERVARS ACTIVE=YES MSG=I SMF=I FLST=UV RLST=UV DESC='SET USER VARIABLES'	Defines and activates the function.
SMFLSTUV member		
SET	MODE=ACT INC JOB=TESTUSRV	Selects the job to test the variables in active mode.
SMRLSTUV5 member		
SET	USRC1=USERVALU USRN1=100 INC DSN=TEST.USER1	Specifies two variables to be set and tested later.
SET	USRC1=USERVAL2 USRN1=200 INC DSN=TEST.USER2	Specifies two variables to be set and tested later.
SMFLSTDP member		
SET	MODE=ACT INC USERC1=USERVALU EXC USERN1>100	The variables established by the USERVARS function have been set and are used in the filter list for DASDPOOL.

Chapter 7 Defining a Pool

This chapter provides instructions for defining pools in MAINVIEW SRM. The following information is included:

Overview.....	7-1
Using the SMPPOOLxx Member.....	7-2
Parameters	7-2
Defining and Monitoring Pools	7-5

Overview

The information in this chapter pertains to the following components of MAINVIEW SRM:

- EasyPOOL
- StorageGUARD
- SG-Auto

Pool members define the composition of DASD pools. Pools must be defined before you begin using these components of MAINVIEW SRM.

Using the SMPOOLxx Member

SMPOOLxx is the pool definition parameter member. It is read by MAINVIEW SRM during system startup. The suffix specification in SMMSYSxx indicates the version of SMPOOLxx that contains pool specifications for the particular configuration of MAINVIEW SRM that is being executed. SMPOOLxx is a required member, even if pooling is not used.

Note: SMS subpools are only used by EasyPOOL. For SMS subpooling, use the SMSPOLxx member (see the *MAINVIEW SRM EasyPOOL User Guide and Reference*).

Parameters

Pools are defined on SET statements. INC/EXC statements that follow a SET statement identify the devices to be included in or excluded from a pool.

Note: Offline devices are not placed in pools, although they may be referenced on an INC statement. Offline devices that are assigned to a pool by an INC statement and are later varied online are automatically picked up by MAINVIEW SRM.

You can change SMPOOLxx parameters

- by editing the member directly
- by using the MAINVIEW SRM subsystem SET commands
- through the **Pools** option on the EZSRMP Parmlib Members menu (see “Defining and Monitoring Pools” on page 7-5)

Parameter Quick Reference

SET statement parameters for pools are described in the following quick reference table. Detailed descriptions can be found in the “Appendix B, Global Parameters.”

Table 7-1 SET Statement Pool Parameter Quick Reference

Parameter	Required	Description
POOLNAME=xxxxxxx	Yes	name to be assigned to a pool
USELIMIT=nnn	No	upper space threshold for new allocations
SGDCOLLECT=YES/NO	No	whether or not a pool is processed by StorageGUARD
TYPE=xxxx	No	device type

INC/EXC statement parameters for pool are described in the following quick reference table.

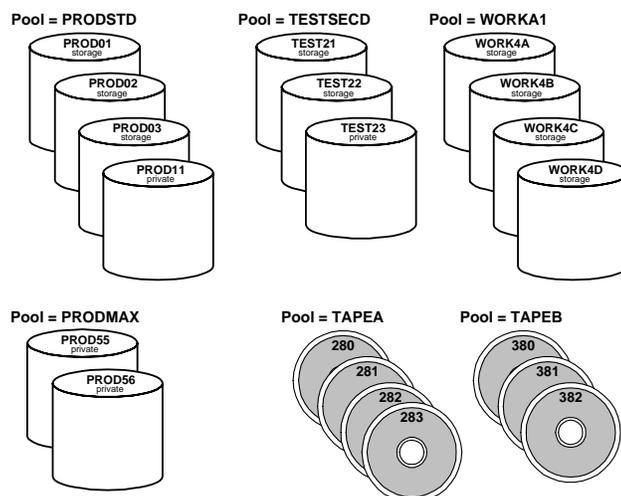
Table 7-2 INC/EXC Statement Pool Parameter Quick Reference

Parameter	Required	Description
ADR=xxxx	No	device address of tape unit in the pool
ADR=(xxxx,xxxx,...)	No	multiple tape device addresses (up to 15)
VOL=xxxxxx	No	volume serial number of the device in the pool
VOL=(xxxxxx,xxxxxx,...)	No	multiple volume serial numbers (up to 15)

Example

The following figure shows an example of a system storage configuration for pools. The statements following the figure define the configuration.

Figure 7-1 Example Storage Configuration for Pools



<pre>SET POOLNAME=PRODSTD USELIMIT=90 INC VOL=PROD0/ INC VOL=PROD11</pre>	<p>The PRODSTD pool includes all volumes with names that begin with PROD0 and volume PROD11. The upper space limit (USELIMIT) is 90%; 10% is generally reserved for expansion.</p>
<pre>SET POOLNAME=PRODMAX USELIMIT=80 INC VOL=(PROD55,PROD56)</pre>	<p>The PRODMAX pool includes volumes PROD55 and PROD56. The upper space limit is 80% (this pool is for very large data sets, which also need larger secondary extents; 20% is reserved for expansion.)</p>
<pre>SET POOLNAME=TESTSECD INC VOL=(TEST21,TEST22,TEST23)</pre>	<p>The TESTSECD pool includes volumes TEST21, TEST22, and TEST23. There is no provision for an upper space limit (no USELIMIT is set).</p>
<pre>SET POOLNAME=WORKA1 USELIMIT=95 INC VOL=WORK4/</pre>	<p>The WORKA1 pool includes all volumes with names that begin with WORK4. The upper space limit is 95%; 5% is generally reserved for expansion.</p>
<pre>SET POOLNAME=TAPEA TYPE=3490 INC ADR=(0280,0281,0282,0283)</pre>	<p>The tape pool TAPEA includes the 3490 tape units 280-283.</p>
<pre>SET POOLNAME=TAPEB TYPE=3480 INC ADR=038%</pre>	<p>The tape pool TAPEB includes 3480 tape units 380, 381, and 382.</p>

The following statement examples show how various pooling parameters may be used.

SET POOLNAME=&STORGRP SGDCOLLECT=YES INC VOL=/ 	Defines a pool for DFSMS storage groups to include only DFSMS volumes. Turns on the StorageGUARD storage space data collector option.
SET POOLNAME=WORK SGDCOLLECT=YES INC VOL=WRK/	Uses name masking to define a pool for work volumes to include all volumes with the WRK prefix. Turns on the StorageGUARD storage space data collector option.
SET POOLNAME=LARG SGDCOLLECT=YES INC VOL=(VOL001,VOL002,VOL005)	Defines a pool for large volumes to include specific volume serial numbers. Turns on the StorageGUARD storage space data collector option.
SET POOLNAME=&VOLSER(1,3) SGDCOLLECT=YES INC VOL=/ 	Defines a generic pool for unidentified volumes so that StorageGUARD can collect data on them for reporting purposes only.
SET POOLNAME=TTAPE INC UNIT=TAPE	Defines tape pools.
SET POOLNAME=TCART INC UNIT=CART	
SET POOLNAME=TTAPE2 INC ADR=(0480, 0481, 0482)	

Defining and Monitoring Pools

After you have identified the criteria for pools at your data center, use the following steps to define pools:

- Step 1** Click **Parmlib Members** on the EZSRM Menu. A list of members is displayed.
- Step 2** Select **Pools** from the selection list.

- Step 3** Type **E** in the line command field of the **CMD** column next to an active SMPOOL xx member, or place your cursor on a member name; then press **Enter**.
- Step 4** Edit the member to specify the data center pool definitions.
- Step 5** Press **PF3** to save and exit the member.
- Step 6** Type **R** in the line command field of the **CMD** column next to the SMPOOL xx member you edited; then press **Enter**.

The member is refreshed. By refreshing SMPOOL xx , you have made it the active pool list. It will remain active until SVOS is stopped or until you refresh another pool member.

For monitoring information about pools, see “Monitoring Pools and SMS Storage Groups” on page 12-1.

Chapter 8 Using Variables

This chapter provides instructions for defining and using variables in MAINVIEW SRM. The following information is included:

Overview.....	8-1
Using the SMVARSxx Member.....	8-1
Using the Variables Option.....	8-4

Overview

Variables provide the following benefits

- lengthy but widely used selection criteria are entered only once
- changes to widely used selection criteria are made in only one place
- parameters can be built by concatenating variables together

Using the SMVARSxx Member

SMVARSxx is the variables definition parameter member. It is read by MAINVIEW SRM during system startup. The suffix specification in SMMSYSxx indicates that version of SMVARSxx that contains variable specifications for the current execution of MAINVIEW SRM.

SMVARSxx variables allow a user-specified name to be assigned any number of selection parameters. The variable name is then specified in any other MAINVIEW SRM selection statement.

You can change SMVARSxx parameters

- by editing the member directly
- by using the MAINVIEW SRM subsystem SET command
- through the **Variables** option on the EZSRMP Parmlib Members menu (see “Using the Variables Option” on page 8-4)

SMVARS_{xx} is an optional member.

Parameter Quick Reference

SET statement variable parameters are described in the following quick reference table. Detailed descriptions can be found in the “Appendix B, Global Parameters.”

Table 8-1 SET Statement Variable Parameters

Parameter	Required	Description
VARIABLE=xxxxxxx	Yes	Specifies the name that is assigned to the variable.

INC/EXC statement variable parameters are described in the following quick reference table.

Table 8-2 INC/EXC Statement Variable Parameters

Parameter	Required	Description
VALUE=xxxxxxxxxxx	Yes	Specifies a value for the variable.

Usage Notes

Variables are specified by SET statements. One or more VALUE statements following the SET statement identify the value(s) of the variable. Any selection parameter values that are valid for filter and rule lists can be specified in the value(s) of the variable. For example:

```
SET VARIABLE=TESTDSN
   VALUE=TEST /
   VALUE= * * . TEST
```

At execution time, variables are substituted for selection statement parameter values. For example:

```
INC DSN=&TESTDSN
INC DSN=&TESTPREFIX&DIVCODE&FILETYPE
```

Note: Blanks or commas are not allowed between concatenated variable names.

See “Name Masking” on page 3-5 for more information on the use of variables in selection and action parameters.

Variables are supported for non-BOOLEAN text fields only. Variable values should not include any parameter keywords, other variable names, embedded blanks, or the INC/EXC operands. For example, the following statements are *invalid*:

```
VALUE=INC DSN=TEST/  (invalid use of INC operand)
VALUE=TEST/  PROD/   (invalid use of an embedded blank)
VALUE=&APMSTR        (invalid use of a variable name)
VALUE='POOL=PROD*'  (invalid use of a parameter keyword)
```

The following examples show you how to define and use variables.

Definition

```
SET VARIABLE=FINCVOL
  VALUE=FINC/
  VALUE=GLAP/
  VALUE=PAY/
  VALUE=RECV9?
```

FINCVOL is a variable that is defined with four volume specifications.

Usage

```
INC VOL=&FINCVOL
```

Definition

```
SET VARIABLE=TESTDSN
  VALUE=**.TEST*
  VALUE=*.*.TSTMSTR
  VALUE=CICSTEST./
```

TESTDSN is a variable that is designed to include any data set names that have final qualifier that begins with TEST, or that the third qualifier is TSTMSTR, or with a high-level qualifier that is CICSTEST.

Usage

```
SMFLSTxx  SET MODE=ACT
           INC DSN=&TESTDSN
```

A variable is an excellent method of defining such widely used specifications as job classes or programmer groups; for example:

```
SET VARIABLE=TESTJOBS
    VALUE=T
    VALUE=S
    VALUE=K
SET VARIABLE=PRODJOBS
    VALUE=P
    VALUE=A
    VALUE=C
    VALUE=U
SET VARIABLE=STGADM
    VALUE=JSMITH/
    VALUE=STGADM/
    VALUE=TMALLEY/
```

The &TESTJOBS and &PRODJOBS variables can be used in any selection criteria to distinguish between test and production jobs. The &STGADM variable can be used in any selection criteria to include or exclude resources for the storage administrator; if the storage administrator changes, only the variable definition must be changed, rather than all the filter and rule lists that are based on the storage administrator's user ID.

Using the Variables Option

After you have identified the criteria for variables at your data center, you may use the Variables option to define variables using the following steps:

- Step 1** Click **Parmlib Members** on the EZSRM Menu. A list of members is displayed.
- Step 2** Select **Variables** from the selection list.
- Step 3** Type **E** in the line command field of the **CMD** column next to an active SMVARS_{xx} member; then press **Enter**.
- Step 4** Edit the member to specify the data center variable definitions.
- Step 5** Press **PF3** to save and exit the member.
- Step 6** Type **R** in the line command field of the **CMD** column next to the SMVARS_{xx} member you edited; then press **Enter**. The member is refreshed.

By refreshing SMVARS_{xx}, you have made it the active variable definition. It will remain active until SVOS is stopped or until you refresh another variable member.

Chapter 9 Defining a Calendar

This chapter provides instructions for defining and using the calendar in MAINVIEW SRM. The following information is included:

Overview.....	9-1
Using the SMCALSxx Member	9-1
Parameters	9-2
Using the Calendars Option	9-3

Overview

Calendar members define company holidays and other non-working days in the calendar year.

Using the SMCALSxx Member

SMCALSxx is the calendar definition parameter member. It is read by MAINVIEW SRM during system startup. The suffix specification in SMMSYSxx indicates the version of SMCALSxx that contains the calendar specifications for the particular configuration of MAINVIEW SRM that is being executed. SMCALSxx is an optional member.

Parameters

Calendar year definitions are specified on SET statements. INC statements following the SET statement identify the non-working days to be included. SMCALSxx parameters identify non-working days that are not to be considered as usage days by DFHSM. SMCALSxx offers much more flexibility in defining non-working days than does DFHSM.

Note: Any day not specified in SMCALSxx is a working day.

You can change SMCALSxx parameters

- by editing the member directly
- by using the MAINVIEW SRM subsystem SET command
- through the **Calendars** option on the EZSRMP Parmlib Members menu (see “Using the Calendars Option” on page 9-3)

Parameter Quick Reference

SET statement calendar parameters are described in the following quick reference table. Detailed descriptions can be found in “Appendix B, Global Parameters.”

Table 9-1 SET Statement Calendar Parameters

Parameter	Required	Description
YEAR=nnnn	Yes	year to which the following days apply

INC/EXC statement calendar parameters are described in the following quick reference table.

Table 9-2 INC/EXC Statement Calendar Parameters

Parameter	Required	Description
FREE=nn.nn- <i>nn.nn</i>	No	from-to range of non-working (free) days
MON=F/W	No	day of week
TUE=F/W	No	day of week
WED=F/W	No	day of week
THU=F/W	No	day of week
FRI=F/W	No	day of week
SAT=F/W	No	day of week
SUN=F/W	No	day of week

Example

The following statement examples show how various calendar parameters may be used.

SET	YEAR=2001 SUN=F FREE=01.05 FREE=12.06 FREE=24.11-30.11	Non-working days in 2001 include: all Sundays, May 1, June 12, and November 24–30. Note that dates are specified day, and then month.
SET	YEAR=2002 FREE=02.01 FREE=13.10 FREE=21.12-26.12	Non-working days in 2002 are January 2, October 13, and December 21–26

Using the Calendars Option

After you have identified the criteria for calendars at your data center, you may use the Calendars option to define calendars as follows:

- Step 1** Click **Parmlib Members** on the EZSRM Menu. A list of members is displayed.
- Step 2** Select **Calendars** from the selection list.
- Step 3** Type **E** in the line command field of the **CMD** column next to an active SMCALSxx member; then press **Enter**.
- Step 4** Edit the member to specify the data center calendar definitions.
- Step 5** Press **PF3** to save and exit the member.
- Step 6** Type **R** in the line command field of the **CMD** column next to the SMCALSxx member that you edited; then press **Enter**.

The member is refreshed. By refreshing SMCALSxx, you have made it the active calendar definition. It will remain active until SVOS is stopped or until you refresh another calendar member.

Chapter 10 Using Diagnostics

This chapter provides instructions for when and how to use diagnostics in MAINVIEW SRM. The following information is included:

Overview	10-1
Using the SMDIAGxx Member	10-1
Parameters	10-2
Using the Diagnostics Option	10-3
Using the Tracing Facility	10-4
Activating Tracing	10-4
How Function Tracing Works	10-6
Additional Trace Options	10-9

Overview

Diagnostic parameters are available for use when you are directed by BMC Software Customer Support to use them. This chapter provides an explanation of how the SMDIAGxx system parameter is used and provides an explanation and example of the tracing facility.

Using the SMDIAGxx Member

SMDIAGxx is the diagnostic parameter member. It is read by MAINVIEW SRM during system start up. The DIAG=xx specification in SMMSYSxx indicates which version of SMDIAGxx contains diagnostic specifications for the current execution of MAINVIEW SRM.

Parameters

Diagnostic definitions are specified by SET statements. INC/EXC statements following the SET statement identify the specific situations to be debugged.

You can change SMDIAGxx parameters

- by editing the member directly
- by using the MAINVIEW SRM subsystem SET command
- through the **Diagnostics** option on the EZSRMP Parmlib Members menu (see “Using the Diagnostics Option” on page 10-3)

FUNCTION, JOB, STEP, and PGM variables are not valid with the ABEND action parameter. The MODULE variable is not valid with the IGNORE action parameter. The FUNCTION variable is not valid with the MODULE action parameter.

The TRACE action parameter (see “Using the Tracing Facility” on page 10-4) can be used instead of //PROTRACE to get an FLST/RLST trace; otherwise, SMDIAGxx should be used only at the direction of BMC Software technical support. The action parameters specified in the SMDIAGxx member override any special JCL DD statements, such as //PROTRACE, as well as the TRACE function parameter.

The IGNORE parameter will override the // PROIGN DD statement. If the DIAG member specifies IGNORE=NO, even the presence of a //PROIGN DD statement will not prevent MAINVIEW SRM functions from being activated.

Parameter Quick Reference

SET statement diagnostic parameters are described in the following quick reference table.

Table 10-1 SET Statement Diagnostic Parameters

Parameter	Required	Description
ABEND=YES/NO	No	forces S0C3 abend when a particular module is entered
DEBUG=YES/NO	No	optional debugging information from a MAINVIEW SRM module
DUMP=YES/NO	No	issues SDUMP if program abend occurs
IGNORE=YES/NO	No	skip this function
MODTRC=YES/NO	No	MAINVIEW SRM module trace
TRACE=YES/NO	No	FLST/RLST trace output

INC/EXC statement diagnostic parameters are described in the following quick reference table.

Table 10-2 INC/EXC Statement Diagnostic Parameters

Parameter	Description
FUNCTION=xxxxxxx	valid MAINVIEW SRM function name (up to eight characters)
JOB=xxxxxxx	job name (up to eight characters)
MODULE=xxxxxxx	valid MAINVIEW SRM module name (up to eight characters)
PGM=xxxxxxx	valid MAINVIEW SRM program name (up to eight characters)
STEP=xxxxxxx	step name (up to eight characters)

Examples

This example shows you how to use diagnostic parameters.

SET TRACE=YES DEBUG=NO MODTRC=NO DUMP=NO INC JOB=RICKM/ FUNCTION=SPACLIMI	sets the trace parameter for the SPACLIMI function and job name RICKM
SET IGNORE=YES TRACE=NO DEBUG=NO MODTRC=NO X DUMP=NO INC FUNCTION=OPTBLKSZ JOB=JERRY/	skips the OPTBLKSZ function on job name JERRY

Using the Diagnostics Option

Once you have identified your criteria for diagnostics, you may use the Diagnostic option to define diagnostics as instructed in the following steps.

- Step 1** Click **Parmlib Members** on the EZSRM Menu. A list of members is displayed.
- Step 2** Select **Diagnostics** from the selection list.
- Step 3** Type **E** in the line command field of the **CMD** column next to an active **SMDIAGxx** member; then press **Enter**.
- Step 4** Edit the member to specify the diagnostic definitions.
- Step 5** Press **PF3** to save and exit the member.

- Step 6** Type **R** in the line command field of the **CMD** column next to the **SMDIAGxx** member you edited; then press **Enter**.

The member is refreshed. By refreshing **SMDIAGxx**, you have made it the active diagnostic list. It will remain active until **SVOS** is stopped or until you refresh another diagnostic member.

Using the Tracing Facility

You can display the **MAINVIEW SRM** selection processing steps with the **MAINVIEW SRM** trace facility. The trace output shows the exact steps a **MAINVIEW SRM** function takes in selecting resources and applying rule list operands.

Activating Tracing

The most flexible mechanism for tracing **MAINVIEW SRM** function activity is to use the **SMDIAGxx** member of the **MAINVIEW SRM** parmlib. This mechanism allows you to specify tracing by job name, function, and so on and gives you full masking capability on each of the selection fields. You can turn tracing on or off dynamically without the need to make **JCL** modifications.

Alternatively, you can declare tracing at the individual function level for a specific job. All **MAINVIEW SRM** activity for that function is traced for any job with the specified name. The **TRACE** action parameter on the function definition **SET** statement specifies the name of the target job:

```
SET  FUNCTION=DSNCHECK
      ACTIVE=YES
      TRACE=PR44LGX           trace job PR44LGX for DSNCHECK
      MSG=I
      FLST= . . . . .
```

Note that the job name that you specify on the **TRACE** parameter must be exact; name masking does not apply. The use of the **SMDIAGxx** parmlib member makes tracing far more flexible than this historic function tracing.

You may also activate tracing by specifying `TRACEDD=ddname` in the `SMMSYSxx` parmlib member and including a DD name in the JCL for the job to be traced. All MAINVIEW SRM functions are traced for the job step that contains the specified DD name. (See “Additional Trace Options” on page 10-9.)

The output of the trace is a series of MAINVIEW SRM messages (SMS0042I), which are directed to the job log or TSO user. Three variations of the 042I message are issued:

- selection start

This message is issued at each SET statement (the beginning of a selection process) and is identified by a question mark as the first character in the text:

```
SMS0042I TRACE ?SET FLST
```

The message text identifies a SET statement and the source (filter list or rule list) of the statement.

- selection failure

This message is issued for each parameter of each INC or EXC statement that *is not satisfied* by the current resource. It is identified by a minus sign as the first character in the text:

```
SMS0042I TRACE - JOB : EMPCRMX = EMPDJE/
```

The message text identifies the selection parameter (JOB, in this example), the current resource name (EMPCRMX), and the parameter value (EMPDJE/).

- selection success

This message is issued for each parameter of each INC or EXC statement that *is satisfied* by the current resource. It is identified by a plus sign as the first character in the text:

```
SMS0042I TRACE + JOB : EMPCRMX = EMPC/
```

The message text identifies the selection parameter (JOB, in this example), the current resource name (EMPCRMX), and the parameter value (EMPC/).

How Function Tracing Works

The following example shows how function tracing works. Figure 10-1 shows a job, EMPCRMX, that is defined in SMFUNCCxx as the job to be traced. This particular job allocates a new data set that is named EMPCRM.SPACLIMI.TEST.

Figure 10-1 **Tracing Example**

```
//EMPCRMX   JOB (3500), 'MORRIS', MSGCLASS=R, CLASS=F
//STEP10   EXEC PGM=IEFBR14
//DD2      DD DSN=EMPCRM.TEST.TEST,
//          DISP=(NEW,DELETE,DELETE),
//          SPACE=(TRK,(100,100)), UNIT=BABDA,
//          DCB=(LRECL=80, BLKSIZE=3120, RECFM=FB, DSORG=PO)
```

One hundred tracks are requested for the new data set; the limit in the rule is 1 kilobyte. The filter list and rule lists are shown in following the diagram.

Tracing a Function

SPACLIMI Function Definitions

SMFUNCxx

```
SET NAME=SPACLIMI
  ACTIVE=YES  MSG=I  SMF=1
  FLST=IV  RLST=09
  DESC='LMIT ALLOC RQST'
  TRACE=EMPCRM
```

SMFLSTIV

```
SET MODE=ACT
  INC JOB=EMPDJE/
  INC DSN=EMPDJE/
```

```
SET MODE=ACT
  INC JOB=EMPEJD/
  INCJOB=EMPJBS/XMODE=JOB
  INCJOB=EMPCRM/XMODE=JOB
```

```
SET MODE=ACT
  INC JOB=CMREX/
  EXC XMODE=TSO
```

SMRLST09

```
SET LIMIT=1KB
  DSN=EMPCRM.*.TEST
```

```
SET LIMIT=10MB
  DSN=EMPCRM.*.PROD
```

Trace Output

```
SMS042I TRACE JOB=EMPCRMX ,FUNC=SPACLIMI
2 SMS042I TRACE ?SET FLST
  SMS042I TRACE - JOB :EMPCRMX = EMPDJE/
  SMS042I TRACE - DSN :EMPCRM.SPACLIMI.TEXT = EMPDJE/
3 SMS042I TRACE ?SET FLST
  SMS042I TRACE - JOB :EMPCRMX = EMPEDJ/
  SMS042I TRACE - JOB :EMPCRMX = EMPJBS/
  SMS042I TRACE + JOB :EMPCRMX = EMPCRM/
  SMS042I TRACE + XMODE:JOB=JOB
4 SMS042I TRACE + INC FLST
  SMS042I TRACE + SET FLST
5 SMS042I TRACE ?SET FLST
  SMS042I TRACE + DSN :EMPCRM.SPACLIMI.TEXT = EMPCRM.*.TEST
6 SMS042I TRACE + INC RLST
  SMS042I TRACE + SET FLST
  SMS042I EMPCRMX,STEP10.DD2,EMPCRM.SPACLIMI.TEST
  REQUESTED SPACE 4636KB EXECEEDS LIMIT
```

Resource **EMPCRM.SPACLIMI.TEST** is selected by the filter list and rule list; the rule is applied, and the data set size is found to exceed the limit set in the action parameter

The numbers in the following explanation correspond to the numbers on the Trace Function diagram:

1. The trace is activated for the SPACLIMI function when job name EMPCRMX is found to match the job name selection parameter on the INC statement.
2. The first SET statement in the filter list is checked (?SET FLST).

The first selection statement fails (SMS0042I TRACE - JOB ...message is issued) because the resource job name EMPCRMX does not match the name mask EMPDJE/.

The second selection statement is checked because it has an OR relationship with the first statement. The selection fails (- DSN ...), because the resource data set name EMPCRM.SPACLIMI.TEST does not match the name mask EMPDJE/.

3. The second filter list SET statement is checked (?SET FLST).

The first selection statement fails (SMS0042I TRACE - JOB ...) because the resource job name EMPCRMX does not match the name mask EMPEJD/.

The second selection statement fails (SMS0042I TRACE - JOB ...) because the resource job name EMPCRMX does not match the name mask EMPJBS/. The second selection parameter in the INC statement is not checked, because it has an AND relationship with the first parameter, which failed.

The third selection statement succeeds (SMS0042I TRACE + JOB ...) because the resource job name EMPCRMX matches the name mask EMPCRM/.

Since the XMODE parameter is on the same line as the JOB parameter, the two conditions have an AND relationship; therefore, the XMODE parameter is also checked. The selection succeeds, because the resource job execution mode (batch) matches the XMODE parameter value of JOB.

4. The filter list check is complete because the resource has been selected. The last SET statement in the filter list is not checked.

The trace shows a successful filter list selection (SMS0042I TRACE +INC FLST) and a successful filter list execution (SMS0042I TRACE +SET FLST).

- The rule list for the SPACLIMI function is now checked (SMS0042I TRACE ?SET RLST).

The selection statement succeeds (SMS0042I TRACE + DSN ...) because the resource data set name EMPCRM.SPACLIMI.TEST matches the name mask EMPCRM.*.TEST.

- The rule list checking is complete because the resource has been selected. The final SET statement in the rule list is not processed.

The limit of 1KB is enforced for this resource.

The trace shows a successful rule list selection (SMS0042I TRACE +INC RLST) and a successful rule list execution (SMS0042I TRACE +SET RLST).

- The SPACLIMI function causes the data set allocation to fail because the requested space for the data set (100 tracks) exceeds the limit of 1KB in the rule.

Additional Trace Options

You can include the following parameters in the SMMSYS xx parmlib member to control tracing activity:

Parameter	Default	Description
MODTRCDD= <i>ddname</i>	PROMTRCE	activates MAINVIEW SRM module entry/exit tracing
TRACEDD= <i>ddname</i>	PROTRACE	activates standard MAINVIEW SRM tracing (as in TRACE=job name)
IGNOREDD= <i>ddname</i>	PROIGN	deactivates tracing for duration of job step
DUMPDD= <i>ddname</i>	PRODUMP	produces a SYS1.DUMP xx dump if MAINVIEW SRM abends
DIAGMSDD= <i>ddname</i>	PRO\$D\$M\$	activates WTO message tracing

If you include the DD name that is specified in the tracing DD parameter in a job step, all MAINVIEW SRM activity for the job step is traced or not traced (if you specify IGNOREDD).

TRACEDD is the same trace as produced by the TRACE parameter in the function definition in SMFUNC xx . IGNOREDD can be useful when a quick override to a MAINVIEW SRM rule is needed. The other three tracing DD parameters are primarily for use by BMC Software personnel in problem determination.

The MAINVIEW SRM connections are not invoked into the operating system early enough for the following functions to be displayed in the DD name-activated trace:

DSNCHECK
SMSACSDC
SMSACSMC
SMSACSSC
SMSACSSG
SMSACSTE

You can use the TRACE=YES parameter in the SMDIAGxx member to activate tracing for these or any other functions.

Chapter 11 Using Filter and Rule List Parameters

This chapter provides instructions for defining and using filter and rule list parameters in MAINVIEW SRM. The following information is included:

Overview	11-1
Using SMFLSTxx - Filter List Parameters	11-2
SET Statement Parameters	11-2
INC/EXC Parameters	11-5
Using SMRLSTxx - Rule List Parameters	11-6
Using the Filter List and Rule List Options	11-9

Overview

The information in this chapter pertains to the following components of MAINVIEW SRM.

- DMS2HSM
- EasyHSM
- EasyPOOL
- EasySMS
- SG-Control
- StorageGUARD
- StopX37/II
- Enterprise Storage Automation

Note: Enterprise Storage Automation generates events through filter and rule list parameters of MAINVIEW SRM functions.

Using SMFLSTxx - Filter List Parameters

SMFLSTxx is the filter list member for one or more functions. It is read by MAINVIEW SRM during system startup. The filter list suffix specification in SMFUNCxx indicates the version of SMFLSTxx that contains specifications for the particular configuration of MAINVIEW SRM that is being executed. SMFLSTxx parameters control selection of resources for a function. SMFLSTxx is an optional member. However, if an SMFLSTxx member is not specified for a function, no resources are selected for the function.

Filter lists are defined on SET statements and INC/EXC statements. INC/EXC statements that follow the SET statement identify the resources for selection. The SET statement parameters apply only to the resources that are selected by the INC/EXC statement parameters that follow them.

You can change SMFLSTxx parameters

- by editing the member directly
- by using the MAINVIEW SRM system SET command
- through the **Filter List** and **Rule List** options on the EZSRMP Parmlib Members menu (see “Using the Filter List and Rule List Options” on page 11-9)

SET Statement Parameters

SET statement filter list parameters are described in the following table.

Table 11-1 SET Statement Parameter Quick Reference

Parameter	Required	Description
MODE=ACT/INACT/SIM	Yes	status of the function Simulate mode (SIM) is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.
MSG=I/W/E/S/N	No	level of messages to be generated The MSG parameter is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.
SMF=I/W/E/S/N	No	level of messages to be generated to SMF The SMF parameter is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.

MODE

Purpose: The MODE parameter specifies the status of the function for the resources that are selected by following INC/EXC statements. When MODE=INACT is specified, any selected resources are bypassed for processing by the function. When MODE=SIM is specified, selected resources are processed in simulation mode, in which the action is not applied but a message is issued (depending on the MSG parameter) to indicate the action that would be applied if MODE=ACT were specified.

The MODE parameter is the most significant filter list SET statement parameter. The MODE parameter defines the processing mode for the selected resources. The MODE parameter can be set to one of the following values:

- active (the function acts on the selected resources)
- inactive (the function does nothing for the selected resources)
- simulate (the function reports activity as if it were active, but it does not actually take any action for the selected resources)

Different sets of selected resources can have different processing modes for a function.

Note: Simulate mode (SIM) is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.

Syntax: MODE=[ACT | INACT | SIM]

- ACTive
- INACTive
- SIMulate

Required: Yes

Default: None

MSG

Purpose: The MSG parameter specifies the message generation option for resources that are selected by the following parameters. Informational and error messages can be produced, or all messages can be suppressed. Note that the MSG option on the function definition in the SMFUNCxx member is overridden by this option for specific selected resources.

Note: The MSG parameter is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.

Syntax: MSG=[I / W / E / S / N]

I	Informational and error messages
W	Warning messages
E	Error messages only
S	Severe messages
N	No messages

Required: No

Default: The MSG parameter on the function definition in SMFUNCxx.

SMF

Purpose: The SMF parameter specifies the SMF message generation option for resources that are selected by the following parameters. Informational and error messages can be written to the SMF data set, or all messages can be omitted from the SMF data set. Note that the SMF option on the function definition in the SMFUNCxx member is overridden by this option for specific selected resources.

Note: The SMF parameter is used only for EasyHSM, EasySMS, EasyPOOL, StopX37/II, and DMS2HSM.

Syntax: MSG=[I / W / E / S / N]

I	Informational and error messages
W	Warning messages
E	Error messages only
S	Severe messages
N	No messages

Required: No

Default: None

INC/EXC Parameters

The INC/EXC parameters for the filter and rule lists are extensive and are explained in the *MAINVIEW SRM Reference Summary*.

You can specify multiple INC/EXC parameters. When you specify the parameters on the same line, selection is based on all parameters being true; that is, a Boolean AND condition applies to the parameter criteria. The resource is selected if the first parameter and the second parameter are true.

Example

For example:

```
SET MODE=ACT
INC JOB=*TST XMODE=JOB
```

Resources are selected if the job name ends with TST *and* the job is executing in batch mode.

When you specify multiple parameters on separate lines, selection is based on one parameter or the other being true; that is, a Boolean OR condition applies to the parameter criteria. The resource is selected if the first parameter or the second parameter is true.

Example

For example:

```
SET MODE=ACT
INC DSN=ISR. /
INC DSN=ISP. /
```

Resources are selected if the data set name begins with ISR *or* if the data set name begins with ISP.

When a resource has satisfied the selection criteria on an INC statement, the resource is selected for processing and no further examination of the filter list is performed. When a resource has satisfied the criteria on an EXC statement, however, the resource is excluded only from that filter list entry; the resource is then passed to the next SET statement in the filter list for consideration. Excluded resources are considered by each entry in the filter list.

- An INC statement selects a resource and exits the filter list.
- An EXC statement rejects a resource for the current entry only and continues checking the filter list.

You can use MAINVIEW SRM name masking in all parameters for the INC and EXC statements. However, you cannot use name masking on SET parameters because a SET statement parameter must specify a definite value. See the information in Chapter 3, “Understanding General Syntax Rules,” for full details on selection parameter syntax, including name masking and ordering of statements.

Using SMRLSTxx - Rule List Parameters

SMRLSTxx is the rule list parmlib member for one or more functions. It is read by MAINVIEW SRM during system startup. The rule list suffix specification in SMFUNCxx indicates the version of SMRLSTxx that contains specifications for the particular configuration of MAINVIEW SRM that is being executed. SMRLSTxx is an optional member. If an SMRLSTxx member is not specified for a function, the default for the function is used unless it does not have a default. If there is not a default for the function and an SMRLSTxx member is not specified, the function is not processed.

SMRLSTxx parameters control the actions of a function against the storage management objects or properties the one selected for a function, and they control how MAINVIEW SRM responds to a request from the operating system or DFSMS. For example, an SMRLSTxx member may modify the blocksize or retention period for a data set, or it may reject a volume for data set allocation.

The following functions do not use a rule list because they perform processing that does not need to be tailored for specific resources.

- FORCECAT
- OPENEMPT
- SUPJSCAT
- FRAGCNTL
- SUPVOLRF
- MODDELET

- SMSMCREN
- TAPEDEFR

A rule list is composed of one or more rules. A rule is composed of a single SET statement and one or more INC/EXC statements. The SET statement identifies a data or system property to be modified and specifies a value for it.

For example, the following SET statement (for the OPTBLKSZ) function specifies that the block size for new data sets is 10,796:

```
SET BLKSIZE=10796                (OPTBLKSZ function)
```

In the next example, the SET statement (for the SETEXPT function) specifies an expiration date of (EXPDT) 94180, regardless of any JCL specification (REPLACE), for new data sets:

```
SET EXPDT=94180 REPLACE=YES      (SETEXPDT function)
```

In the next example, the SET statement (for DSNCHECK function) specifies that the maximum number of qualifiers in a data set name is six.

```
SET MAXQLF=6                    (DSNCHECK function)
```

The INC/EXC statements in rule lists operate in the same way as INC/EXC statements in filter lists to select the specific resources to modify with the SET statement parameter. In the next example, the INC statement (for the OPTBLKSZ function) specifies a mandatory block size for new data sets that are larger than 10 megabytes.

```
SET BLKSIZE=10796                (OPTBLKSZ function)
INC MAXSIZE>10MB
```

In the next example, the INC statement (for SETEXPDT function) specifies that new data sets in any of the test pools (POOL) or with a second job account field (JOBACCT2) of TEST are assigned an expiration date of 94180.

```
SET EXPDT=94180                  (SETEXPDT function)
INC POOL=TEST/
INC JOBACCT2=TEST
```

At least one INC/EXC selection parameter must be specified for a rule. A SET statement without an INC/EXC statement is not applied to any resources.

Resource selection for a rule is a two step process.

- Step 1** A resource is selected by the function's filter list
- Step 2** The resource is then passed on to the rule list for further checking.

A resource must be included in both sets of selection results for the function SET parameters to be applied.

When a resource has satisfied the selection criteria on an INC statement, the resource is selected for processing and no further examination of the rule list is performed. When a resource has satisfied the criteria on an EXC statement, however, the resource is excluded only from that rule list entry; the resource is then passed to the next SET statement in the rule list for consideration. Excluded resources are considered by each entry in the rule list.

- An INC statement selects a resource and exits the rule list.
- An EXC statement rejects a resource for the current entry only and continues checking the rule list.

You can use MAINVIEW SRM name masking in all parameters for the INC and EXC statements. However, you cannot use name masking cannot be used on rule SET parameters because a SET statement parameter must specify a definite value. See the information in Chapter 3, "Understanding General Syntax Rules," for full details on selection parameter syntax, including name masking and ordering of statements.

Parameter Explanations

The parameters for the rule and filter lists are extensive, and are explained in the *MAINVIEW SRM Reference Summary*.

The following example shows how pools are selected and controlled by using the filter and rule lists.

SMFLSTxx contains		
SET	MODE=SIM MSG=E INC POOL=TEST/	Select all data sets that reside in the test pool.
SMRLSTxx contains		
SET	LIMIT=5M INC VOL=TEST5/	All data sets residing in any test pool are selected by the filter list. The two rules limit these data sets to 5MB on volumes TEST5A and TEST5B and to 2MB on other volumes beginning with TEST.
SET	LIMIT=2M INC VOL=TEST/	

Using the Filter List and Rule List Options

Before you can use a function, you must first perform the following actions:

- Define a filter list (in an SMFLST xx member), with MODE=ACTIVE, and with INCLUDE parameters to apply the function to a set of resources (data sets, volumes, pools, jobs, and so forth).
- Define a rule list (in an SMRLST xx member), specifying how resources are affected.

Use the **Filter List** and **Rule List** options to define the parameters in SMFLST xx and SMRLST xx , as instructed in the following steps.

- Step 1** Click on **Parmlib Members** on the EZSRM Menu. A list of members displays.
- Step 2** Select **Filter List** or **Rule List** from the selection list.
- Step 3** Type **E** in the line command field of the **CMD** column next to an active SMFLST xx or SMRLST xx member, then press **Enter**.
- Step 4** Edit the member to specify the filter list or rule list definitions.
- Step 5** Press **PF3** to save and exit the member.
- Step 6** Type **R** in the line command of the **CMD** column next to the SMFLST xx or SMRLST xx member you edited, then press **Enter**. The member is refreshed.

By refreshing SMFLST xx or SMRLST xx , you have made it the active filter or rule list. It will remain active until SVOS is stopped or until you refresh another member.

Chapter 12 Monitoring Pools and SMS Storage Groups

This chapter provides information about using the Pools, Storage Group, and SMS Pools Monitor options in MAINVIEW SRM. The following information is included:

Overview	12-2
Monitor Pool Views	12-3
Monitor Volume Space Views	12-4
Monitor Volumes in a Pool View	12-5
Monitor SMS Storage Group Views	12-6
Monitor SMS Storage Group Volume Space View	12-6
Monitor SMS Pool Views	12-7
Monitor Performance Views	12-7
Monitor Volume Performance Summary Views	12-9
Monitor Volume Performance Statistics View	12-10

Overview

The real time monitor options display views and submenus that give you real-time storage information.

When you access MAINVIEW SRM, the EZSRM Menu, shown in Figure 12-1, is displayed initially. The Pools, SMS Storage Groups, and SMS Pools options display initial views from which many views are available through actions, hyperlinks, and commands.

Figure 12-1 EZSRM Menu

```

14MAY2001 09:43:29 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> PAGE
CURR WIN ==> 1           ALT WIN ==>
>W1 =EZSRM=====SJSG=====*=====14MAY2001==09:43:29===MVSRM===D===1
                                EZSRM  Menu

SRM Real Time Monitor  +-----+ SRM Historical Data
                        |         |
. Pools                 | Place cursor on | > Historical Space
. SMS Storage Groups    | menu item and  | > Historical Performance
. SMS Pools             | press ENTER   | > EasyHSM
> RAID Configurations  +-----+ > SGControl Applications
> Storage Performance                                     > SMF Report Library

SRM Administration                                         SRM Tools and Menus

> Parmlib Members                                         > Workbench
. Functions                                                . MVSRM View List
. SRM Component Status                                     . MVSRM Batch Reports
                                                            . MainView Messages
                                                            . Return....
    
```

The following views are described in this chapter:

View	Description	Page
MPOOL	displays all pools that are defined in the system	12-3
MVOLSPC	provides space information for volumes in selected pools	12-4
MPOOLVOL	lists all volumes and unit addresses that belong to the selected pool	12-5
MSMSG	lists all SMS storage groups that are defined in the system	12-6
MVOLSSPC	provides space information for volumes in the selected SMS storage group	12-6
MSMSP	lists all the SMS pools defined in the system	12-7

View	Description	Page
MVOLPER	displays performance data (over the last 5 minute interval, maintained by the DASD Performance Option (DPO)) for a selected pool, SMS pool, EMC physical device, IBM physical device, SMS storage group, or VOLSER	12-7
MVOLPERS	provides a summary view that is grouped by volume serial number; this view shows the average and maximum statistics for each volume across the SSI context	12-9
MVOLPAV	displays performance statistics that gathered by the DASD Performance Option (DPO) for the base and all alias UCBs of a parallel access volume (PAV)	12-10

Monitor Pool Views

The Pools option displays all the pools that are defined in the system, from this view you can select specific pools for more information.

View Type	View Name	View Invocation
Tabular	MPOOL	MPOOL

The following actions are available from this view:

Action	Takes you to	View Name
D	the definition of the selected pool	
S	space information for volumes in the selected pool	MVOLSPC
V	a list of all volumes and unit addresses that belong to the selected pool	MPOOLVOL
P	performance data (over the last 5 minute interval, maintained by DASD Performance Option (DPO)) for the pool	MVOLPER

Monitor Volume Space Views

The MVOLSPC view lists current space information for volumes in a selected pool or SMS pool.

View Invocation: MVOLSPC <i>grouptype groupname</i>		
View	Group Type	Group Name
MVOLSPC	POLS SMSS	8-byte pool name 8-byte SMS pool name

Example

```
MVOLSPC POLS ALLVOLS
```

displays space information for the volumes in pool ALLVOLS

Example

```
MVOLSPC SMSS TESTSMS
```

displays space information for the volumes in SMS pool TESTSMS

No actions are available from this view.

Monitor Volumes in a Pool View

The MPOOLVOL view lists all volumes and unit addresses that belong to a selected pool, SMS pool, or SMS storage group.

View Invocation: MPOOLVOL <i>grouptype groupname</i>		
View	Group Type	Group Name
MPOOLVOL	POLV SMSV SSGV	8-byte pool name 8-byte SMS pool name 30-byte SMS storage group

Example

```
MPOOLVOL POLV ALLVOLS
```

displays volume information for the volumes in pool ALLVOLS

Example

```
MPOOLVOL SMSV TESTSMS
```

displays volume information for the volumes in SMS pool TESTSMS

Example

```
MPOOLVOL SSGV STG2
```

displays volume information for the volumes in SMS storage group STG2

No actions are available from this view.

Monitor SMS Storage Group Views

The SMS Storage Groups option lists all SMS storage groups that are defined in the system.

View Type	View Name	View Invocation
Tabular	MSMSG	MSMG

The following actions are available from this view:

Action	Takes you to	View Name
P	performance data (over the last 5 minute interval, maintained by DPO) for the SMS group	MVOLPER
S	space information for volumes in the selected SMS storage group	MVOLSSPC
V	a list of all volumes and unit addresses that belong to the selected SMS storage group	MPOOLVOL

Monitor SMS Storage Group Volume Space View

The MVOLSSPC view provides space information for volumes in a selected SMS storage group.

View Invocation: MVOLSSPC <i>grouptype groupname</i>		
View	Group Type	Group Name
MVOLSSPC	SSGS	30-byte SMS storage group

Example

```
MVOLSSPC SSGS STG2
```

displays space information for the volumes in storage group STG2

No actions are available from this view.

Monitor SMS Pool Views

The SMS Pools option lists all SMS pools that are defined in the system.

View Type	View Name	View Invocation
Tabular	MSMSP	MSMP

The following actions are available from this view:

Action	Takes you to	View Name
D	contents of storage that correspond to the definition of the selected SMS pool	
P	performance data (over the last 5 minute interval, maintained by DPO) for the SMS group	MVOLPER
S	space information for volumes in the selected SMS group	MVOLSPC
V	a list of all volumes and unit addresses that belong to the selected SMS group	MPOOLVOL

Monitor Performance Views

The MVOLPER view displays current performance statistics for volumes in a selected pool, SMS pool, EMC physical device, IBM physical device, SMS storage group, or VOLSER.

View Type	View Name	View Invocation
Tabular	MVOLPER	MVOLPER <i>groupname</i>

View Invocation: MVOLPER <i>groupname</i>		
View	Group Type	Group Name
MVOLPER	SVOL POOL SMSG SMSP EMCP IBMD	6-byte volume serial number 8-byte pool name 30-byte SMS group name 8-byte SMS pool name 9-byte EMC physical ID in the following format: <ul style="list-style-type: none"> • 5-byte serial number • 2-byte director number in Hexidecimal • 2-byte SCSI ID in Hexidecimal 10-byte EMC physical ID in the following format: <ul style="list-style-type: none"> • 5-byte serial number • 5-byte drawer number or rank ID

Example

MVOLPER POOL APPL4

displays performance information for the volumes in pool APPL4

Example

MVOLPER SMSG SMS1

displays performance information for the volumes in the SMS group that is named SMS1

Example

MVOLPER EMCP 010470AC0

displays performance information for the volumes on EMC Symmetrix with serial number 01047 on director 0A and SCSI C0

The following actions are available on the MVOLPER view:

Action	Takes you to	View Name
P	performance statistics gathered by the DASD Performance Option (DPO) for the base and all alias UCBs of a parallel access volume (PAV)	MVOLPAV

Monitor Volume Performance Summary Views

The MVOLPERS view is a summary view that is grouped by volume serial number. This view shows the average and maximum statistics for each volume across the SSI context a selected pool, SMS pool, EMC physical device, IBM physical device, SMS storage group, or VOLSER.

View Type	View Name	View Invocation
Summary	MVOLPERS	MVOLPERS <i>groupname</i>

View Invocation: MVOLPERS <i>groupname</i>		
View	Group Type	Group Name
MVOLPERS	SVOL POOL SMSG SMSP EMCP IBMD	6-byte volume serial number 8-byte pool name 30-byte SMS storage group name 8-byte SMS pool name 9-byte EMC physical ID in the following format: <ul style="list-style-type: none"> • 5-byte serial number • 2-byte director number in Hexidecimal • 2-byte SCSI ID in Hexidecimal 10-byte EMC physical ID in the following format: <ul style="list-style-type: none"> • 5-byte serial number • 5-byte drawer number or Rank ID

Example

```
MVOLPERS POOL APPL4
```

displays a summary of performance information for the volumes in pool APPL4

Example

```
MVOLPERS SMSG SMS1
```

displays a summary of performance information for the volumes in the SMS group named SMS1

Example

MVOLPERS EMCP 010470AC0

displays a summary of performance information for the volumes on EMC Symmetrix with serial number 01047 on director 0A and SCSI C0

Tip: Click on the **NumSYS** field to expand the summarized line in MVOLPER form.

Monitor Volume Performance Statistics View

The MVOLPAV view displays performance statistics that are gathered by the DASD Performance Option (DPO) for the base and all alias UCBs of a parallel access volume (PAV).

View Type	View Name	View Invocation
Summary	MVOLPAV	MVOLPAV <i>volser</i>

No actions are available on this view.

Chapter 13 Processing SMF Records

This chapter provides information about processing SMF records in MAINVIEW SRM. The following information is included:

Overview	13-1
Using the PRSMF000 Utility	13-2
SMF Record Layout	13-2

Overview

All MAINVIEW SRM message activity displays on the job log. In addition, all these activity messages can be written to the SMF data set. The messages that are written to SMF contain the same information as those that are written to the job log and can be accumulated for any length of time. The messages can be

- extracted by the SMF ID code with the IBM IFASMFDP utility
- analyzed with the MAINVIEW SRM PRSMF000 utility
- reported on by function

Note: StopX37/II provides a separate logging facility that logs an SMF record for each attempted job recovery.

Using the PRSMF000 Utility

The PRSMF000 utility sorts the messages by function and date, and it writes one PDS member per function. The MAINVIEW SRM ISPF interface allows you to view of the processed SMF data.

Note: In Release 3.5 of ProSMS, the SMF record format was expanded to accommodate the StopX37/II recovery information that was used to generate the SMFX37B report. To provide compatibility between the old and new SMF record formats, program SMSMFCNV was added to the LOG ANALYSIS job flow. SMSMFCNV converts the new SMF record format (Release 3.5 and later) to the old format (Release 3.4 and earlier), which is processed by program PRSMF000 to produce the SMF message library. This program allows SMF message records from any release to be processed by PRSMF000. However, the ProSMS SMF message library should be viewed by using the ProSMS ISPF interface from the same release that created the records.

A sample of the required JCL for SMF record processing is supplied in the JCLSMFL member of the *?prefix.BBSAMP* library. The SMF record layout is also supplied, and an example of the record layout is shown in Figure 13-1. The production of SMF messages is controlled by action parameters in the following members:

- SMMSYSxx The SMF record ID can be specified with the SMFID parameter. See SMFID in the “Appendix B, Global Parameters.”

- SMFUNCxx Function definitions control the level of produced messages. See the “Function Member Parameters” on page B-77

- SMFLSTxx Message production can also be controlled at the level of resource selection. See “Using SMFLSTxx - Filter List Parameters” on page 11-2.

SMF Record Layout

Figure 13-1 shows the record layout for SMF.

Note: This macro is supplied as member @SMF on the distribution tape data set *?prefix.BBSAMP*.

Figure 13-1 SMF Record Layout (Part 1 of 5)

```

@SMF DSECT
*
*   SMF REC - PROSMS SMF RECORD LAYOUT.
*
* DSECT NAME - SMFPRHDR (DEFAULT).
* CREATED BY - SMM90050.
*   SUBPOOL - N/A.
*   SIZE - SET BY CALLER OF SMFPRHDR.
* SERIALIZED - N/A.
* RESIDENCY - ENVIRONMENT DEPENDANT.
* POINTED TO - N/A.
*
*   FUNCTION - FORMAT OF THE RECORD TO BE WRITTEN TO SMF BY VARIOUS
*   PROSMS COMPONENTS.
*
SMFPRHDR DSECT
SMFPRLEN DC   H'0'           RECORD LENGTH FOR VB OR VBS.
SMFPRSEG DC   H'0'           SEGMENT DESCRIPTOR FOR VBS.
*
SMFPRFLG DC   B'10000010'    DEFAULT TO VS/2 XA.
SMFPRRRF EQU  X'80'          NEW SMF RECORD FORMAT.
SMFPRSUT EQU  X'40'          SUBTYPE FIELD IS USED.
SMFPRV4 EQU   X'10'          MVS/ESA V4.
SMFPRESA EQU  X'08'          MVS/ESA.
SMFPRVXA EQU  X'04'          MVS/XA.
SMFPROS EQU   X'02'          OS/VS2.
SMFPRBFY EQU  X'01'          BERRYMAN MACHINE (PR/SM).
*
SMFPRRTY DC   FL1'0'         RECORD TYPE (NUMBER).
SMFPRIME DC   FL4'00'        TIME IN HUNDRETHS REC WRITTEN.
SMFPRDTE DC   PL4'00'        DATE RECORD WAS WRITTEN.
SMFPRSID DC   CL4' '         SYSTEM ID FROM SMCASID IN SMCA.
*
*   END OF STANDARD SMF HEADER (W/O SUBTYPES)
*
SMFPRSSI DC   CL4' '         SUBSYSTEM ID.
*
SMFPRSTY DC   H'0'           RECORD SUBTYPE.
SMFSTMSG EQU  1              - MESSAGE SMF RECORD.
SMFSTREC EQU  2              - RECOVERY PERFORMED.
*
*   END OF STANDARD SMF HEADER (W/ SUBTYPES)
*
SMFPRTRN DC   AL2(0)         # OF TRIPLETS IN RECORD.
DC   H'0'           RESERVED.
*
*   "TRIPLET" IS A TERM FOR 3 CONTIGUOUS FIELDS THAT DESCRIBE A
*   SUB-SECTION OF THIS RECORD.  THE 3 FIELDS ARE THE OFFSET FROM THE
*   BEGINNING OF THE RECORD, THE LENGTH OF A DATA SECTION, AND THE
*   COUNT OF THE NUMBER OF OF DATA SECTIONS THAT MAKE UP THE
*   SUB-SECTION.  IT IS POSSIBLE THAT A SUB-SECTION WILL NOT BE
*   GENERATED.  IN THAT CASE THE COUNT WILL BE ZERO.  FOR ANY RECORD
*   WITH TRIPLETS, THE LAST PART OF THE HEADER SECTION BEGINS WITH
*   THE FIRST TRIPLET.  THIS TRIPLET DESCRIBES THE PRODUCT SECTION.
*
SMFPRTRP DS   0X             ** PRODUCT SECTION TRIPLET.
SMFPRPRS DC   AL4(0)         OFFSET TO THE PRODUCT SECTION.
SMFPRPRL DC   AL2(0)         LENGTH OF PRODUCT SECTION.
SMFPRPRN DC   H'1'          NUMBER OF SECTIONS (ALWAYS 1).
*
SMFPRJCS DS   0X             ** JOB INFORMATION CONTROL SECTION
SMFPRJIO DC   AL4(0)         OFFSET TO THE JOB INFO SECTION.
SMFPRJIL DC   AL2(0)         LENGTH OF JOB INFO SECTION.
SMFPRJRN DC   H'1'          NUMBER OF SECTIONS (ALWAYS 1).
*
SMFPRMSG DS   0X             ** MESSAGE CONTROL SECTION.
SMFPRMCS DC   AL4(0)         OFFSET TO THE MSG CNTL SECTION.
SMFPRMCL DC   AL2(0)         LENGTH OF MSG CNTL SECTION.
SMFPRMCN DC   Y(0)          NUMBER OF SECTIONS.
*
SMFPRDAS DS   0X             ** DATA SECTION.
SMFPRASS DC   AL4(0)         OFFSET TO THE DATA SECTION.
SMFPRASL DC   AL2(0)         LENGTH OF THE DATA SECTION.

```

Figure 13-1 SMF Record Layout (part 2 of 5)

```

SMFPRASN DC    Y(0)                NUMBER OF SECTIONS.
*
SMFPRDAC DS    0X                    ** DATA CONTROL SECTION.
SMFPRDCS DC    AL4(0)                OFFSET TO THE DATA CNTL SECTION
SMFPRDCL DC    AL2(0)                LENGTH OF THE DATA CNTL SECTION
SMFPRDCN DC    Y(0)                NUMBER OF SECTIONS.
*
          ORG    ,                    MAKE SURE LOC CNTR AT MAX.
          DS     0F                    FILL TO FULLWORD.
*
SMFPRHDR_LEN  EQU (*-SMFPRHDR)       L(HEADER SECTION).
SMFPRTRP_LEN  EQU (*-SMFPRTRP)       L(TRIPLET SECTION).
SMFPRTR# EQU   (*-SMFPRTRP)/8        MAX COUT OF TRIPLETS.

*
*   PROSMS PRODUCT CONTROL SECTION.
*
SMFPRPRO DSECT
SMFPRPRD DC    CL8'PROSMS'           PRODUCT NAME.
SMFPROVS DC    X'00'                 PRODUCT VERSION NUMBER.
SMFPRMPRE DC   XL3'00'               MESSAGE PREFIX
SMFPRMODE DC   XL1'00'               MODE
SMFPRRS1 DC    XL3'00'               RESERVED.
SMFPRMVS DC    CL8' '                MVS SOFTWARE LEVEL FROM CVT-40.
SMFPRXNM DC    CL8' '                SYSPLEX NAME (FROM ECVTSPLX).
SMFPRSNM DC    CL8' '                NAME OF THE CURRENT SYSTEM.
SMFPRFUN DC    CL8' '                FUNCTION NAME.
SMFPRMOD DC    CL8' '                MODULE NAME.
SMFPRCSN DC    CL8' '                CSECT NAME.
SMFOSYS DC    CL12' '                OS ID.
SMFOSFM DC    CL12' '                FMID.
SMFDPL DC    CL12' '                DFP LEVEL.
SMFPCPUM DC    CL12' '                CPU MODEL.
SMFPCPUS DC    CL12' '                CPU SERIAL NUMBER.
SMFTSOE DC    CL12' '                TSO/E LEVEL.
SMFSSID DC    CL4' '                SUBSYSTEM ID USED BY LOADER.
SMFPRSCL EQU   (*-SMFPRPRO)          LENGTH OF PRODUCT SECTION.
*
*   PROSMS JOB INFORMATION SECTION.
*
SMFPRJOB DSECT
SMFJNAME DC    CL8' '                JOB NAME.
SMFPRJID DC    CL8' '                JES JOB ID.
SMFPRJS DC     CL8' '                CURRENT JOBSTEP.
SMFPRJPS DC    CL8' '                CURRENT JOB PROCSTEP.
SMFPRJSP DC    CL8' '                PROGRAM NAME ON EXEC CARD.
SMFRGRP DC     CL8' '                RACF GROUP
SMFRUID DC     CL8' '                RACF USERID
SMFJST DC      F'0'                 JOB START TIME
SMFJSD DC      F'0'                 JOB START DATE
SMFJCLAS DC    C' '                 JOB CLASS
          DS     0D                    DOUBLEWORD ALIGNMENT.
SMFPRJSL EQU   (*-SMFPRJOB)          LENGTH OF PRODUCT SECTION.
*
*   -----
*   M E S S A G E   C O N T R O L   S E C T I O N   ( M C S )
*
PSMPMCS DSECT
PSMMSGLN DC    Y(0)                LENGTH OF MESSAGE.
PSMMSGRL DC    Y(0)                RESERVED.
PSMMSGTX DS    0C                    START OF MESSAGE TEXT.
*
*   -----
*   D A T A   C O N T R O L   S E C T I O N   ( D C S )
*
*   THIS IS BASICALLY A COPY OF THE GWA.  USE THE DCS TRIPLET TO
*   DETERMINE THE AVAILABLE SET OF FIELDS ALONG WITH A DETERMINATION
*   OF ITS INCLUDING INTO THE RECORD.  IF THE LENGTH IS ZERO, IT IS
*   NOT IN THE RECORD.
*
*   -- NOTE -- ANY CHANGE MADE TO THE GWA IN THE COPY BOOK SMM4WORK
*   MUST BE MADE HERE AS WELL.

```

Figure 13-1 SMF Record Layout (part 3 of 5)

PSMPDCS	DSECT				
PSMMSGBLEN	EQU	256			MESSAGE BUFFER LEN - MUST BE
*					SAME AS PMSGBLEN!!!!!!
*					
PSMEYE	DS	CL4			SMM4
PSMRDW	DS	H			LENGTH OF RECORD
PSMRDWZ	DS	H			HALFWORD INITIALIZED TO ZERO
PSMSMUV	DS	F			SMUV ADDRESS
PSMSMAT	DS	F			SMAT ADDRESS
PSMSMFNM	DS	C			SMF "NAME" FIELD
PSMSMFRT	DS	C			SMF RECORD TYPE
PSMSMFTI	DS	XL4			TIME OF RECORD
PSMSMFDA	DS	XL4			DATE OF RECORD
PSMSMFSI	DS	CL4			SYSTEM IDENTIFICATION (SID)
PSMSMFJN	DS	CL8			JOB NAME
PSMSMFET	DS	XL4			ENTRY TIME TO RDR (.01 SECS)
PSMSMFED	DS	XL4			ENTRY DATE TO RDR (00YYDDDF)
PSMX37	DS	CL4			C'X37'
PSMVERS	DS	CL2			RELEASE/VERSION NUMBERS
PSMCPUTI	DS	F			CUMMULATIVE CPU TIME
PSMSTEPT	DS	A			STEP CPU TIME
PSMERROR	DS	C			Y=RECOVERY NOT ATTEMPTED
PSMFFLAG	DS	C			TYPE OF RECOVERY
PSMFRED	EQU	00			REDUCE (SPACSECR)
PSMFSEC	EQU	04			ADD SECONDARY (SPACSECA)
PSMFSWI	EQU	08			VOLUME ADD (SPACVOLA)
PSMFSNA	EQU	12			REDUCE PRIMARY (SPACPRIM)
PSMFRCAT	EQU	16			RECAT (NOCATLG2)
PSMFVRED	EQU	20			VSAM REDUCE (SPACSECR)
PSMFSECI	EQU	24			SECNDRY INCREASE (SPACSECI)
PSMFSECB	EQU	28			SECNDRY BESTFIT (SPACSECB)
PSMFSWIR	EQU	32			VOL ADD SECRED (SPACSWIR)
PSMSKIP	DS	C			SKIP FLAG BITS
PSMSKIPEXCP	EQU	X'80'	1..	SKIP EXCP CHECK
PSMSKIPNOTE	EQU	X'40'	.1..	SKIP NOTE/POINT CHECK
PSMSKIPDISP	EQU	X'20'	..1..	SKIP DISP CHECK
PSMSKIPDSN	EQU	X'10'	...1..	SKIP DSNAM CHECK
PSMSKIPENQU	EQU	X'08'	1..	SKIP DSNAM ENQ CHECK
PSMSKIPDC	EQU	X'04'1..	SKIP DEVICE CHARACTERISTICS
PSMSKIPCTG	EQU	X'02'1..	SKIP CONTIG CHECK
*SPARE	EQU	X'01'1	SPARE
*					
PSMSTEPN	DS	AL1			STEP NUMBER
PSMDATE	DS	F			CURRENT DATE
PSMTIME	DS	F			CURRENT TIME
PSMBTIME	DS	F			CURRENT TIME (BINARY)
PSMYEAR	DS	F			CURRENT YEAR
PSMNDAY	DS	F			CURRENT # OF DAYS SINCE
PSMCDAY	DS	CL10			CURRENT DAY
PSMCTIME	DS	CL8			CURRENT TIME (CHARACTER)
PSMSTYP	DS	CL3			SPACE ALLOCATION TYPE
PSMIDCVS	DS	C			C'Y' = REDUCE PRIMARY FOR
*					NON-JCL-ALLOC VSAM (IDCAMS)
*					
*					INFORMATION GATHERED FOR USE DURING SELECTION PROCESSING.
*					
PSMJNAME	DS	CL8			NAME OF JOB
PSMJACT1	DS	CL20			JOB ACCT #1
PSMJACT2	DS	CL20			JOB ACCT #2
PSMJACT3	DS	CL20			JOB ACCT #3
PSMSACT1	DS	CL20			STEP ACCT #1
PSMSACT2	DS	CL20			STEP ACCT #2
PSMSACT3	DS	CL20			STEP ACCT #3
PSMPGMNM	DS	CL20			PROGRAMMER NAME FIELD
PSMRGRP	DS	CL8			RACF GROUP
PSMRUID	DS	CL8			RACF USERID
PSMACF2_USERID	DS	CL24			ACF2 USERID
PSMOWNER	DS	CL40			VSAM OWNER ID
PSMDSNM	DS	CL44			NAME OF DATASET
PSMCAT	DS	CL44			CATALOG NAME DATASET IS IN
PSMDSORG	DS	CL3			DATASET ORGANIZATION
PSMDSTYP	DS	CL4			DATASET TYPE (PERM/TEMP/GDG)

Figure 13-1 SMF Record Layout (part 4 of 5)

PSMUNITT	DS	CL8	UNIT TYPE (IE. 3350)
PSMDDNM	DS	CL8	ORIGINAL DDNAME
PSMJTYP	DS	CL3	TYPE OF TASK (JOB,STC,TSU)
PSMDISP1	DS	CL3	DATASET DISPOSITION
PSMLTYPE	DS	CL3	ALLOCATION TYPE (CYL,TRK,BLK)
PSMDISP2	DS	CL7	SCHED. DISP (CATLG,KEEP..)
PSMDISP3	DS	CL7	ABTERM DISP (CATLG,KEEP..)
PSMREL	DS	CL1	C'Y' - RLSE CODED IN SPACE PAR
PSMMDAY	DS	CL10	JOB START DAY OF WEEK
PSMMJDAT	DS	CL6	JOB START JULIAN DATE
PSMMDATE	DS	CL8	JOB START DATE (CHARACTER)
PSMMTIME	DS	CL8	JOB START TIME (CHARACTER)
PSMEXPDT	DS	CL8	EXPIRATION DATE
PSMRETPD	DS	F	RETENTION PERIOD
PSMPROG	DS	CL8	PROGRAM NAME ON EXEC CARD
PSMRECOV	DS	CL8	TYPE OF RECOVERY
PSMUNITN	DS	CL8	UNIT NAME
PSMOPER	DS	C	OPER SUPPLIED VOLUME FOR SWITC
PSMJCLAS	DS	C	JOB CLASS
PSMTMDSN	DS	C	TEMP DATASET FLAG (Y=YES)
PSMVOLSQ	DS	H	VOLUME SEQUENCE
PSMNVOL	DS	F	NUMBER OF VOLUMES
PSMPSPC	DS	F	PRIMARY SPACE ALLOCATION
PSMSSPC	DS	F	SECONDARY SPACE ALLOCATION
PSMJST	DS	F	JOB START TIME
PSMJSD	DS	F	JOB START DATE
PSMNUNIT	DS	F	NUMBER OF UNITS
PSMNQUAL	DS	F	NUMBER OF DSNAME QUALIFIERS
PSMHLQ	DS	0CL8	HLQ OF DSNAME
PSMQUAL1	DS	CL8	1ST QUALIFIER
PSMQUAL2	DS	CL8	2ND QUALIFIER
PSMQUAL3	DS	CL8	3RD QUALIFIER
PSMQUAL4	DS	CL8	4TH QUALIFIER
PSMQUAL5	DS	CL8	5TH QUALIFIER
PSMQUAL6	DS	CL8	6TH QUALIFIER
PSMQUAL7	DS	CL8	7TH QUALIFIER
PSMQUAL8	DS	CL8	8TH QUALIFIER
PSMQUALL	DS	CL8	LAST QUALIFIER
PSMQUALS	EQU	*-PSMHLQ	LENGTH OF QUALS
PSMSTEP	DS	CL8	STEP NAME
PSMPROCS	DS	CL8	PROC STEP NAME
PSMCDCT	DS	XL60	DCT FOR CURRENT UCB
PSMMSGB	DS	CL(PSMMSGBLEN)	DEFAULT BUFFER FOR MSGED
*			
* Note: Actual message text begins at PSMMSGB+8. The MSGPREF			
* global is not included in the message ID (XXNnnna).			
*			
PSMTEXT	DS	F	TOTAL NUMBER OF EXTENTS
PSMPOOL	DS	CL8	EASY/POOL POOL NAME FOR VOLUME
PSMREFV	DS	CL6	EASY/POOL REF VOLID
PSMLIB	DS	C	DSNTYPE=LIBRARY (PDSE)
PSMIAM	DS	C	IAM DATASET (Y/N)
PSMSIZE	DS	F	PRIMARY SPACE IN KB
PSMSIZES	DS	F	SECONDARY SPACE IN KB
PSMMSIZE	DS	F	1 PRI SPACE + 15 SECS IN KB
PSMCSPC	DS	F	CURRENT SPACE USED + NEW EXTEN
PSMLRECL	DS	F	DSN LRECL
PSMBLOCK	DS	F	DSN BLOCKSIZE
PSMRECFM	DS	CL3	RECORD FORMAT
PSMSTRIP	DS	C	STRIPED DATASET (Y/N)
PSMSTRTY	DS	CL2	STRIPED DATASET TYPE
PSMSTRCT	DS	F	STRIPED DATASET STRIPE COUNT
PSMHIER	DS	C	HIERARCHICAL DATASET (Y/N)
PSMROUND	DS	C	'Y' - ROUND CODED IN SPACE PAR
PSMSPEC	DS	C	'Y' - SPECIFIC ALLOCATION
PSMSMS13	DS	C	'Y' - DFSMS 1.3 OR HIGHER
*			
* MISC FIELDS			
*			
PSMDVOL	DS	CL6	VOLUME TO SWITCH TO
PSMRNVOL	DS	H	NO. OF ORIGINAL VOLS (RECAT)
PSMRVOL	DS	CL6	ORIGINAL VOLUME NAME (RECAT)
PSMORGK	DS	F	ORIGINAL PSW KEY ON ENTRY

Figure 13-1 SMF Record Layout (part 5 of 5)

PSMADDV	DS	CL1	NEED TO ADD VOLUME TO JFCB
PSMAVGR	DS	CL1	AVGREC KEYWORD IN EFFECT
PSMVCLST	DS	CL44	VSAM CLUSTER DSNAME
PSMVCOMP	DS	CL5	VSAM COMP TYPE (DATA OR INDEX)
PSMVTYP	DS	CL10	TYPE OF VSAM DATA SET
PSMVDTYP	DS	CL4	TYPE OF VSAM CLUSTER
PSMDYNAL	DS	C	C'Y' - DYNAMIC ALLOCATION FLAG
PSMEFLAG	DS	C	C'Y'-MESSAGE IS ALREADY EDITED

*
* Note: Proprietary fields follow. The length of the PSMPDCS
* record is 2112 bytes. (PFMSMFE equates to x'840'.)
*
PSMSMFE EQU (*-PSMPDCS),(*-PSMPDCS) END OF SMF RECORD

Chapter 14 Generating and Managing Batch Reports

This chapter provides instructions for generating and managing batch reports in MAINVIEW SRM. The following information is included:

Overview	14-1
Initiating Report JCL Generation	14-3
Setting Up the MAINVIEW Batch Environment	14-5
Generating MAINVIEW Batch Reports JCL	14-7
Managing MAINVIEW Batch Report JCL Members	14-9
Batch Report Output Members	14-10

Overview

You can print the information on any MAINVIEW SRM view by submitting a batch job. You can use an ISPF dialog panel to assist you in generating the JCL to produce MAINVIEW batch reports of your historical data.

This chapter contains the information you need to

- set up the MAINVIEW batch environment
- generate the MAINVIEW batch report JCL
- manage the MAINVIEW batch report JCL members

You may want to submit a job each day to report on some of the key elements of performance for the previous day. The **TIME** command has several parameters that enable you to specify timeframes relative to today. These parameters allow you submit the same job on a periodic basis without having to change the JCL (for example, specify **TIME yday 15:00 4h** to produce a report for every four hours starting at 3 pm yesterday).

Using this feature, you can enter the command in an ISPF dialog panel much as you would if you were in an online session. The tabular or detail report is directed to a data set or SYSOUT. The report output is nearly the same format as the online tabular and detailed displays; however, it displays all rows from the query and shows as many columns as your data set allows.

Initiating Report JCL Generation

The ISPF dialog panel helps generate the JCL for reports you want to run periodically, and it keeps track of previously generated report JCL members. Each report can have a maximum of 16 queries.

To set up the JCL for reports perform the following steps:

Step 1 Navigate to the menu from which the view is selected:

EZSRM	MAINVIEW SRM common
EZSRMHSM	EasyHSM
EZSRMSGC	SG-Control
EZSRMSGD	StorageGUARD historical space
EZSRMSGP	StorageGUARD historical performance

Step 2 Select the **Batch Reports** option for the chosen product menu; for example, **MVSRMSGD Batch Reports** on the MVSRMSGD Menu.

Tip: If you select the **Batch Reports** option from a menu other than the designated product menu, you can include a **CONTEXT** command in the report specifying the product name; for example, MVSRMSGD for StorageGUARD historical space.

The MAINVIEW Batch Reports panel is displayed, as shown in Figure 14-1.

Figure 14-1 MAINVIEW Batch Reports Panel

```

----- MAINVIEW Batch Reports -----
Option ==>

0 Setup          Set up MAINVIEW Batch Environment
1 Generate       Generate MAINVIEW Batch Reports JCL
2 Edit/Submit    Edit/Submit existing MAINVIEW Batch Reports JCL
3 Browse         Browse MAINVIEW Reports

X Exit          Terminate

                (C) Copyright 2000, BMC Software, Inc.
    
```

The MAINVIEW Batch Reports panel provides the following options:

Setup	displays the MAINVIEW Batch Environment Setup panel on which job and report information is recorded. This information is unlikely to change much
Generate	displays the Generate MAINVIEW Batch Reports JCL panel on which information that is specific to each report is recorded, saved, and submitted to generate the JCL
Edit/Submit	displays the MAINVIEW Batch JCL Member List panel, which lists the members that contain stored The stored members can be browsed, deleted, edited, and resubmitted.
Browse	displays the MAINVIEW Batch Reports list, which lists the batch reports that are stored The list shows that the report is either stored in a sequential data set or in a member of a partitioned data set (PDS). The stored reports can be browsed

Setting Up the MAINVIEW Batch Environment

You should to set up the MAINVIEW batch environment for your reports. After you select the setup information, little change will be made from one report to another.

To set up the MAINVIEW batch environment, perform the following steps:

Step 1 On the MAINVIEW Batch Reports panel, on the **Option** line, type **0**.

The MAINVIEW Batch Environment Setup panel is displayed, as shown in Figure 14-2.

Figure 14-2 MAINVIEW Batch Environment Setup Panel

```

----- MAINVIEW Batch Environment Setup -----
Command ==>

Job Statement Information:
==> //USERID   JOB  (ACCOUNT), 'NAME'
==> //*
==> //*
==> //*
==> //*

Report Title ==>

MAINVIEW Clist Library..... ==> hilevel.BBCLIB
MAINVIEW Clist Name..... ==> MAINVIEW
MAINVIEW BBLINK Library.... ==> hilevel.BBLINK
CAS SSID..... ==> BBSC

Library to save JCL..... ==> 'userid.export.cntl'
Temporary Workfile Unit.... ==> VIO

Press END to save changes and return to the previous panel
Type CANCEL to return to the previous panel without saving changes

```

This panel provides the options for selecting the job and report information you want for the JCL to generate your report.

Step 2 Under **Job Statement Information**, type a job card statement conforms to your installation standards.

Step 3 In the **Report Title** field, type the title you want printed at the top of your reports.

- Step 4** In the **MAINVIEW Clist Library** field, type the name of the library containing the MAINVIEW CLIST.
- Step 5** In the **MAINVIEW Clist Name** field, type the name of the MAINVIEW CLIST.
- Step 6** In the **MAINVIEW BBLINK Library** field, type the name of the MAINVIEW BBLINK library.
- Step 7** In the **CAS SSID** field, type the four-character CAS subsystem ID.
- Step 8** In the **Library to save JCL** field, type an ISPF library name for the saved JCL.
- Step 9** In the **Temporary Workfile Unit** field, type a unit name for the temporary data sets.
- Step 10** Press **End** to save your changes and return to the MAINVIEW Batch Environment Setup panel.

To return to the MAINVIEW Batch Environment Setup panel without making or saving changes, type **CANCEL** on the **COMMAND** line.

Generating MAINVIEW Batch Reports JCL

Before generating your report JCL, some specific information is necessary. To add the necessary information perform the following steps:

Step 1 On the MAINVIEW Batch Reports panel, on the **Option** line, type **1**.

The Generate MAINVIEW Batch Reports JCL panel is displayed, as shown in Figure 14-3.

Figure 14-3 Generate MAINVIEW Batch Reports JCL Panel

```

----- Generate MAINVIEW Batch Reports JCL -----
Command ==>

JCL Member name      ==> JOBNAME  Replace (Y/N)?  YES
JCL Member Description ==> Description  <====
Report format:      ==> ASIS      (ASIS or CSV)
Lines/Page:        ==> 60        (ASIS format only)

Sysout Class ==>
Or
Output data set ==> EXPORT.LST
                   Volume ==>      (If data set uncatalogued)

Enter Queries on the lines below.  Each line will write a separate
report to the output data set.

Press END to save changes and generate the JCL
Type CANCEL to return to the previous panel without saving changes

More:      +

==> TIME LASTWEEKDAY 23:59 1D;JSRM
==> TIME *-7;JCPU
==> MCHAM
==> SYSOVER
==>
    
```

The Generate MAINVIEW Batch Reports JCL panel displays fields for recording the information and the queries necessary for your JCL.

Step 2 Type the input information, output information, and queries you need for your report in the following fields.

JCL Member name	Name for this JCL member. Each JCL report can be given a member name and a description.
JCL Member Description	Description of the report. This value is optional.
Report format	ASIS or CSV format. Enter ASIS to print reports that look like the screen. Enter CSV for comma-separated fields to download the record to a spreadsheet program.

Lines/Page	for ASIS reports, the heading will be printed on each page For a continuous report with the heading on the first page only, enter 0.
Sysout Class	Enter the report SYSOUT class.
Output data set	Enter the name of a sequential data set or partitioned data set with a member name for the report output If the data set is not catalogued, supply the VOLSER. Entering a SYSOUT class overrides the data set specification.
Queries	<p>Enter the kinds of data that you want to see. Enter queries exactly as you would on the COMMAND line in an online session. For example:</p> <ul style="list-style-type: none"> • MCHAN prints the MCHAN view. • MCHAN;FORM JSRM prints MCHAN with the JSRM FORM. • MCHAN on the first line followed by FORM JSRM on the second line prints the MCHAN report first followed by the MCHAN report under the JSRM FORM. <p>In some views, you can establish a timeframe and duration that is different from the current time. You can combine that different timeframe and duration with the first command. For example:</p> <p>TIME LASTWEEKDAY 23:59 1D;SPPOOL</p> <p>Note: This time period remains in effect until it is changed by a subsequent query.</p> <p>Through the use of one-per-line commands, additional reports of data can be created from the same timeframe.</p>

Tip: The generated JCL executes the MAINVIEW CLIST. The CAS and PAS must be started *before* the JCL is executed.

Step 3 Press the **End** key to save changes and generate the JCL.

Step 4 The JCL is presented in an edit session. You probably do not need to make changes.

Step 5 To submit the job, on the **COMMAND** line, type **SUB**, and press **Enter**.

To cancel and return to the MAINVIEW Batch Reports panel without saving changes, type **CANCEL**.

Managing MAINVIEW Batch Report JCL Members

The Edit/Submit option in the MAINVIEW Batch Reports panel opens the JCL member list of generated records. This list provides options to browse, delete, edit, and submit the JCL.

» To display the MAINVIEW Batch JCL Member List, on the MAINVIEW Batch Reports panel **Option** line, type **2**.

The MAINVIEW Batch JCL Member List panel is displayed with a table of stored JCL members, as shown in Figure 14-4.

Figure 14-4 MAINVIEW Batch JCL Member List Panel

```

----- MAINVIEW Batch JCL Member List ----- Row 1 to 6 of 6
Command ==>>                               Scroll ==>> PAGE

Line Commands: B - Browse JCL   DEL - Delete JCL   E - Edit JCL
                SUB - Submit JCL

Press END to return to main menu

LC Member      Description                               Date      Time
-----
  @MCHAN      Channel Activity Report          13:10:25   2001/02/22
  @WMDEV      Weekly Device Report              09:06:46   2000/09/22
  @DMDEV      Daily Device Report               08:45:39   2000/09/18
***** Bottom of data *****
  
```

The table is two panels wide. The directional arrows (>>> or <<<), above the list and on the right, indicate that additional JCL member information is available. Use the RIGHT (usually assigned to **F10**) key to see information on the right and use the LEFT (usually assigned to **F11**) key to return to the information on the left.

To edit, browse, delete, and submit the JCL in this member list, type

- **E** to edit a member
- **B** to browse the JCL
- **DEL** to delete a member
- **SUB** to submit the job

Warning! When a specific JCL member is deleted, the corresponding report data set or member is also deleted.

Batch Report Output Members

The Browse option in the MAINVIEW Batch Reports panel opens the MAINVIEW Batch Report List panel. This panel provides a list of generated online batch members, which can be browsed.

»To display the MAINVIEW Batch Report List panel, on the MAINVIEW Batch Reports panel **Option** line, type **3**.

The MAINVIEW Batch Report List panel is displayed, as shown in Figure 14-5.

Figure 14-5 MAINVIEW Batch Report List Panel

```

----- MAINVIEW Batch Report List ----- Row 1 of 20
Command ==>>                               Scroll ==>> PAGE

  S  Select member to view report online

Press END to return to MAINVIEW Batch Reports Menu

LC  Member      Reports Data Set                               UserID
-----
BPA0583A  BMVDID.BPA0583.JCL(BPA0583L)                BMVDID3
BPA0583B  BMVDID.BPA0583.JCL(BPA0583M)                BMVDID3
CACHE     BMVJOJ.TESTPO.LST(CACHE)                    BMVJOJ2
COUPLING  BMVJOJ.TESTPO.LST(COUPLING)                BMVJOJ2
DEVICES   BBSECH8.JCL.CNTL(DEVZ)                      BBSECH8
JDELAYS   BMVJOJ.TESTPO.LST(DELAYS)                  BMVJOJ2
JONJMV    BMVJOJ.ZYYXXX.LST(JONJMV)                  BMVJOJ2
JOVER     BMVJOJ.COOL                                BMVJOJ2
LOST      BMVJOJ.TESTPO.LST(A9)                      BMVJOJ2
MVBATCH   SYSOUT(*)                                    BBGST09
MVBATCH1  BMVJOJ.A3DLIB(MVBATCH1)                    BMVJOJ2
MCHAN     BSLARD.X                                    BSLARD1
MYBATCH   SYSOUT(R)                                    BMVJOJ2
TERRY1    SYSOUT(A)                                    BBGST06
TESTJN    BMVJOJ.TESTPO.LST(TESTJ2)                  BBGST12
TEST2     BBGST05.PRNT.TEST(DATA2)                  BBGST05
TST1     BMVJOJ.TDSTPO.LST                          BMVJOJ2
    
```

To browse a report, in the LC column on the left side of the report member, type S.

In Figure 14-5, the MCHAN member of the BSLARD.X reports data set is selected.

The MAINVIEW Batch Report for MCHAN is displayed, as shown in Figure 14-6.

Figure 14-6 MAINVIEW Batch Report, MCHAN

```

Display  Filter  View  Print  Options  Help
-----
SDSF OUTPUT DISPLAY JIMA@BR  JOB23052  DSID   106 LINE 0          COLUMNS 02- 133
COMMAND INPUT ==>                               SCROLL ==> CSR
*****
***** TOP OF DATA *****
              MV SRM BATCH REPORT
CMD> MCHAN

                                REPORT PAGE   1
11APR2001  09:29:28              MainView Batch Report          PAGE   1
MCHAN=====SJSG=====*****11APR2001==09:29:28====MVS RM====D===59

Ch SSI      Channel  Share  Part  Total  Total Busy
-- Target  Type      Ind    Busy  Busy  0{50}100
   SJSG     OP       Y      0.00  0.00
  5 SJSG     CN D      Y      0.00  0.00
  6 SJSG     CN D      Y      0.00  0.00
  9 SJSG     CN D      Y      0.00  0.00
  A SJSG     CN D      Y      0.00  0.00
  D SJSG     CN D      Y      0.00  0.00
  E SJSG     CN D      Y      0.00  0.00
  F SJSG     CN D      Y      0.00  0.00
 16 SJSG     CN D      Y      0.00  0.00
 19 SJSG     CN D      Y      0.00  0.00
 1A SJSG     CN D      Y      0.00  0.00
 1B SJSG     CN D      Y      0.00  0.00
    
```

Appendix A Altering Messages with the Message Exit

Some messages can be modified at execution time by using the MAINVIEW SRM message exit. The message exit can perform the following actions:

- suppress the message completely
- change the message text, including lengthening the message
- specify or alter the MCS flags
- specify or alter the descriptor codes
- specify or alter the routing codes

On entry to the message exit, R1 contains the address of a two-word parameter list:

- +0 is the address of the WTO work area (described below).
- +4 is the address of the MAINVIEW SRM SMUV (included as member @SMUV *Installation and Definition Reference* on the installation data set ?*prefix*.BBSAMP).

```
LEN          DC  Y(0)  LENGTH OF WTO WORK AREA.
MCSFLAGS    DC  XL2'00' MCS FLAGS.
TEXT        DC  0C   START OF TEXT.
```

*

```
*THE FOLLOWING TWO FIELDS ARE NOT INCLUDED IN THE LENGTH INDICATED
*IN THE FIELD LEN. WHEN LENGTHENING THE MESSAGE, MAKE SURE YOU
*SAVE THE FOLLOWING TWO FIELDS. AFTER LENGTHENING, RESTORE THEM TO
*THE END OF THE MESSAGE TEXT.
```

*

```
DESCODE     DC  XL2'00' DESCRIPTOR CODES.
ROUTCODE    DC  XL2'00' ROUTING CODES.
```

Note that this work area is not a fixed-length control block. Its length can be determined by using the LEN field. The maximum length of a full-length message, including all supporting fields, is 256 bytes. If the message is lengthened, the trailing routing and descriptor codes must be moved as well.

Changes to the routing code are ORed with ROUTCDE=15.

Your user-written exit must use standard OS/390 linkages conventions. Registers at entry are

R1	address of two-word parameter list
R13	address of an 18-word register save area
R14	return address back to MAINVIEW SRM message processor
R15	entry point address of SMMSGEXT

The MAINVIEW SRM message processor uses the return code in register 15 on return from SMMSGEXT to determine the disposition of the message:

R15 = 0	issue message and log to SMF
R15 < 8	do not issue message, but log to SMF
R15 > or = 8	do not issue message or log to SMF

SMMSGEXT must be reentrant and capable of executing in 31-bit mode. The following linkage editor instructions are recommended:

```
INCLUDE OBJECT(SMMSGEXT)
ENTRY SMMSGEXT
MODE AMODE(31),RMODE(ANY)
SETCODE AC(1)
NAME SMMSGEXT(R)
```

A sample message exit is provided in member SMMSGEXT in the installation library *?prefix.BBSAMP*.

Appendix B Global Parameters

This section contains quick reference lists of global parameters used by MAINVIEW SRM.

SMMSYS xx - Master System Member Parameters	B-1
SMPOOL xx - SMS Subpool Member Parameters	B-73
SMSPOL xx - SMS Subpool Member Parameters	B-73
SMCAL Sxx - Calendar Member Parameters	B-74
SMVAR Sxx - Variable Member Parameters	B-76
SMFUNC xx - Function Member Parameters	B-77
SMDIAG xx - Diagnostic Member Parameters	B-81
SMEVNT xx - Event Member Parameters	B-82
SMVSCF xx - VTOC Scan Facility Parameters	B-86

Master System Member Parameters

SMMSYS xx SMMSYS xx contains the master system parameters for the MAINVIEW SRM system. Default parameter values can be specified that apply to the overall operating environment.

Subordinate Members

SMPOOL xx , SMSPOL xx , SMCAL Sxx , SMVAR Sxx , SMDIAG xx , SMFUNC xx , SMCRI Txx , SMEVNT xx , SMVSCF xx

Parameter Quick Reference

The following table provides a brief description of SET statement master system parameters. Detailed descriptions of each parameter are listed in alphabetical order after the table. The page is cross-referenced in the page number column.

Table 2-1 SET Statement System Parameters (Part 1 of 12)

Applies to										Page #	Description
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD		
AOO_SUBSYS=xxxx	N	X								B-15	Specifies the AutoOperator subsystems that are to receive events
AUTOPROC=xxxxxxxx	N					X				B-15	Specifies the name of the cataloged procedure used to start SG-Auto
BBI3_SSID=xxxx	Y	X								B-15	Specifies the CAS subsystem name to which the SVOS PAS should connect
BCDSn=xxxxxxxxxxx	N				X					B-16	Specifies HSM CDS database files to be used by MAINVIEW SRM
BLKINPUT=YES/NO	N		X							B-16	Changes block size for input data sets
BLKOLDSR=YES/NO	N		X							B-16	Changes blocksize for output data sets opened with disposition of old or shared
CAL=xx	Y		X	X						B-16	Suffix of parameter member SMCALSxx
CHECK=FIRST/ALLVOLS	N		X							B-17	Specifies whether to check all volumes the job requests during allocation or only the first volume requested
CRITLIST=xx	N		X	X						B-17	Specifies the suffix of an SMCRTxx parameter member.
DADSMEX=YES/NO	N							X		B-17	Determines if the DADSM preprocessing exit (IGGP00) is called
DASDGENR=(xxxxxxxx,...)	N		X							B-18	Specifies DASD generic names (1–8 characters) to be processed
DATEFMT=MMDD/DDMM	N			X						B-18	Date format
DCTYPE=(xxxxx,...)	N							X		B-19	Allows choice of one or more device characteristics to be maintained during volume switching (CACHE, SHARED, DUALCOPY, FASTWRITE)
DFREORGPRC=xxxxxxxx	N							X		B-19	Defines default SPACVOLA reorganize procedure name
DIAG=nn	N	X								B-19	Suffix of parameter member SMDIAGxx
DIAGMSDD=xxxxxxxx	N	X								B-20	Established WTO message tracing
DISPLAY=ALL/LIC	N	X								B-20	Display functions list in the ISPF interface

Table 2-1 SET Statement System Parameters (Part 2 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
DMYUNIT=(xxxxxxxxx, zzzzzzzz,....)	N		X							B-20	Unit name conversion (1–8 characters for each unit)
DP_RENAME=YES/NO	N		X							B-21	Determines if DASDPOOL is processed for volume selection during DADSM RENAME
DUMPDD=xxxxxxxx	N	X								B-21	Provides SYS1.DUMPx dump if MAINVIEW SRM abends
EVNT=xx	Y	X								B-21	Specifies the suffix you assign to the name of the SMEVNTxx event definition member
FDRIAM=YES/NO	N						X			B-22	For IAM customers only, FDRIAM=YES determines whether a data set is an IAM data set
FUNC=xx	Y	X								B-23	Suffix of parameter member SMFUNCxx
HISTDAYS=nn	N		X							B-23	Number of days (0-14) specified for gathering historical performance data
HLOGAUTH=nn	Y				X					B-24	Automatic DFHSM log switch interval in hours
HLOGAUTM=nn	N				X					B-24	Automatic DFHSM log switch interval in minutes
HLOGCOLL=YES/NO	N				X					B-25	Activation of EasyHSM logfile data collection
HLOGINDX=xxxxxxxx	N				X					B-25	DSN prefix of EasyHSM log extract file
HLOGPRIM=nnn	N				X					B-25	Size of primary allocation of log extract file
HLOGTASK=xxxxxxxx	N				X					B-26	Name of proc to run on DFHSM logfile swap
HLOGUNIT=xxxxxxxx	N				X					B-26	Unit name for allocation of log extract file
HLOGYDSN=xxxxxxxx	N				X					B-26	Data set name of DFHSM logfile Y
HSMACTID=xxxxxxxx	N				X					B-26	High-level name of DFHSM activity data sets
IGNOREDD=xxxxxx	N	X								B-27	Suppresses MAINVIEW SRM activity for jobstep
JCLEXT=YES/NO	N		X							B-27	Provides volume and unit information after accessing the catalog

Table 2-1 SET Statement System Parameters (Part 3 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
JCLUREQ=YES/NO	N		X							B-28	Determines whether the UNIT information is required in the JCL
MAXVOL=nn	N							X		B-28	Limits number of volumes a data set is allowed to use
MCDSn=xxxxxxxxxxx	N				X					B-29	Specifies HSM migrated data set file allocated during EasyHSM startup
MODTRCDD=xxxxxxxx	N	X								B-29	Sets module entry/exit tracing
MREDUCE=YES/NO	N							X		B-29	Determines if secondary space reduction can occur on multivolume data sets allocated by JCL
MSGID=YES/NO	N	X								B-30	Specifies the inclusion of the MAINVIEW SRM message identifier in the message text
MSGLVL=I/W/E/S	N	X								B-30	Level of messages to be generated
MSGPREF=xxx/SVM	N		X	X	X			X		B-30	MAINVIEW SRM message identifier prefix
NOCATDYN=YES/NO	N							X		B-31	Allows NOCATLG2 to process dynamically allocated data sets
NOCATPFX=xxx	N							X		B-31	Second-level qualifier to be used when renaming a data set during NOCATLG2 processing
NOCATPRG=YES/NO	N							X		B-31	Allows data sets to be scratched before the expiration date during NOCATLG2 processing
NOCATSEC=xxxxxx	N							X		B-32	Level of security performed before scratching or renaming a data set during NOCATLG2 processing (NONE, CREATE, READ, UPDATE, ALTER)
NOCATSMS=YES/NO	N							X		B-32	Allows SMS-managed data sets to be renamed, uncataloged, or scratched during NOCATLG2 processing
NOCATVOL=SAME/DIFF	N							X		B-33	Allows a new data set to be allocated to the same volume it was previously cataloged on during NOCATLG2 processing
NOCATWHEN=ALLOC/TERM	N							X		B-33	Specifies when NOCATLG2 processing is to occur for non-SMS data sets (allocation or step termination)

Table 2-1 SET Statement System Parameters (Part 4 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
OCDS=xxxxxxxxxxx	N				X					B-34	Specifies HSM OCDS data set to be defined and allocated during EasyHSM startup
OPMHLQ=xxxxxxxx	N				X					B-34	MAINVIEW SRM output data set high-level qualifier
ORIGDATA=PRO/POOL	N		X							B-34	Specifies whether VOL and UNIT contain the original volser and unit values from the JCL or contain the current value.
PASSWORD=xxxxxxxx	Y	X								B-35	Specifies a MAINVIEW SRM password
PERFRM_PRC==xxxxxxxx	N							X		B-35	Specifies the name of the procedure used to start the historical performance data collector
POOL=xx	Y	X								B-35	Suffix of parameter member SMPOOLxx
PROCOLD=YES/NO	N		X							B-36	Allows interception of DD statements that specify OLD allocations
REJECT=FIRST/LAST	N		X							B-36	Controls termination of processing of rejected data sets
REQTYPE=YES/NO	N							X		B-36	Specifies if the MNTYPE statement is considered the request type instead of the mount type
SCAT=STEPEND/IMMEDIATE	N							X		B-37	Forces immediate catalog update during volume switch
SG_INITPOOL=nnnnn	N							X		B-37	Specifies the maximum number of defined pools included in a single snapshot.
SG_INITVOL=nnnnn	N							X		B-37	Specifies the maximum number of defined volumes included in a single snapshot.
SG_IXFPNTVL=nn	N							X		B-38	Specifies the number of hours between refreshes of the IXFP data tables
SG_MAXACCT=nnnnn	N							X		B-38	Specifies the maximum number of active accounts in the SG-Control database
SG_MAXPOOL=n	N							X		B-38	Specifies the number of pools that can be assigned to a volume

Table 2-1 SET Statement System Parameters (Part 5 of 12)

Applies to										Page #	Description
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD		
SG_MAXSSDSZ= <i>nnnnn</i>	N								X	B-39	Specifies the maximum number of cylinders used for a solid state disk drive
SG_READNTVL= <i>nnnn</i>	N								X	B-39	Specifies the frequency at which StorageGUARD creates a snapshot in core
SG_RETRYLIM= <i>nnnn</i>	N								X	B-39	Specifies the number of abend conditions that the data collector should ignore
SG_SIBSTK= <i>nn</i>	N								X	B-39	Specifies the IXFP SIBBATCH parameter member to be used by the MAINVIEW SRM IXFP services for communications with the IXFP address space
SG_SPACHLDR= <i>mask</i>	N								X	B-40	Defines a data set name mask that StorageGUARD can use to identify space holder data sets
SG_SUBTASKS= <i>nn</i>	N								X	B-40	Specifies the number of volumes that can be read in parallel
SG_WRITNTVL= <i>nnnn</i>	N								X	B-40	Specifies the frequency at which snapshots are written to the StorageGUARD database
SGA_ENQSCOP= <i>GLOBAL/LOCAL</i>	Y					X				B-41	Specifies the operational environment in which SG-Auto is to run
SGACMD= <i>nn</i>	N					X				B-41	Specifies the two position suffix of the initial command for executing the SG-Auto started task
SGASCAN= <i>YES/NO</i>	N					X				B-41	Specifies whether SG-Auto should be started in SCAN mode
SGASIM= <i>YES/NO</i>	N					X				B-41	Specifies whether SG-Auto should be started in SIMULATION mode
SGC_ADDEXIT= <i>xxxxxxxx</i>	N						X			B-42	Specifies the name of the SG-Control Add Exit
SGC_CHKEXIT= <i>xxxxxxxx</i>	N						X			B-42	Specifies the name of the SG-Control Check Exit
SGC_DEFEXIT= <i>xxxxxxxx</i>	N						X			B-42	Specifies the name of the SG-Control Default Exit
SGC_KEYEXIT= <i>xxxxxxxx</i>	N						X			B-42	Specifies the name of the SG-Control Account Code Build Exit

Table 2-1 SET Statement System Parameters (Part 6 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
SGC_SECEXIT=xxxxxxxx	N						X			B-43	Specifies the name of the SG-Control Security Exit
SGC_SELEXIT=xxxxxxxx	N						X			B-43	Specifies the name of the SG-Control Select Exit
SGC_STOGRP=YES/NO	N						X			B-43	Specifies whether or not to retrieve SMS storage group information; this parameter should be set to YES only if SMS storage group information is required in FLST or RLST processing
SGC_STORCLS=YES/NO	N						X			B-43	Specifies whether or not to retrieve storage class information; this parameter should be set to YES only if SMS storage class information is required in FLST or RLST processing
SGCDSN=xxxxxx...xxxxxx	N						X			B-44	Specifies the data set name for the dynamic allocation/deallocation of SG-Control database DD, namely SGCDB
SGD_PROCNM=SGDCOLLS	N								X	B-44	Specifies the name of the data collector started task (1-8 characters)
SGD_SMFID=nnn	N								X	B-44	Specifies an SMF record number for MAINVIEW SRM audit records written to the SMF data set for StorageGUARD
SGDCOLLECT=YES/NO	N								X	B-44	Indicates what the default is for StorageGUARD pool collection; there is a corresponding parameter at the POOL level to override the default
SGDCOLLECT n =YES/NO	N								X	B-44	Specifies whether a pool is processed by the StorageGUARD alternate data collector assigned a suffix of n where n may be a number in the range of 1-8.
SGDPROCNM n =xxxxxxxx	N								X	B-45	Specifies the cataloged procedure to be started for a specified copy of StorageGUARD
SGDSMFID n =	N								X	B-45	Specifies an SMF record number for MAINVIEW SRM audit records written to the SMF data set for StorageGUARD for a specified copy of StorageGUARD
SGEXITACCT n =	N								X	B-46	Specifies the name of the account user exit routine for a specified copy of StorageGUARD

Table 2-1 SET Statement System Parameters (Part 7 of 12)

Applies to										Page #	Description
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD		
SGEXITPOOLn=	N								X	B-46	Specifies the name of the pool user exit routine for a specified copy of StorageGUARD
SGEXITVOLn=	N								X	B-46	Specifies the name of the volume user exit routine for a specified copy of StorageGUARD
SGINITPOOLn=	N								X	B-47	Specifies the maximum number of defined volumes included in a single snapshot for a specified copy of StorageGUARD
SGINITVOLn=	N								X	B-47	Specifies the number of hours between refreshes of the IXFP data tables for a specified copy of StorageGUARD
SGMAXACCTn=	N								X	B-48	Specifies the maximum number of active accounts in the SG-Control database for a specified copy of StorageGUARD
SGMAXPOOLn=n	N								X	B-48	Specifies the number of pools that can be assigned to a volume for a specified copy of StorageGUARD
SGMAXSSDSZn=nnnnn	N								X	B-48	Specifies the maximum number of cylinders used for a solid state disk drive for a specified copy of StorageGUARD for a specified copy of StorageGUARD
SGP_EXITBBS=nn	N								X	B-49	Specifies the number of megabytes to allocate in a scope common data space for the StorageGUARD performance exit buffer block
SGP_EXITLIB=xxxxxx	N								X	B-49	Specifies the default library where the StorageGUARD performance collector SMF exits reside
SGP_MAXCCUS=nnnn	N								X	B-49	Identifies the maximum number of control units that are in use during an interval.
SGP_MAXDIRS=nnnn	N								X	B-50	Identifies the maximum number of directors that are in use during an interval.
SGP_MAXDSNS=nnnn	N								X	B-50	Identifies the maximum number of data set names that are in use during an interval.

Table 2-1 SET Statement System Parameters (Part 8 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
SGP_MAXJOBS= <i>nnnn</i>	N								X	B-50	Identifies the maximum number of jobs (batch, TSO, and started tasks) that are in use during an interval.
SGP_MAXLCUS= <i>nnnn</i>	N								X	B-50	Defines the maximum number of logical control unit/CHIP combinations in used during an interval.
SGP_MAXPOLs= <i>nnnn</i>	N								X	B-51	Identifies the maximum number of pools that are in use during an interval.
SGP_MAXPTHS= <i>nnnn</i>	N								X	B-51	Identifies the maximum number of CHPIDs that are in use during an interval.
SGP_MAXPVLS= <i>nnnn</i>	N								X	B-51	Identifies the maximum number of physical volumes that are in use during an interval.
SGP_MAXRRKS= <i>nnnn</i>									X	B-51	Identifies the maximum number of RAID ranks that are in use during an interval.
SGP_MAXRSFS= <i>nnnn</i>	N								X	B-52	Identifies the maximum number of RVA frames that are in use during an interval.
SGP_MAXSCLS= <i>nnnn</i>	N								X	B-52	Identifies the maximum number of storage classes that are in use during an interval.
SGP_MAXVOLS= <i>nnnn</i>	N								X	B-52	Identifies the total number of online DASD volumes on the OS/390 image being monitored.
SGP_RDFCOMP= <i>YES/NO</i>	N								X	B-52	Specifies whether data compression is in effect for records being written to the StorageGUARD performance resource data files
SGP_SIBSTK= <i>xxxxxxxx</i>	N								X	B-53	Identifies the IXFP SIBBATCH parameter member to be used by the MAINVIEW SRM IXFP services for communications with the IXFP address space
SGP_SMF42= <i>YES/NO</i>	N								X	B-53	Specifies whether or not the SMF 42 record is to be written to the SMF data set
SGP_TRACE= <i>xxxxxxx</i>	N								X	B-53	Specifies the trace default for the StorageGUARD Performance collector. The default is NOTRACE

Table 2-1 SET Statement System Parameters (Part 9 of 12)

Applies to										Page #	Description
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD		
SGPROCACCT <i>n=xxxxxxx</i>	N								X	B-53	Specifies the name of the account REXX procedure. Do not specify this parameter unless the StorageGUARD Automation Facility is being used for a specified copy of StorageGUARD
SGPROCPOOL <i>n=xxxxxxx</i>	N								X	B-54	Specifies the name of the pool REXX procedure. Do not specify this parameter unless the StorageGUARD Automation Facility is being used for a specified copy of StorageGUARD
SGPROCVLER <i>n=xxxxxxx</i>	N								X	B-54	Specifies the name of a REXX procedure that will be invoked if the data collector encounters an I/O error while reading a VTOC for a specified copy of StorageGUARD
SGPROCVOL <i>n=xxxxxxx</i>	N								X	B-54	Specifies the name of the volume REXX procedure. Do not specify this parameter unless The StorageGUARD Automation Facility is being used for a specified copy of StorageGUARD
SGREADNTVL <i>n=nnnn</i>	N								X	B-55	Specifies the frequency at which StorageGUARD creates a snapshot in core for a specified copy of StorageGUARD
SGRETRYLIM <i>n=nnnn</i>	N								X	B-55	Specifies the number of abend conditions that the data collector should ignore for a specified copy of StorageGUARD for a specified copy of StorageGUARD
SGSPACHLDR <i>n=xxxxxxxxx</i>	N								X	B-55	Specifies the IXFP SIBBATCH parameter member to be used by the MAINVIEW SRM IXFP services for communications with the IXFP address space for a specified copy of StorageGUARD
SGSUBTASKS <i>n=nn</i>	N								X	B-56	Defines a data set name mask that StorageGUARD can use to identify space holder data sets for a specified copy of StorageGUARD
SGWRITNTVL <i>n=nnnn</i>	N								X	B-56	Specifies the number of volumes that can be read in parallel

Table 2-1 SET Statement System Parameters (Part 10 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
SIZEISPRIM= YES/NO	N		X					X		B-56	Determines if the SIZE filter/rule list parameter includes only the size of the primary extent or the size of the primary and one secondary extent
SKIP=(CHECK=(xxx,xxx,...),DD name=xxxxxxx, PROG=xxxxxxx)	N							X		B-57	Specifies checks to be bypassed during volume switching
SMFID=nnn	N	X								B-58	Record number for MAINVIEW SRM SMF records
SMS_ALLOC= YES/NO	N		X	X						B-58	Determines if SMSSELCT is processed for SMSPOOL during DADSM ALLOCATE
SMS_EXTEND= YES/NO	N		X	X						B-59	Determines if SMSSELCT is processed for SMSPOOL_EXT during DADSM EXTENDNV
SMSPOOL=xx	N		X	X						B-59	Specifies the suffix of an SMSPOLxx parameter member
STKSCR=(xxx,xxx,xxx,xxx)	N		X							B-59	STK silo support
SYSLIB=xxxxxxxxxxxxxxxx	N	Y								B-60	Specifies a default data set to be allocated at SVOS startup
SYSLIB2=xxxxxxxxxxxxxxxx	N	Y								B-60	Specifies a default data set to be allocated at SVOS startup
SYSLIB3=xxxxxxxxxxxxxxxx	N	Y								B-60	Specifies a default data set to be allocated at SVOS startup
TAPEGENR=(xxxxxxxx,....)	N		X							B-60	Specifies tape generic names (1–8 characters) to be processed
TRACEDD=xxxxxxx	N	X								B-61	Traces MAINVIEW SRM activity for jobstep
TRKCYL=nnnnn	Y		X					X		B-61	Default device tracks per cylinder
TRKLEN=nnnnnnn	Y		X					X		B-62	Default device bytes per track
USECAT= YES/NO	N			X						B-62	ACS selection criteria catalog name usage
VAR=xx	N	X								B-62	Suffix of parameter member SMVARSxx
VSAMJCL=CLUS/COMP	N									B-63	Controls level of processing of VSAM data sets

Table 2-1 SET Statement System Parameters (Part 11 of 12)

Applies to										Page #	Description
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD		
VSAMLIMWARN=xx	N							X		B-63	Specifies the percentage value to be used before issuing the 4 GB-limit message
VSAMPRIM=YES/NO	N							X		B-63	Use primary size for VSAM volume extensions
VSAMZSEC=YES/NO	N							X		B-64	Controls out-of-space recoveries for VSAM files with zero secondary space coded
VSCAN_MNTSK=nn	N								X	B-64	Specifies the minimum number of tasks (TCBs) used by the VTOC scan to perform the collection.
VSCAN_MXTSK=nn	N								X	B-64	Specifies the maximum number of tasks (TCBs) used by the VTOC scan to perform the collection.
VSCAN_OINDX=xxxxxxxxxx	Y								X	B-65	Specifies the prefix name of the VTOC scan collection data set.
VSCAN_OPRI=nnnn	N								X	B-65	Specifies the primary allocation size in cylinders for the VTOC scan collection data set.
VSCAN_OSEC=nnnn	N								X	B-65	Specifies the secondary allocation size in cylinders for the VTOC scan collection data set.
VSCAN_OUNIT=xxxxxxxx	N								X	B-66	Specifies the device type of the VTOC scan collection data set.
VSCAN_OVOL=xxxxxx	N								X	B-66	Specifies the volume serial number of the VTOC scan collection data set.
VSCAN_TPRI=nnnn	N								X	B-66	Specifies the primary allocation size in cylinders for the VTOC scan temporary data set.
VSCAN_TSEC=nnnn	N								X	B-66	Specifies the set secondary allocation size in cylinders for the VTOC scan temporary data.
VSCAN_TUNIT=xxxxxxxx	N								X	B-67	Specifies the device type for the VTOC scan temporary data set.
VSCAN_TVOL=xxxxxx									X	B-67	Specifies the volume serial number for the VTOC scan temporary data set.
WTODC=n	N	X								B-67	Specifies the message descriptor code(s)

Table 2-1 SET Statement System Parameters (Part 12 of 12)

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
WTORC= <i>nn</i>	N	X								B-68	Specifies routing codes assigned to message text (1-16)
X37POOL= <i>NEW/ORIG</i>	N							X		B-68	Specifies which volume is used by X37 to determine the pool name in EOVS processing

Table 2-2 INC/EXC Statement Parameter Quick Reference for SMMSYSxx

Applies to											
Parameter	Required	All	EasyPOOL	EasySMS	EasyHSM	SG-Auto	SG-Control	StopX37/II	StorageGUARD	Page #	Description
FORPLEXNAME=xxxxxxxx	N		X	X	X		X	X	X	B-22	Specifies one or more user-defined systems that can be included or excluded in a sysplex environment
FORSMFID=xxxxxxxx	N		X	X	X		X	X	X	B-22	Specifies SMF records that can be included or excluded in a sysplex environment
FORSYSID=xxxxxxxx	N		X	X	X		X	X	X	B-22	Specifies user-defined system IDs that can be included or excluded in a sysplex environment

Usage Notes

SMMSYSxx is the global or master parameter member. It is read by MAINVIEW SRM during subsystem startup. The suffix specifications in SMMSYSxx indicate which versions of parameter members (SMPOOLxx, SMCALSxx, SMVARSxx, SMFUNCxx) contain specifications for the current execution of MAINVIEW SRM.

SMMSYSxx parameters can be changed by editing the member directly, by use of the MAINVIEW SRM subsystem command set, or through the ISPF interface MAINVIEW SRM Global Administration panel.

SMMSYSxx is a required member. It must be identified on MAINVIEW SRM start up either by accepting the default value of 00 or by explicit specification. For example:

```
/S SVOS,SUF-xx
```

All parameters for SMMSYSxx are on the SET statement.

Parameter Explanations

AOO_SUBSYS=

Purpose: Specifies the AutoOperator subsystems that are to receive events. You may specify up to three AutoOperator subsystems.

Syntax: AOO_SUBSYS=xxxx

where *xxxx* is the four-character subsystem name used by the AutoOperator component within the OS/390 image.

Required: Only if routing events to AutoOperator.

Default: None

AUTOPROC=

Purpose: Specifies the name of the cataloged procedure used to start SG-Auto. The cataloged procedure is distributed in BBSAMP as member SGAPROC.

Syntax: AUTOPROC=xxxxxxxx

where *xxxxxxxx* is a 1–8 character string.

Required: No

Default: None

BBI3_SSID=

Purpose: Specifies the CAS subsystem name to which the SVOS PAS should connect. Since the BBI3 connection occurs during SVOS startup, SVOS will not start if BBI3_SSID is not specified. To update the value of BBI3_SSID, SVOS must be stopped and restarted; it cannot be refreshed.

The CAS subsystem name is specified in the SSID= parameter on the PARM= keyword for the CAS JCL EXEC statement.

Syntax: BBI3_SSID=xxxx

where *xxxx* is a 1–4 character string.

Required: Yes

Default: None

BCDS n =

Purpose: Specifies HSM CDS database files to be used by MAINVIEW SRM.

Syntax: **BCDS n =xxxxxxxx**

where n is the multi-cluster number. The numbers 2, 3, and 4 are specified only if defining a multicluster data set

where xxxxxxxx is a backup data set name

Required: No

Default: None

BLKINPUT=

Purpose: Changes block size for input data sets.

Syntax: **BLKINPUT=YES/NO**

Required: No

Default: No

BLKOLDSR=

Purpose: Changes blocksize for output data sets opened with disposition of old or shared.

Syntax: **BLKOLDSR=YES/NO**

Required: No

Default: No

CAL=

Purpose: Specifies the suffix of the SMCALS xx member. SMCALS xx contains calendar parameters used to specify non-working days for the DFHSM-related functions.

Syntax: **CAL= xx**

where xx is any two-character string. A single character is not allowed.

Required: No

Default: None

CHECK=

Purpose: Specifies candidate volume replacement during allocation for multivolume data sets. If multiple volumes are requested for a data set, VSAM or non-VSAM, the CHECK parameter controls whether the first volume or all volumes are immediately assigned from the assigned pool.

Syntax: CHECK=*FIRST/ALLVOLS*

Required: No

Default: CHECK=FIRST

CRITLIST=

Purpose: Specifies the suffix of the SMCRIT $_{xx}$ member. SMCRIT $_{xx}$ members contain lists of critical data set names to be used by the VOLSEL option of CRITDSN.

Syntax: CRITLIST= $_{xx}$

Required: No

Default: None

DADSMEX=

Purpose: Determines if the DADSM preprocessing exit (IGGP $_{RE00}$) is called. If it is called and the exit returns a zero, the volume is used for a volume switch. If the exit returns a four, scanning continues for an acceptable volume. If the exit returns an eight, no volume switch occurs.

Syntax: DADSMEX=*YES/NO*

Required: No

Default: DADSMEX=YES

DASDGENR=

Purpose: Specifies the only generic DASD names that are to be processed for specific functions if no space requirements are specified. Standard pooling functionality allows the JCL UNIT parameter to specify a pool name. If this is not desired, DASDGENR can be used to intercept data set allocations without space information. Such allocations will only be considered for subsequent processing if the unit is found in an internal table (3380, 3350, and so on) or if the unit is found in the DASDGENR list. DASDGENR affects functions DASDPOOL, DSNCHECK, SETEXPDT, and FORCECAT.

Syntax: DASDGENR=(xxxxxxxx,xxxxxxxx,...)

where xxxxxxxx is a generic DASD name 1–8 characters long. Multiple names can be specified.

Required: No

Default: None

Note: If this parameter is not coded, all data sets with DASD generics or esoterics will be processed. If any parameter is coded for DASDGENR, only unit names in the DASDGENR list will be processed, so all generic/esoteric names that are to be processed by EasyPOOL should be specified.

DATEFMT=

Purpose: Specifies the format of calendar dates in MAINVIEW SRM reports and display screens.

This parameter does not apply to the format of input dates for SMCALSxx parameters.

Syntax: DATEFMT=*MMDD/DDMM*

where MMDD specifies a date format of mm/dd/yyyy (American style), and DDMM specifies a date format of dd/mm/yyyy (European style).

Required: No

Default: DATEFMT= MMDD

DCTYPE=

Purpose: Specifies which device characteristics are to be maintained across volumes during a volume switch. For example, if your installation wishes to segregate data sets residing on cached and non-cached devices, the **DCTYPE=(CACHE)** operand can be used to ensure that the cache property is maintained during a volume switch. Data sets residing on a cached device only switch to volumes that are also cached. Likewise, data sets on non-cached devices switch to only non-cached volumes.

Syntax: **DCTYPE=(CACHE,SHARE, DUALCOPY, FASTWRITE)**

Multiple device characteristics can be specified.

If **DCTYPE** has been specified, **CHECK=DC** on the **SKIP** parameter can be used to ignore the device characteristic checks for special conditions.

Required: No

Default: None

DFREORGPRC=

Purpose: Defines default SPACVOLA reorganize procedure name of the procedure that is started if SPACVOLA reorganize processing is requested.

Syntax: **DFREORGPRC=xxxxxxxx**

where *xxxxxxxx* is the name of the reorganize procedure.

Required: No

Default: **DFREORGPRC=REORGPRC**

DIAG=

Purpose: Suffix of parameter member **SMDIAGxx**.

Syntax: **DIAG=xx**

where *xx* is any two-character string. A single character is not allowed.

Required: No

Default: None

DIAGMSDD=

Purpose: Establishes WTO message tracing.

Syntax: `DIAGMSDD=xxxxxxx`

where `xxxxxxx` is a 1–8 character DD name.

Required: No

Default: `DIAGMSDD=PRODN$`

DISPLAY=

Purpose: Determines which functions are displayed in the ISPF interface functions panel. `DISPLAY=ALL` displays all functions for the MAINVIEW SRM products; `DISPLAY=LIC` shows only those functions in the products for which you are licensed; `DISPLAY=ACT` shows only those functions that are active.

Syntax: `DISPLAY=ALL/LIC/ACT`

Required: No

Default: None

DMYUNIT=

Purpose: Defines the conversion of a nonexistent UNIT parameter to a valid UNIT parameter only if `JCLEXT=YES`.

Syntax: `DMYUNIT=(xxxxxxx,zzzzzzzz,...xxxxxxx,zzzzzzzz)`

where the first `xxxxxxx` is the invalid UNIT parameter to be converted to the valid UNIT parameter `zzzzzzzz`. Multiple pairs of DMYUNITs can be specified.

Required: No

Default: None

DP_RENAME=

Purpose: Specifies to EasyPOOL that during DADSM RENAME, DASDPOOL will be driven to verify that the POOL containing the volume on which the data set currently resides is also a POOL that would be assigned to the renamed data set. If the first POOL in which the current volume is found does not match a POOL that would be assigned to the renamed data set, the RENAME will be denied.

Syntax: DP_RENAME=YES/NO

Required: No

Default: DP_RENAME=NO

Note: The new FLST/RLST parameter DADSM_FUNC should be used to limit the data sets processed by enabling this option.

DUMPDD=

Purpose: Produces SYS1.DUMPxx dump if MAINVIEW SRM abends.

Syntax: DUMPDD=xxxxxxxx

where xxxxxxxx is a 1–8 character DD name.

Required: No

Default: DUMPDD=PRODUMP

ETS_ID=

Note: ETS_ID is no longer supported. The Ensign Alarm Console is no longer supported in the Enterprise Storage Automation component. If the keyword is specified, the SVM0766I messages is issued and the value is ignored.

EVNT=

Purpose: Specifies the suffix you assign to the name of the SMEVNTxx event definition member.

Syntax: EVNT=xx

where xx is the two-character suffix of the SMEVNTxx member name.

Required: Yes

Default: None

FDRIAM=

Purpose: Determines whether a data set is an IAM data set.

Note: *Only IAM customers should set this parameter to FDRIAM=YES.*

Syntax: FDRIAM=YES/NO

Required: No

Default: FDRIAM=NO

FORPLEXNAME=

Purpose: Specifies one or more user-defined PLEXNAMEs that can be included or excluded in a sysplex environment.

Syntax: FORPLEXNAME=xxxxxxxx

Required: No

Default: FORPLEXNAME=*current image*

FORSMFID=

Purpose: Specifies SMF records that can be included or excluded in a sysplex environment.

Syntax: FORSMFID=xxxxxxxx

Required: No

Default: FORSMFID=*current image*

FORSYSID=

Purpose: Specifies user-defined system IDs that can be included or excluded in a sysplex environment.

Syntax: FORSYSID=xxxxxxxx

Required: No

Default: FORSYSID=*current image*

FUNC=

Purpose: Specifies the suffix of the SMFUNC xx member. SMFUNC xx contains function definition parameters. A function must be included in the SMFUNC xx member to be available during MAINVIEW SRM execution. Each function definition identifies two other members that define

- The resources affected by the function (SMFLST xx —the filter list)
- How those resources are affected (SMRLST xx —the rules list)

Note that some functions do not use a rules list, but all functions require a filter list. See the discussion for SMFUNC xx in “Function Member Parameters” on page B-77.

Syntax: FUNC= xx

where xx is any two-character string. A single character is not allowed.

Required: Yes

Default: None

HISTDAYS=

Purpose: Specifies the number of days of data to retrieve from the StorageGUARD historical performance data collector and temporarily store in a data space. Values are 0-14 (7 or 14 recommended), which indicate the number of days of data to be retrieved. The default value is 0, which indicates that the data space is created, but no historical performance data is gathered and no HISTDPO pooling can be performed. If this value is changed, the historical performance data collector (SVSGP) must be stopped and restarted.

Syntax: HISTDAYS= nn

Required: No

Default: HISTDAYS=0

HLOGAUTH=

Purpose: Specifies the *hours* component of the duration between automatic logfile switching.

EasyHSM reporting extracts records from the DFHSM logfiles. If HLOGCOLL=YES is specified, MAINVIEW SRM will switch the DFHSM logfile and extract the required records for EasyHSM automatically. The switching interval is specified in hours and minutes by the parameters HLOGAUTH and HLOGAUTM.

Syntax: HLOGAUTH=*nn*

where *nn* specifies a number of hours in the range 0–24.

Required: Yes

Default: HLOGAUTH=00

HLOGAUTM=

Purpose: Specifies the *minutes* component of the duration between automatic logfile switching.

EasyHSM reporting extracts records from the DFHSM logfiles. If HLOGCOLL=YES is specified, MAINVIEW SRM will switch the DFHSM logfile and extract the required records for EasyHSM automatically. The switching interval is specified in hours and minutes by the parameters HLOGAUTH and HLOGAUTM.

Syntax: HLOGAUTM=*nn*

where *nn* specifies a number of minutes in the range 0–60.

Required: No

Default: HLOGAUTM=00

HLOGCOLL=

Purpose: Specifies whether MAINVIEW SRM will perform DFHSM logfile switching and record extraction for EasyHSM automatically.

EasyHSM reporting requires certain records from the DFHSM logfiles. If HLOGCOLL=YES is specified, MAINVIEW SRM will switch the DFHSM logfiles and run a record extraction program automatically at the interval specified by the HLOGAUTH/M parameters. For more information on DFHSM logfile switching and extraction, see the *MAINVIEW SRM EasyHSM User Guide and Reference*.

Syntax: HLOGCOLL=YES/NO

Required: No

Default: HLOGCOLL=NO

HLOGINDX=

Purpose: Specifies the prefix of the EasyHSM data set that will contain the records extracted from the DFHSM logfile. The prefix may contain up to 20 characters in any number of name qualifiers. The full data set name generated for the log extract file is

prefix.Dyymmdd.Thhmmss.SYSsystem-id

Syntax: HLOGINDX=xxxxxxxxxxxxxxxxxxxx

Required: A name is required for the EasyHSM reporting facility to be functional.

Default: None

HLOGPRIM=

Purpose: Specifies the number of tracks to be allocated for the log extract file. One-half of the primary extent is allocated for the secondary (with a minimum of 1). If not specified, 15 tracks are used for primary and 10 tracks for secondary.

Syntax: HLOGPRIM=*nnn*

where *nnn* is a number in the range 1–999.

Required: No

Default: HLOGPRIM=15

HLOGTASK=

Purpose: Specifies the name of the procedure to be run following the EasyHSM DFHSM logfile switch program execution.

EasyHSM reporting extracts records from the DFHSM logfiles. If HLOGCOLL=YES is specified, MAINVIEW SRM will switch the DFHSM logfile and extract the required records for EasyHSM automatically. HLOGTASK may be used to run a task associated with the logfile switch performed by the MAINVIEW SRM utility.

Syntax: HLOGTASK=xxxxxxx

Required: No

Default: None

HLOGUNIT=

Purpose: Specifies the esoteric or generic unit name for the allocation of the log extract file. If not specified, SYSALLDA is used.

Syntax: HLOGUNIT=xxxxxxx

Required: No

Default: HLOGUNIT=SYSALLDA

HLOGYDSN=

Purpose: Specifies the fully-qualified data set name of DFHSM logfile Y.

Syntax: HLOGYDSN=xxxxxxxx.xxxxxxx...

Required: A name is required for the EasyHSM reporting facility to be functional.

Default: None

HSMACTID=

Purpose: Specifies the high-level data set name qualifier for the DFHSM activity data sets. This name qualifier is used by the EasyHSM output management facility to build the name of the DFHSM activity data sets that are used as input.

Syntax: HSMACTID=xxxxxxx

where xxxxxxx is any 1–8 character string.

Required: No

Default: HSMACTID=DFHSM

IGNOREDD=

Purpose: Suppresses all MAINVIEW SRM activity for the jobstep containing the specified DD name. No MAINVIEW SRM functions will occur for any data set in the jobstep. Note that the IGNORE parameter in the DIAG member will override the DD statement's presence.

Syntax: IGNOREDD=xxxxxxx
where xxxxxxx is 1–8 character DD name.

Required: No

Default: IGNOREDD=PROIGN

JCLEXT=

Purpose: Specifies if EasyPOOL will obtain volume and unit information after MVS accessed the catalog.

Syntax: JCLEXT=YES/NO

Required: No

Default: JCLEXT=YES

Note: If JCLEXT=YES is specified, all non-valid unit names must be specified in DMYUNIT; otherwise, MVS will fail the allocation. Also, JCLEXT=YES should be used carefully if PROCOLD=YES is also specified, because JCLEXT will find a unit and volume from the catalog, whereas PROCOLD=YES will allow the existing data set to be reprocessed, possibly assigning a different (and invalid) volume.

JCLEXT=NO is primarily supplied for compatibility with the POOLDASD product. Under MAINVIEW SRM, there is no significant benefit to specifying JCLEXT=NO.

JCLUREQ=

Purpose: When PROCOLD=YES is specified and EasyPOOL is analyzing a DD statement for an existing data set, the JCLUREQ parameter determines if UNIT information is required. If JCLUREQ=YES is specified, MAINVIEW SRM intercepts the DD statement only when the JCL specifies UNIT parameter. This allows the installation to correct JCL that uses an invalid unit parameter for existing data sets without analyzing DD statements that are correctly specified. EasyPOOL examines every DD statement associated with an existing data set if you specify JCLUREQ=NO.

Note: JCLUREQ=NO needs to be specified when processing UNIT=AFF groups since the unit field in the SIOT consists of blanks.

Syntax: JCLUREQ=YES/NO

Required: No

Default: JCLUREQ=NO

MAXVOL=

Purpose: Limits the number of volumes that SPACVOLA allows a data set to use. When additional units are allocated with UNIT=(SYSDA,*n*), they are counted if space is obtained on a volume. If *n* is greater than the MAXVOL value, SPACVOLA does not limit the use of the additional volumes, but does not add additional volumes through a volume switch.

Syntax: MAXVOL=*nn*

where *nn* is any number in the range 1–59.

Required: No

Default: MAXVOL=5

Note: This option can be set globally and overridden by individual SPACVOLA RLST action statements.

The MAXVOL parameter and MVS will allow a data set to extend across as many as 59 volumes. However, some products using DFP 3.3 CAMLST services will only process up to 20 volumes, due to a limitation in the CAMLST processing (see the IBM manual SC26-4567 *MVS/DFP 3.3 System Programming Reference*, section 4.3, Retrieving Information from a Catalog). If you are using DFP 3.3 or earlier, and are using products that use CAMLST services to process multivolume data sets, you may wish to set the MAXVOL parameter to less than 20 volumes.

MCDSn=

Purpose: Specifies HSM migrated data set file allocated during EasyHSM startup.

Syntax: `MCDSn=xxxxxxxx`

where *n* is the multi-cluster number. The numbers 2, 3, and 4 are specified only if defining a multicluster data set.

where `xxxxxxxx` is a migrated data set name

Required: No

Default: None

MODTRCDD=

Purpose: Establishes module entry/exit tracing.

Syntax: `MODTRCDD=xxxxxxxx`

where `xxxxxxxx` is a 1–8 character DD name.

Required: No

Default: `MODTRCDD=PROTRCE`

MREDUCE=

Purpose: Determines whether secondary space reduction can occur on multivolume data sets that were allocated with JCL. For example, `MREDUCE=NO` would not allow secondary space reduction for the first three volumes when allocated with `UNIT=(SYSDA,4)`.

Syntax: `MREDUCE=YES/NO`

Required: No

Default: `MREDUCE=YES`

MSGID=

Purpose: Specifies the inclusion of the MAINVIEW SRM message identifier in the message text. For example:

MSGID=YES

```
15.00.30 JOB29640 SVM3352I EMPCRMX,SA,DD1,EMPCRM.Q2.TEST  
REQUESTED SPACE 0463KB EXCEEDS LIMIT
```

MSGID=NO

```
14.59.37 JOB29639 EMPCRMX,SA,DD1,EMPCRM.Q2.TEST  
REQUESTED SPACE 0463KB EXCEEDS LIMIT
```

Syntax: MSGID=YES/NO

Required: No

Default: MSGID=YES

MSGLVL=

Purpose: Specifies the level of messages that will be printed. The options are I=Informational, W=Warning, E=Error, S=Severe error. Messages are inclusive of increasing levels of severity. When a particular level is chosen, messages of that level and greater will be printed. For example, if E is chosen, messages of severity E or S are printed.

Syntax: MSGLVL=I/W/E/S

Required: No

Default: MSGLVL=I

MSGPREF=

Purpose: Specifies the three-character message identifier prefix on MAINVIEW SRM messages.

Syntax: MSGPREF=*xxx/SVM*

where *xxx* is any three-character string. Fewer than three characters is not allowed.

Required: No

Default: MSGPREF=SVM

NOCATDYN=

Purpose: Allows NOCATLG2 to process dynamically allocated data sets.

Syntax: NOCATDYN=*YES/NO*

Required: No

Default: NOCATDYN=NO

NOCATPFX=

Purpose: Specifies the second-level qualifier to be used by NOCATLG2 when renaming a data set. NOCATLG2 can be directed to rename existing data sets by specifying the NOCATLG2=RENAME operand on the RLST action parameter.

Syntax: NOCATPFX=*xxx*

where *xxx* is 1 to 3 alphanumeric or national characters. The first character must be alphabetic.

Required: No

Default: NOCATPFX=BAB

NOCATPRG=

Purpose: Specifies if NOCATLG2 can scratch a data set that has not reached its expiration date when NOCATLG2=SCRATCH is specified in the RLST action parameter. NOCATPRG=YES indicates that the data set is to be scratched regardless of expiration date (that is, scratch is issued with the PURGE option). This option can be set globally and overridden on individual action statements by the PURGE operand on the RLST action parameter.

Syntax: NOCATPRG=*YES/NO*

Required: No

Default: NOCATPRG=NO

NOCATSEC=

Purpose: Specifies the level of security checking that NOCATLG2 performs before scratching or renaming a data set. If the creator of the new data set does not have the specified level of authority, NOCATLG2 will not scratch or rename the existing data set.

Syntax: NOCATSEC=NONE/CREATE/READ/UPDATE/ALTER

Required: No

Default: NOCATSEC=NONE

NOCATSMS=

Purpose: Specifies whether SMS-managed data sets can be renamed, uncataloged, or scratched by NOCATLG2. If NOCATSMS=YES is specified, NOCATLG2 renames, uncatalogs, or scratches an existing SMS-managed data set if NOCATLG2=RENAME, UNCATLG, or SCRATCH is specified. The existing SMS-managed data set is renamed/uncataloged/scratched whether or not the new data set to be allocated is SMS-managed.

Note: If NOCATLG2=UNCATLG is specified and the existing data set is SMS-managed, NOCATLG2 issues a DELETE NOSCRATCH to remove the catalog entry for the data set. The data set will exist on the SMS-managed volume but will not be cataloged. The catalog entry will point to the newly allocated data set instead.

Syntax: NOCATSMS=YES/NO

Required: No

Default: NOCATSMS=NO

NOCATVOL=

Purpose: Allows a new data set to be allocated to the same volume to which it was previously cataloged. This can occur when a catalog entry is *orphaned* because a data set is removed from a volume without the data set being uncataloged or when a catalog entry is added for a data set but the data set is never actually created. This operand acts differently depending on the NOCATWHEN setting in the NOCATLG2 RLST. If NOCATWHEN=TERM is specified and NOCATLG2=UNCATLG is specified in the relevant RLST action parameter, NOCATVOL=SAME indicates that the data set should be uncataloged and recataloged even if the old and new volumes are the same and NOCATVOL=DIFF means that the old catalog entry is left in the catalog if the old and new volumes are the same.

If the relevant RLST action parameter specifies NOCATLG2=SCRATCH or NOCATLG2=RENAME, NOCATVOL=DIFF is forced and the NOCATLG2 processing will fail with an error message if the old and new volumes are the same.

If NOCATWHEN=ALLOC is specified, NOCATVOL is ignored and NOCATLG2 processing occurs, even if the old and new volumes are the same.

Syntax: NOCATVOL=SAME/DIFF

Required: No

Default: NOCATVOL=DIFF

NOCATWHEN=

Purpose: Specifies when NOCATGL2 processing occurs for non-SMS managed data sets. ALLOC indicates that NOCATGL2 processing occurs during data set allocation. TERM indicates that NOCATGL2 processing occurs during step termination. NOCATGL2 processing for SMS-managed data sets must occur during data set allocation.

Syntax: NOCATWHEN=ALLOC/TERM

Required: No

Default: NOCATWHEN=ALLOC

Note: When NOCATWHEN=ALLOC and no volser is specified, you will not be able to filter on parameter VOL=. To be able to filter on VOL=, you must specify NOCATWHEN=TERM.

OCDS=

Purpose: Specifies HSM OCDS data sets to be defined and allocated during EasyHSM startup.

Syntax: OCDS=xxxxxxxx
where xxxxxxxx is an OCDS data set name

Required: No

Default: None

OPMHLQ=

Purpose: Specifies the high-level qualifier for the data sets created by the MAINVIEW SRM Output Management Facility, a part of EasyHSM that allows selection of DFHSM and DFSS messages for reporting and analysis.

Syntax: OPMHLQ=xxxxxxxx
where xxxxxxxx is any 1–8 character string.

Required: Only if the MAINVIEW SRM Output Management Facility is to be used.

Default: None

ORIGDATA=

Purpose: During EasyPOOL processing, ORIGDATA specifies whether VOL and UNIT contain the original volser and unit values from the JCL or the current value. If ORIGDATA=PRO is used, the selection fields VOL and UNIT will always contain the original volser and unit from the JCL. This is how these fields are handled in earlier releases of MAINVIEW SRM. In POOL-DASD these fields reflected any changes made to the volser and unit by showing the current value. If ORIGDATA=POOL is used, MAINVIEW SRM will reflect the current values for these fields.

Note: The fields ORIGVOL and ORIGUNIT will always contain the original VOLSER and UNIT that were specified in the JCL.

Syntax: ORIGDATA=PRO/POOL

Required: No

Default: ORIGDATA=PRO

PASSWORD=

Purpose: Specifies the password(s) supplied by BMC Software. One or more passwords can be required, depending on which MAINVIEW SRM components you purchased.

Syntax: PASSWORD=xxxxxxxxxx
where xxxxxxxxxxxx is a 16-character string.

Required: Yes

Default: None

PERFRM_PRC=

Purpose: Specifies the name of the procedure used to start the historical performance collector. The procedure is distributed in *?prefix.BBSAMP* as member SGPPROC.

Syntax: PERFRM_PRC=xxxxxxxx
where xxxxxxxx is an 8-character string.

Required: No

Default: None

POOL=

Purpose: Specifies the suffix of an SMPOOL_{xx} member. SMPOOL_{xx} contains non-SMS managed device pool definition parameters. It names pools and assigns volumes to pools.

Syntax: POOL=xx
where xx is any two-character string. A single character is not allowed.

Required: Yes

Default: None

PROCOLD=

Purpose: Specifies if EasyPOOL will intercept DD statements that specify OLD allocations. EasyPOOL always intercepts NEW and MOD allocations.

EasyPOOL also intercepts DD statements that specify the UNIT parameter when PROCOLD=YES is specified, which is useful for correcting questionable JCL. For example, assume DASDPOOL pools a data set to a TEST volume. If a later step wants to use the data set and specifies UNIT=PROD, the job receives a JCL error. You can correct this situation by specifying PROCOLD=YES.

Syntax: PROCOLD=*YES/NO*

Required: No

Default: PROCOLD=NO

Note: PROCOLD=YES is useful to override volsers that are hardcoded in JCL for old data sets. PROCOLD=YES will logically remove that hard-coded volume. However, use caution if you also specify JCLEXT=YES, which will assign a unit and volume from the catalog.

REJECT=

Purpose: Controls termination of processing at the first or last data set rejected by REJECT=YES in DASDPOOL or DSNCHECK. If termination is to take place on the first rejected data set, code REJECT=FIRST. If all data sets are to be processed before control is returned to MVS allocation, code REJECT=LAST.

Syntax: REJECT=FIRST/LAST

Required: No

Default: REJECT=FIRST

REQTYPE=

Purpose: Specifies whether the MNTYPE statement in SPACVOLA is considered the request type instead of the mount type. For example, VOL=SER=WORK01 can be considered a private request even though the pack was mounted storage. MNTYPE defaults to the request type.

Syntax: REQTYPE=*YES/NO*

Required: No

Default: REQTYPE=YES

SCAT=

Purpose: Forces a catalog update to occur during the volume switch. By default, when SPACVOLA performs a volume switch on a permanent data set, the catalog is updated to contain the new volume(s) at step termination. For SMS-managed data sets, the catalog is always updated immediately.

Syntax: SCAT=STEPEND/IMMEDIATE

Required: No

Default: SCAT=STEPEND

SG_INITPOOL=

Purpose: Specifies the maximum number of defined pools included in a single snapshot. The maximum number of defined pools at initialization of a new linear data set is 3,995 unless a greater value is specified on this parameter. After initialization, data is collected into a snapshot for the number of pools specified on this parameter.

Syntax: SG_INITPOOL=*nnnnnn*

where *nnnnnn* is a value in the range 10–999999

Required: No

Default: 1000

Note: Do not modify the default value unless you must.

SG_INITVOL=

Purpose: Specifies the maximum number of defined volumes included in a single snapshot. The maximum number of defined volumes at initialization of a new linear data set is 6,625 unless a greater value is specified on this parameter. After initialization, data is collected into a snapshot for the number of volumes specified on this parameter.

Syntax: SG_INITVOL=*nnnnnn*

where *nnnnnn* is a value in the range 10–999999

Required: No

Default: 3000

Note: Do not modify the default value unless you must.

SG_IXFPNTVL=

Purpose: Specifies the number of hours between refreshes of the IXFP data tables

Syntax: SG_IXFPNTVL=*nn*

Required: No

Default: None

SG_MAXACCT=

Purpose: Specifies the maximum number of active accounts in the SG-Control database.

Syntax: SG_MAXACCT=*nnnnn*

where *nnnnn* is a value in the range 328–32765

Required: No

Default: Extracted from the SG-Control database

SG_MAXPOOL=

Purpose: Specifies the number of pools that can be assigned to a volume. Used by the data collector when building pool snapshots.

Syntax: SGMAXPOOL=*n*

where *n* is a value in the range 1–8

Required: No

Default: SGMAXPOOL=1

SG_MAXSSDSZ=

Purpose: Specifies the number of cylinders used for a solid state disk drive. Used to distinguish between emulated and real DASD. Any device that does not exceed the value specified on this parameter is considered a solid state device.

Syntax: `SG_MAXSSDSZ=nnnnn`
where *nnnnn* is a value less than 32766

Required: No

Default: 0

SG_READNTVL=

Purpose: Specifies the frequency at which StorageGUARD creates a snapshot in core.

Syntax: `SG_READNTVL=nnnn`
where *nnnn* is a value in the range 5–9999

Required: No

Default: `SG_READNTVL=30`

SG_RETRYLIM=

Purpose: Specifies the number of abend conditions the data collector should ignore.

Syntax: `SG_RETRYLIM=nnnn`
where *nnnn* is a value from 5–9999

Required: No

Default: `SG_RETRYLIM=10`

SG_SIBSTK=

Purpose: Specifies the IXFP SIBBATCH parameter member to be used by the MAINVIEW SRM IXFP services for communications with the IXFP address space.

Syntax: `SG_SIBSTK=xxxxxxxx`

Required: No

Default: None

SG_SPACHLDR=

Purpose: Defines a data set name mask that StorageGUARD can use to identify space holder data sets.

Syntax: SG_SPACHLDR=MASK
where MASK is a space holder data set name mask

Required: No

Default: None

SG_SUBTASKS=

Purpose: Defines the number of volumes that can be read in parallel.

Syntax: SG_SUBTASKS=*nn*
where *nn* is a value in the range 2–10

Required: No

Default: None

SG_WRITNTVL=

Purpose: Defines the frequency at which snapshots are written to the StorageGUARD database.

Syntax: SG_WRITNTVL=*nn*
where *nn* is a value in the range 1–1439

Required: No

Default: SG_WRITNTVL=30

SGA_ENQSCOP=

Purpose: Specifies the operational environment in which SG-Auto is to run. If GLOBAL is specified, SG-Auto issues an ENQ with the SYSTEMS parameter. If LOCAL is specified, SG-Auto issues an ENQ with the SYSTEM parameter. Refer to the appropriate IBM documentation for a description of the ENQ macro options.

Syntax: SGA_ENQSCOP=GLOBAL/LOCAL

Required: Required for SG-Auto

Default: GLOBAL

SGACMD=

Purpose: Specifies the two-position suffix of the initial command for executing the SG-Auto started task. The suffix will be appended to SGACMD to form the member name as it exists in the MAINVIEW SRM parmlib.

Syntax: SGACMD=

Required: No

Default: None

SGASCAN=

Purpose: Specifies whether SG-Auto should be started in scan mode.

Syntax: SGASCAN=YES/NO

Required: Yes

Default: None

SGASIM=

Purpose: Specifies whether SG-Auto should be started in SIMULATION mode.

Syntax: SGASIM=YES/NO

Required: No

Default: None

SGC_ADDEXIT=

Purpose: Defines the name of SG-Control Add Exit.

Syntax: SGC_ADDEXIT=xxxxxxx
where xxxxxxxx is a 1–8 character string

Required: No

Default: None

SGC_CHKEXIT=

Purpose: Defines the name of SG-Control Check Exit.

Syntax: SGC_CHKEXIT=xxxxxxx
where xxxxxxxx is a 1–8 character string

Required: No

Default: None

SGC_DEFEXIT=

Purpose: Defines the name of SG-Control Default Exit.

Syntax: SGC_DEFEXIT=xxxxxxx
where xxxxxxxx is a 1–8 character string

Required: No

Default: None

SGC_KEYEXIT=

Purpose: Defines the name of SG-Control Account Code Build Exit.

Syntax: SGC_KEYEXIT=xxxxxxx
where xxxxxxxx is a 1–8 character string

Required: No

Default: None

SGC_SECEXIT=

Purpose: Defines the name of SG-Control Security Exit.

Syntax: `SGC_SECEXIT=xxxxxxxx`
where `xxxxxxxx` is a 1–8 character string

Required: No

Default: None

SGC_SELEXIT=

Purpose: Defines the name of SG-Control Select Exit.

Syntax: `SGC_SELEXIT=xxxxxxxx`
where `xxxxxxxx` is a 1–8 character string

Required: No

Default: None

SGC_STOGRP=

Purpose: Specifies whether or not to retrieve SMS storage group information. This parameter may only be set to YES if SMS storage group information is required for FLST or RLST processing.

Syntax: `SGC_STOGRP=YES/NO`

Required: No

Default: `SGC_STOGRP=NO`

SGC_STORCLS=

Purpose: Specifies whether or not to retrieve storage class information. This parameter may only be set to YES if SMS storage class information is required for FLST or RLST processing. `SGC_STORCLS` includes data class, management class, and storage class information.

Syntax: `SGC_STORCLS=YES/NO`

Required: No

Default: `SGC_STORCLS=NO`

SGCDSN=

Purpose: Specifies the data set name for the dynamic allocation/deallocation of SG-Control database DD, namely SGCDB. The dynamic allocation occurs with the start of the SVSGC component and deallocation occurs with the stop of SVSGC. This parameter is used only if the SGCDB DD statement is not present with the SVOS startup JCL.

Syntax: SGCDSN=xxxxxx...xxxxxx
where xxxxxx...xxxxxx is a 1 to 44 character string for name of data set

Required: No

Default: None

SGD_PROCNM=

Purpose: Specifies the name of the StorageGUARD data collector started task.

Syntax: SGD_PROCNM=SGDCOLLS

Required: Required to run the data collector

Default: None

SGD_SMFID=

Purpose: Controls the generation of SMF records for StorageGUARD.

Syntax: SGD_SMFID=*nnn*
where *nnn* is a value in the range 0–255

Required: No

Default: SGD_SMFID=0

SGDCOLLECT=

Purpose: Specifies if StorageGUARD will collect pool data. This parameter may be overridden at the pool level.

Syntax: SGDCOLLECT=*YES/NO*

Required: No

Default: None

SGDCOLLECT n =

Purpose: Specifies whether StorageGUARD will collect pool data in an alternate data collector. The alternate data collector to be used is identified by the suffix of n . This parameter may be overridden at the pool level.

Syntax: SGDCOLLECT n =*YES/NO*
where n is a value in the range of 1–8

Required: No

Default: None

SGDPROCCNM n =

Purpose: Specifies the cataloged procedure to be started for a specified copy of StorageGUARD. The name of the procedure must be unique. Make sure that each procedure resides in a separate set of linear data sets (SGRDPOOL, SGRDVOL, and so on).

Syntax: SGDPROCCNM n =*xxxxxxxx*
where n is a value in the range of 1–8 and *xxxxxxxx* is the procedure name

Required: No

Default: None (For example, if SGDPROCCNM4 is not defined, you will not be able to issue the S SVSGD4 SVOS command.)

SGDSMFID n =

Purpose: Specifies the SMF record number for MAINVIEW SRM audit records written to the SMF data set for a specified copy of StorageGUARD. (Note that SMF message generation is also controlled by the SMF parameter on individual MAINVIEW SRM functions in member SMFUNC xx and by the SMF parameter on filter list specifications.)

Syntax: SGDSMFID n =*nnn*
where n is a value in the range of 1–8 and where *nnn* is a number in the range 128–255. A zero can be specified to deactivate SMF recording.

Required: No

Default: None

SGEXITACCT n =

Purpose: Defines the name of the account user exit routine for a specified copy of StorageGUARD.

Syntax: SGEXITACCT n =xxxxxxxx

where n is a value in the range of 1–8 and xxxxxxxx is 1–8 characters

Required: No

Default: None

SGEXITPOOL n =

Purpose: Defines the name of the pool user exit routine for a specified copy of StorageGUARD.

Syntax: SGEXITPOOL n =xxxxxxxx

where n is a value in the range of 1–8 and xxxxxxxx is 1–8 characters

Required: No

Default: None

SGEXITVOL n =

Purpose: Defines the name of the volume user exit routine for a specified copy of StorageGUARD.

Syntax: SGEXITVOL n =xxxxxxxx

where n is a value in the range of 1–8 and xxxxxxxx is 1–8 characters

Required: No

Default: None

SGINITPOOL n =

Purpose: Specifies the maximum number of defined pools included in a single snapshot for a specified copy of StorageGUARD. The maximum number of defined pools at initialization of a new linear data set is 3,995 unless a greater value is specified on this parameter. After initialization, data is collected into a snapshot for the number of pools specified on this parameter.

Syntax: SGINITPOOL n =*nnnnnn*

where n is a value in the range of 1–8 and where *nnnnnn* is a value in the range 10–999999

Required: No

Default: None

SGINITVOL n =

Purpose: Specifies the maximum number of defined volumes included in a single snapshot for a specified copy of StorageGUARD. The maximum number of defined volumes at initialization of a new linear data set is 6,625 unless a greater value is specified on this parameter. After initialization, data is collected into a snapshot for the number of volumes specified on this parameter.

Syntax: SG_INITVOL=*nnnnnn*

where *nnnnnn* is a value in the range 10–999999

Syntax: SGINITVOL n =*nnnnnn*

where n is a value in the range of 1–8 and where *nnnnnn* is a value in the range 10–999999

Required: No

Default: None

SGMAXACCT n =

- Purpose:** Specifies the maximum number of active accounts in the SG-Control database.
- Syntax:** SGMAXACCT n = $nnnnn$
where n is a value in the range of 1–8 and where $nnnnn$ is a value in the range 328–32765
- Required:** No
- Default:** Extracted from the SG-Control database

SGMAXPOOL n =

- Purpose:** Specifies the number of pools that can be assigned to a volume for the specified copy of StorageGUARD. Used by the data collector when building pool snapshots.
- Syntax:** SGMAXPOOL n = n
where n is a value in the range of 1–8 and n is a value in the range 1–8
- Required:** No
- Default:** None

SGMAXSSDSZ n =

- Purpose:** Specifies the number of cylinders used for a solid state disk drive for a specified copy of StorageGUARD. Used to distinguish between emulated and real DASD. Any device that does not exceed the value specified on this parameter is considered a solid state device.
- Syntax:** SGMAXSSDSZ n = $nnnnn$
where n is a value in the range of 1–8 and is a value less than 32766
- Required:** No
- Default:** 0

SGP_EXITBBS=

Purpose: Specifies the number of megabytes to allocate in a scope common data space for the StorageGUARD performance exit buffer block.

Syntax: SGP_EXITBBS=*nn*

where *nn* is a number in the range 15-99

Required: No

Default: SGP_EXITBBS=15

SGP_EXITLIB=

Purpose: Specifies the default library where the StorageGUARD Performance collector SMF exits reside.

Note: EXITLIB in SGPPROC should point to the library that contains the exit load modules: SGPERU83 and SGPERU84. If EXITLIB is *not coded* or is *left as a null* in the started task, it will default to what is coded in SGP_EXITLIB for which the default is SYS1.LINKLIB. Change SGP_EXITLIB= to the appropriate load library.

Syntax: SGP_EXITLIB=*xxxxxxxx*

Required: No

Default: SGP_EXITLIB=SYS1.LINKLIB

SGP_MAXCCUS=

Purpose: Defines the maximum number of control units that are in use during a single collection interval. The minimum value is 1; the maximum value is 310,000.

Syntax: SGP_MAXCCUS=*nnnn*

Required: No

Default: SGP_MAXCCUS=256

SGP_MAXDIRS=

Purpose: Defines the maximum number of directors that are in use during a single collection interval. The minimum value is 1; the maximum value is 20,133,000.

Syntax: SGP_MAXDIRS=*nnnn*

Required: No

Default: SGP_MAXDIRS=256

SGP_MAXDSNS=

Purpose: Defines the maximum number of data set names that are in use during a single collection interval. The minimum value is 1; the maximum value is 160,000.

Syntax: SGP_MAXDSNS=*nnnn*

Required: No

Default: SGP_MAXDSNS=1000

SGP_MAXJOBS=

Purpose: Defines the maximum number of jobs (batch, TSO, and started tasks) that are in use during a single collection interval. The minimum value is 1; the maximum value is 465,000.

Syntax: SGP_MAXJOBS=*nnnn*

Required: No

Default: SGP_MAXJOBS=200

SGP_MAXLCUS=

Purpose: Defines the maximum number of logical control unit/CHIP combinations in use during an interval. This is the maximum number of actual LCUs in use multiplied by the average number of CHPs carrying data traffic to the LCU. The minimum value is 1; the maximum value is 290,000.

Syntax: SGP_MAXLCUS=*nnnn*

Required: No

Default: SGP_MAXLCUS=256

SGP_MAXPOLLS=

Purpose: Defines the maximum number of pools that are in use during a single collection interval. The minimum value is 1; the maximum value is 316,000.

Syntax: SGP_MAXPOLLS=*nnnn*

Required: No

Default: SGP_MAXPOLLS=256

SGP_MAXPTHS=

Purpose: Defines the maximum number of CHPIDs that are in use during a single collection interval. The minimum value is 1; the maximum value is 267,000.

Syntax: SGP_MAXPTHS=*nnnn*

Required: No

Default: SGP_MAXPTHS=100

SGP_MAXPVLS=

Purpose: Defines the maximum number of physical volumes that are in use during a single collection interval. The minimum value is 1; the maximum value is 6,400,000.

Syntax: SGP_MAXPVLS=*nnnn*

Required: No

Default: SGP_MAXPVLS=250

SGP_MAXRRKS=

Purpose: Defines the maximum number of RAID ranks that are in use during a single collection interval. The minimum value is 1; the maximum value is 512.

Syntax: SGP_MAXRRKS=*nnnn*

Required: No

Default: SGP_MAXRRKS=64

SGP_MAXRSFS=

Purpose: Defines the maximum number of RVA frames that are in use during a single collection interval. The minimum value is 1; the maximum value is 512.

Syntax: `SGP_MAXRSFS=nnnn`

Required: No

Default: `SGP_MAXRSFS=16`

SGP_MAXSCLS=

Purpose: Defines the maximum number of storage classes that are in use during a single collection interval. The minimum value is 1; the maximum value is 466,000.

Syntax: `SGP_MAXSCLS=nnnn`

Required: No

Default: `SGP_MAXSCLS=256`

SGP_MAXVOLS=

Purpose: Defines the total number of online DASD volumes on the OS/390 image being monitored. Note that this is the only `SGP_MAXxxxx` parameter that depends on neither the interval length nor the amount of activity on the system. The minimum value is 1; the maximum value is 438,000.

Syntax: `SGP_MAXVOLS=nnnn`

Required: No

Default: `SGP_MAXVOLS=250`

SGP_RDFCOMP=

Purpose: Specifies whether data compression is in effect for records being written to the StorageGUARD performance resource data files.

Syntax: `SGP_RDFCOMP=YES/NO`

Required: No

Default: No

SGP_SIBSTK=

Purpose: Identifies the IXFP SIBBATCH parameter member to be used by the MAINVIEW SRM IXFP services for communications with the IXFP address space. The presence of this system parameter value indicates RVA collection is to be activated.

Syntax: SGP_SIBSTK=xxxxxxx

Required: No

Default: None

SGP_SMF42=

Purpose: Determines if the SMF 42 record is written to the SMF data set. If set to NO, the historical performance data collector does not allow the record to be written.

Syntax: SGP_SMF42=YES/NO

Required: No

Default: SGP_SMF42=NO

SGP_TRACE=

Purpose: Specifies the trace default for the StorageGUARD Performance collector.

Syntax: SGP_TRACE=xxxxxxx

Required: No

Default: SGP_TRACE=NOTRACE

SGPROCACCT n =

Purpose: Defines the name of the account REXX procedure. This parameter must not be specified unless the StorageGUARD Automation Facility is being used.

Syntax: SGPROCACCT n =xxxxxxx

where n is a value in the range of 1–8 and xxxxxxx is a string of 1–8 characters

Required: No

Default: None

SGPROCPOOL n =

Purpose: Defines the name of the pool REXX procedure for a specified copy of StorageGUARD. This parameter must not be specified unless the StorageGUARD Automation Facility is being used.

Syntax: SGPROCPOOL n =*xxxxxxxx*
where n is a value in the range of 1–8 and *xxxxxxxx* is a 1–8 string

Required: No

Default: None

SGPROCVLER n =

Purpose: Defines the name of a REXX procedure that will be invoked if the data collector encounters an I/O error while reading a VTOC for a specified copy of StorageGUARD.

Syntax: SGPROCVLER n =*xxxxxxxx*
where n is a value in the range of 1–8 and *xxxxxxxx* is a 1–8 character string

Required: No

Default: None

SGPROCVOL n =

Purpose: Defines the name of the volume REXX procedure for a specified copy of StorageGUARD. This parameter must not be specified unless the StorageGUARD Automation Facility is being used.

Syntax: SGPROCVOL n =*xxxxxxxx*
where n is a value in the range of 1–8 and *xxxxxxxx* is a 1–8 character string

Required: No

Default: None

SGREADNTVL n =

Purpose: Specifies the frequency at which StorageGUARD creates a snapshot in core for a specified copy of StorageGUARD.

Syntax: SGREADNTVL n = $nnnn$

where n is a value in the range of 1–8 and where $nnnn$ is a value in the range 5–9999

Required: No

Default: SG_READNTVL=30

SGRETRYLIM n =

Purpose: Specifies the number of abend conditions the data collector should ignore for a specified copy of StorageGUARD.

Syntax: SGRETRYLIM n = $nnnn$

where n is a value in the range of 1–8 and where $nnnn$ is a value from 5–9999

Required: No

Default: SG_RETRYLIM=10

SGSPACHLDR n =

Purpose: Defines a data set name mask that StorageGUARD can use to identify space holder data sets for a specified copy of StorageGUARD.

Syntax: SGSPACHLDR n = $xxxxxxxx$

where n is a value in the range of 1–8 and where $xxxxxxxx$ is a space holder data set name mask

Required: No

Default: None

SGSUBTASKS n =

Purpose: Defines the number of volumes that can be read in parallel for a specified copy of StorageGUARD.

Syntax: SGSUBTASKS $n=nn$

where n is a value in the range of 1–8 and where nn is a value in the range 2–10

Required: No

Default: None

SGWRITNTVL n =

Purpose: Defines the frequency at which snapshots are written to the StorageGUARD database for a specified copy of StorageGUARD.

Syntax: SGWRITNTVL $n=nnnn$

where n is a value in the range of 1–8 and where nn is a value in the range 1–1439

Required: No

Default: SG_WRITNTVL=30

SIZEISPRIM=

Purpose: Determines if the SIZE filter/rule list parameter includes only the size of the primary extent or the size of the primary and one secondary extent.

Syntax: SIZEISPRIM=YES/NO

Required: No

Default: SIZEISPRIM=YES

SKIP=

Purpose: Specifies checks to be bypassed during volume switching. There are several conditions in which the SPACVOLA function does not perform a volume switch. Some of these conditions can be bypassed with the SKIP statement. If your installation has an application that can handle data sets that dynamically become multivolume, a SKIP statement can be added to the selection language to bypass requested checks. The NOCHECK operand on the RLST action parameter can also be used to override these checks, and takes precedence over the SKIP statement.

Note: Thorough testing and verification that multivolume data sets are usable by the application is recommended before overriding these checks.

Syntax: SKIP= (CHECK=(xxxxxx,xxxxxx,...),DD name=xxxxxxx, PROG=xxxxxxx)

where CHECK=(xxxxxx,xxxxxx,...) is one or more of the following options:

EXCP	Bypasses a data set being processed with EXCPs
NOTE	Bypasses a data set being processed with NOTE macros
POINT	Bypasses a data set being processed with POINT macros
DSNAME	Bypasses a data set allocated to another DD statement within the same jobstep
ENQ	Bypasses a permanent data set allocated to a DD statement within another job
DISP	Bypasses a permanent data set being accessed without the use of a catalog
DC	Bypasses a data set that resides on a cached device Under normal conditions, the volume switch will occur only to packs that have the same device characteristics.
CONTIG	Bypasses a data set allocated with a contiguous space requirement

where

DD name=xxxxxxx is any valid file name. If DD name is not specified on the parameter, the file name is not considered in deciding whether to bypass volume switch checks. Only one DD name operand is allowed per SKIP parameter.

PROG=xxxxxxx is any valid program name. If PROG is not specified on the parameter, the program name is not considered in deciding whether to bypass volume switch checks. Only one PROG operand is allowed per SKIP parameter.

Required: No

Default: None

SMFID=

Purpose: Specifies the SMF record number for MAINVIEW SRM audit records written to the SMF data set. (Note that SMF message generation is also controlled by the SMF parameter on individual MAINVIEW SRM functions in member SMFUNCxx and by the SMF parameter on filter list specifications.)

Syntax: SMFID=*nnn*

where *nnn* is a number in the range 128–255. A zero can be specified to deactivate SMF recording.

Required: No

Default: None

SMS_ALLOC=

Purpose: Specifies to EasyPOOL that SMSSELCT will be driven during DADSM ALLOCATE. If a POOL is coded in SMSPOOL, the current volume will be compared to the volumes in the POOL. If the current volume is not in a POOL assigned to the data set, the volume will be rejected with a DADSM return code of 4.

Syntax: SMS_ALLOC=*YES/NO*

Required: No

Default: SMS_ALLOC=NO

Note: The new FLST/RLST parameter DADSM_FUNC should be used to limit the data sets processed by enabling this option.

SMS_EXTEND=

Purpose: Specifies to EasyPOOL that SMSSELECT will be driven during DADSM EXTENDNV (new volume). If a POOL is coded in SMSPOOL_EXT, the current volume will be compared to the volumes in the POOL. If the current volume is not in a POOL assigned to the data set, the volume will be rejected with a DADSM return code of 4.

Syntax: SMS_EXTEND=*YES/NO*

Required: No

Default: SMS_EXTEND=NO

Note: The new FLST/RLST parameter DADSM_FUNC should be used to limit the data sets processed by enabling this option.

SMSPool=

Purpose: Specifies the suffix of the SMS pool member. An SMSPOOL xx member contains device pool definition parameters. It names SMS subpools and assigns volumes to them.

Syntax: SMSPool= xx

where xx is the two-character suffix of the SMS pool member.

Required: No

Default: None

STKSCR=

Purpose: Specifies the default location of scratch tapes for the STKSUPP function.

Syntax: STKSCR=(xxx,xxx,xxx,xxx)

The four suboperands of STKSCR are

- Standard-label tapes
- Non-label tapes
- ASCII tapes
- Non-standard label tapes

For each suboperand, xxx specifies IN (inside a silo), OUT (outside a silo), or a number (specific silo number).

Required: No

Default: None

SYSLIB=**SYSLIBn=**

Purpose: Specifies a cataloged data set name for the LPALIB library concatenations that is to be allocated at SVOS startup as a default. This parameter can be overridden by a SYSLIB DD statement in JCL. LPALIB data sets must be the same as they were when the system was last IPLd with a CLPA and/or an MPLA. There is a limit of three data sets that can be concatenated.

Syntax: SYSLIB=xxxxxxxxxxx

where xxxxxxxxxxxx is a fully qualified cataloged data set name for the LPALIB library concatenations.

SYSLIBn=xxxxxxxxxxx

where *n* is data set 2 or 3 and where xxxxxxxxxxxx is a fully qualified cataloged data set name for the LPALIB library concatenations.

Required: No

Default: None

TAPEGENR=

Purpose: Specifies tape device generic names that some EasyPOOL functions will intercept.

If you want to intercept all tape requests, specify ALLTAPE as the first generic name. (However, you cannot use ALLTAPE when JCLEXT=NO.) TAPEGENR affects functions DSNCHECK and SETEXPDT.

Syntax: TAPEGENR=(xxxxxxxx,xxxxxxxx,xxxxxxxx,....)

Required: No

Default: None

Note: If this parameter is not coded, all data sets with tape generics or esoterics will be processed. If any parameter is coded for TAPEGENR, only those tape unit names in TAPEGENR will be processed, so all generic/esoteric unit names that are to be processed should be specified. PROCOLD determines whether EasyPOOL intercepts DD parameters associated with existing data sets. Specify PROCOLD=YES if you want to convert unit information for existing tape data sets. EasyPOOL can then intercept DD parameters for existing data sets that also specify UNIT.

TRACEDD=

Purpose: Traces all MAINVIEW SRM functions for the jobstep containing the specified DD name. This is the same type of filter/rule list trace as produced by the TRACE parameter for the SMFUNCxx function definition; however, using TRACEDD, *all* MAINVIEW SRM functions will be traced for a single jobstep, based on the presence of a JCL DD name.

Syntax: TRACEDD=xxxxxxx
where xxxxxxx is a 1–8 character DD name.

Required: No

Default: None

TRKCYL=

Purpose: Specifies the number of tracks per cylinder for the default device type. The value specified for 3380/3390/9345 devices should be 15. (Note that this specification is the same as the SCDS base configuration DEFINE under ISMF for DFSMS.)

TRKCYL and TRKLEN are used by the DASDPOOL function to convert allocations in tracks or cylinders to megabytes for volume selection based on available space; for example, VOLSEL=BESTFIT. The information specified on these two parameters should reflect the devices that are most prevalent in your environment.

Syntax: TRKCYL=nnnnn
where nnnnn is a 1 to 5 digit number.

Required: Yes

Default: None

TRKLEN=

Purpose: Specifies the number of bytes per track for the default device type. Valid values are:

3380 - 47,476

3390 - 56,664

9345 - 46,456

Note: Note that this specification is the same as the SCDS base configuration DEFINE under ISMF for DFSMS.)

TRKCYL and TRKLEN are used by the DASDPOOL function to convert allocations in tracks or cylinders to megabytes for volume selection based on available space; for example, VOLSEL=BESTFIT. The information specified on these two parameters should reflect the devices that are most prevalent in your environment.

Syntax: TRKLEN=*nnnnnnnn*

where *nnnnnnnn* is a 1 to 7 digit number.

Required: Yes

Default: None

USECAT=

Purpose: Specifies whether the catalog name is used as a selection criteria in any MAINVIEW SRM ACS replacement function (SMSACSDC, SMSACSMC, SMSACSSC, SMSACSSG). Can cause an embrace with catalog functions.

Syntax: USECAT=*YES/NO*

Required: No

Default: USECAT=NO

VAR=

Purpose: Specifies the suffix of the SMVARS*xx* member. SMVARS*xx* contains variables definition parameters. The values of defined variables are substituted in MAINVIEW SRM selection statements to simplify the specification of large selection criteria used in multiple statements.

Syntax: VAR=*xx*

where *xx* is any two-character string. A single character is not allowed.

Required: No

Default: None

VSAMJCL=

Purpose: Controls the level of processing of VSAM data sets by EasyPOOL.

With VSAMJCL=CLUSTER, the EasyPOOL functions are invoked for the VSAM cluster if the volume list is defined at the cluster level, or if volume lists of the two components are identical to each other. The EasyPOOL functions are invoked at the component level if the volume list is defined at the component level and are *not* identical to each other.

VSAMJCL=COMPONENT causes the JCL function to process at the component level regardless of how the volume list is defined.

Syntax: VSAMJCL=CLUS/COMP

Required: No

Default: VSAMJCL=COMP

VSAMLIMWARN=

Purpose: Specifies the percentage value to be used before issuing the RESOLVE SRM 4GB limit message. This is an informational message to show how close a non-extended format VSAM file is to the 4GB limit.

Syntax: VSAMLIMWARN=*xx*

where *xx* is a two-digit number in the range 0–99

Required: No

Default: VSAMLIMWARN=90

VSAMPRIM=

Purpose: Specifies that volume additions to a VSAM file (by SPACVOLA) will use the primary allocation size instead of the secondary.

Syntax: VSAMPRIM=YES Use the *primary* allocation size.

VSAMPRIM=NO Use the *secondary* allocation size.

Required: No

Default: VSAMPRIM=NO

VSAMZSEC=

Purpose: Specifies which StopX37/II function controls recoveries for VSAM out-of-space conditions when no secondary allocation amount was specified when the data set was defined. Specifying VSAMZSEC=YES indicates that the SPACSECA function controls whether recovery is allowed. VSAMZSEC=NO specifies that the SPACVOLA function will determine whether recovery is allowed. If an out-of-space condition occurs for a VSAM data set because no secondary allocation amount was specified, and VSAMZSEC=YES is specified, StopX37/II will only recover from the error if the SPACSECA function is active for the same data set. Specifying VSAMZSEC=YES and not activating the SPACSECA function for a data set indicates that VSAM data sets that do not have a secondary allocation amount cannot be recovered.

Syntax: VSAMZSEC=YES/NO

Required: No

Default: VSAMZSEC=YES

VSCAN_MNTSK=

Purpose: Specifies the minimum number of tasks (TCBs) used by the VTOC scan to perform the collection.

Syntax: VSCAN_MNTSK=*nn*
where *nn* is 2 to 30

Required: No

Default: 2

VSCAN_MXTSK=

Purpose: Specifies the maximum number of tasks (TCBs) used by the VTOC scan to perform the collection.

Syntax: VSCAN_MXTSK=*nn*
where *nn* is 2 to 30

Required: No

Default: 8

VSCAN_OINDEX=

Purpose: Specifies the prefix name of the VTOC scan collection data set. *Dyymmdd.Thhmmss* is appended to the prefix to complete the full data set name.

Syntax: VSCAN_OINDEX=xxxxxxxxxxx...

where xxxxxxxxxxx... is 1 to 28 characters, following standard data set naming conventions

Required: Yes

Default: None

VSCAN_OPRI=

Purpose: Specifies the primary allocation size in cylinders for the VTOC scan collection data set.

Syntax: VSCAN_OPRI=nnnn

where nnnn is 1 to 4369

Required: No

Default: 10

VSCAN_OSEC=

Purpose: Specifies the secondary allocation size in cylinders for the VTOC scan collection data set.

Syntax: VSCAN_OSEC=nnnn

where nnnn is 1 to 4369

Required: No

Default: 10

VSCAN_OUNIT=

Purpose: Specifies the device type of the VTOC scan collection data set.

Syntax: VSCAN_OUNIT= xxxxxxxx

where xxxxxxxx is a 1- to 8-character valid device number or name defined in your environment

Required: Yes

Default: None

VSCAN_OVOL=

Purpose: Specifies the volume serial number of the VTOC scan collection data set.

Syntax: VSCAN_OVOL=xxxxxxx

where xxxxxx is a 1- to 6- character valid volume serial number defined in your environment

Required: No

Default: None

VSCAN_TPRI=

Purpose: Specifies the primary allocation size in cylinders for the VTOC scan temporary data set.

Syntax: VSCAN_TPRI=nnnn

where nnnn is 1 to 4369

Required: No

Default: 10

VSCAN_TSEC=

Purpose: Specifies the set secondary allocation size in cylinders for the VTOC scan temporary data.

Syntax: VSCAN_TSEC=nnnn

where nnnn is 1 to 4369

Required: No

Default: 10

VSCAN_TUNIT=

Purpose: Specifies the device type for the VTOC scan temporary data set.

Syntax: VSCAN_TUNIT=xxxxxxxx

where xxxxxxxx is a 1- to 8-character valid device number or name defined in your environment

Required: Yes

Default: None

VSCAN_TVOL=

Purpose: Specifies the volume serial number for the VTOC scan temporary data set.

Syntax: VSCAN_TVOL=xxxxxx

where xxxxxx is a 1- to 6-character valid volume serial number defined in your environment

Required: No

Default: None

WTODC=

Purpose: Specifies the message descriptor code(s) to be assigned to messages written by MAINVIEW SRM. Examine the DESC keyword parameter on the WTO statement found in *MVS Supervisor Services and Macro Instructions* for an explanation of description codes.

Syntax: See WTO macro in *MVS Supervisor Services and Macro Instructions*.

Required: No

Default: None

WTORC=

Purpose: Specifies the routing code to be assigned to the message text. For more information, see the WTO macro's ROUTCDE= parameter in the *MVS Supervisor Services and Macro Instructions*.

Syntax: WTORC=*nn*

where *nn* is a number from 0 to 16. If you need more than one code, enclose them in parentheses, separated with commas.

Required: No

Default: WTORC=0

X37POOL=

Purpose: Specifies which volume will be used by X37 to determine the POOL name in EOVS processing.

Syntax: X37POOL=NEW/ORIG

Required: No

Default: X37POOL=ORIG

Pool Member Parameters

SMPOOLxx SMPOOLxx organizes DASD volumes into pools.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMPOOLxx and a brief description of INC/EXC statements used in SMPOOLxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-3 SET Statement Pool Parameter Quick Reference

Parameter	Required	Description
POOLNAME=xxxxxxx	Yes	Name to be assigned to pool
USELIMIT=nnn	No	Upper space threshold for new allocations
SGDCOLLECT=YES NO	No	Specifies whether or not a pool is processed by StorageGUARD.
SGDCOLLECTn=YES NO	No	Specifies whether a pool is processed by an alternate StorageGUARD data collector indicated by a suffix of <i>n</i> .
TYPE=xxxx	No	Device type

Table 2-4 INC/EXC Statement Pool Parameter Quick Reference

Parameter	Required	Description
ADR=xxxx	No	Device address of tape unit in pool
ADR=(xxxx,xxxx,...)	No	Multiple tape device addresses (up to 15)
VOL=xxxxxx	No	Volume serial number of device in pool
VOL=(xxxxxx,xxxxxx,...)	No	Multiple volume serial numbers (up to 15)

Parameter Explanations

ADR=

Purpose: Specifies the device addresses of tape units to be included in or excluded from the pool. MAINVIEW SRM name masking can be used.

Syntax: ADR=xxxx or ADR=(xxxx,xxxx,...)

where *xxxx* is a 4-byte character string. Up to 15 addresses can be specified by enclosing the numbers in parentheses.

Four-character device addresses were introduced with MVS/ESA 5.1. You must specify a full four-character address even if you are running an earlier release of MVS.

Required: No

Default: None

POOLNAME=

Purpose: Specifies the name of the pool. (Note that this definition is independent of the MVSCP.) The pool names specified need not be defined to MVS as esoteric device names.

Syntax: POOLNAME=xxxxxxx

where xxxxxxx is a 1–8 character string.

In addition to the 1–8 character string, the following may be specified for StorageGUARD to derive the pool name dynamically from the device being processed:

POOLNAME=&xxxxxxx/(start,end)

where &xxxxxxx is one of the following:

&VOL
&UNIT
&MNTYPE
&STOGROUP
&STORGRP

Start and end are used to specify which characters will be used in the pool name. If start and end are not used, all characters will be used. For example, if the volume serial number is ABC123 and POOLNAME=&VOL is specified, the pool name will be ABC123. If POOLNAME=&VOL(1,3) is specified, the pool name will be ABC. If POOLNAME=&VOL(3,6) is specified, the pool name will be C123.

Use of variable-named pools applies only to StorageGUARD. When variable-named pools are used, the default for SGDCOLLECT is YES. If NO is specified on SGDCOLLECT, it is ignored.

Required: Yes

Default: None

SGDCOLLECT=

Purpose: Specifies whether a pool is processed by StorageGUARD.

Syntax: SGDCOLLECT=*YES/NO*

Required: No

Default: No

SGDCOLLECT n =

Purpose: Specifies whether a pool is processed by an alternate StorageGUARD data collector. The alternate data collector to be used is identified by the suffix n .

Syntax: SGDCOLLECT n =*YES/NO*

Required: No

Default: No

TYPE=

Purpose: Specifies the type of device.

Syntax: TYPE=*xxxxx*

where *xxxxx* is one of the following values: DASD, 3420, 3480, 3490.

Types 3420/3480/3490 are used to define tape pools for the TAPEPOOL function.

Required: No

Default: TYPE=DASD

USELIMIT=

Purpose: Specifies an upper space limit for DASD volumes in a pool. MAINVIEW SRM attempts to prevent allocation of a new data set to a given DASD volume if that allocation would cause the volume USELIMIT threshold to be exceeded. This threshold is provided to ensure sufficient space on a volume for existing data sets to be extended with secondary extents. The USELIMIT parameter is similar to the high allocation threshold provided by DFSMS.

This parameter does not apply to tape devices or to DFSMS-managed DASD volumes.

The USELIMIT parameter on a pool will not prevent a pool assignment, even if a volume within the USELIMIT percentage cannot be found. In this case, the last volume found that would satisfy the primary allocation will be selected.

Note that USELIMIT applies only to primary allocation processing; during allocation of secondary extents, the USELIMIT is not enforced. If primary allocations are consistently too small for all data sets on a volume, thus requiring extensive secondary allocations, it is still possible to exceed the USELIMIT and fill a volume.

USELIMIT will only be applied by DASDPOOL when the VOLSEL parameter has been specified. USELIMIT is also applied by the SPACVOLA function.

Syntax: USELIMIT=*nnn*

where *nnn* is a number in the range 1–100.

Required: No

Default: None

VOL=

Purpose: Specifies the volume serial numbers of DASD devices to be included in or excluded from the pool. MAINVIEW SRM name masking can be used.

Syntax: VOL=*xxxxxx* or VOL=(*xxxxxx,xxxxxx,...*)

where *xxxxxx* is a 1–8 character string. Up to 15 volumes can be specified by enclosing the numbers in parentheses.

Required: No

Default: None

SMS Subpool Member Parameters

SMSPOLxx Organizes SMS-managed DASD volumes into subpools; SMS subpools are only used by EasyPOOL.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMSPOLxx and a brief description of INC/EXC statements used in SMSPOLxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-5 SET Statement SMS Pool Parameters

Parameter	Required	Description
POOLNAME=xxxxxxx	Yes	Name to be assigned to SMS subpool

Table 2-6 INC/EXC Statement SMS Pool Parameters

Parameter	Required	Description
VOL=xxxxxx	No	Volume serial number of device in subpool
VOL=(xxxxxx,xxxxxx,...)	No	Multiple volume serial numbers (up to 15)

Parameter Explanations

POOLNAME=

Purpose: Specifies the name of the SMS subpool. (Note that this definition is independent of the MVSCP.) The SMS subpool names specified need not be defined to MVS as esoteric device names.

Syntax: POOLNAME=xxxxxxx
where xxxxxxx is a 1–8 character string.

Required: Yes

Default: None

VOL=

Purpose: Specifies the volume serial numbers of SMS-managed DASD devices to be included in or excluded from the subpool. MAINVIEW SRM name masking can be used.

Syntax: VOL=xxxxxx or VOL=(xxxxxx,xxxxxx,...)

where xxxxxx is a 1–6 character string. Up to 15 volumes can be specified by enclosing the numbers in parentheses.

Required: No

Default: None

Calendar Member Parameters

SMCAL\$xx SMCAL\$xx defines non-working days for DFHSM migration processing and other date-related processing.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMCAL\$xx and a brief description of INC/EXC statements used in SMCAL\$xx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-7 SET Statement Calendar Parameters

Parameter	Required	Description
YEAR=\$nnnn	Yes	Year to which the following days apply

Table 2-8 INC/EXC Statement Calendar Parameters

Parameter	Required	Description
FREE=\$nn.nn-\$nn.nn	No	From–to range of non-working (free) days
MON=\$F/W	No	Day of week
TUE=\$F/W	No	Day of week
WED=\$F/W	No	Day of week
THU=\$F/W	No	Day of week
FRI=\$F/W	No	Day of week
SAT=\$F/W	No	Day of week
SUN=\$F/W	No	Day of week

Parameter Explanations**YEAR=**

Purpose: Specifies the year being defined.

Syntax: YEAR=*nnnn*

where *nnnn* is a four-digit year in the range 1900–2100.

Required: Yes

Default: None

FREE=

Purpose: Specifies a single date or a date range that represents non-working days (days that are not considered as usage days by DFHSM).

Note that the DATEFMT parameter in SMMSYSxx does not apply to date specifications in SMCALSxx.

Syntax: FREE=*nn.nn-nn.nn*

where *nn.nn* is a date specification of the form dd.mm, where dd and mm both are two-digit numbers. For example:

07.12	December 7
15.02	February 15
01.07-05.07	July 1–5

Required: No

Default: None

MON-SUN=

Purpose: Specifies that a specific day of the week is either a non-working (free) day or a working (usage) day.

Syntax: MON=F/W

where F identifies a non-working day, and W identifies a working day.

Required: No

Default: None

Variable Member Parameters

SMVARSxx SMVARSxx defines variables to contain MAINVIEW SRM selection parameters. These variables can be included in filter and rules lists.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMVARSxx and a brief description of INC/EXC statements used in SMVARSxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-9 SET Statement Variable Parameters

Parameter	Required	Description
VARIABLE=xxxxxxxx	Yes	Name assigned to variable

Table 2-10 INC/EXC Statement Variable Parameters

Parameter	Required	Description
VALUE=xxxxxxxxxxxxxx	Yes	Any values valid in selection parameters

SMVARSxx is an optional member.

Parameter Explanations

VARIABLE=

Purpose: Specifies the name of the variable.

Syntax: VARIABLE=xxxxxxxx

where xxxxxxxx is a 1- to 30-character string.

Required: Yes

Default: None

VALUE=

Purpose: Specifies a value for the variable.

Syntax: VALUE=xxxxxxxxxxxxxx

where xxxxxxxxxxxxxx is any character string, with no embedded blanks.

Required: Yes—at least one value must be declared for a variable.

Default: None

Function Member Parameters

SMFUNCxx SMFUNCxx defines and activates functions. A function must have an entry in SMFUNCxx to be available to the executing MAINVIEW SRM subsystem. A function's parameters include specification of a filter list member and a rule list member (if required). These two PARMLIB members give tremendous flexibility in applying a function's processing to data resources.

Subordinate Members

SMFLSTxx, SMRLSTxx

Parameter Quick Reference

The following table provides a brief description of SET statements used in SMVARSxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-11 SET Statement Function Parameters

Parameter	Required	Description
NAME=xxxxxxx	Yes	MAINVIEW SRM-assigned name of the function
FLST=xx	No	Suffix of filter list member SMFLSTxx
RLST=xx	No	Suffix of rule list member SMRLSTxx
ACTIVE=YES/NO	Yes	Status of the function
MSG=I/W/E/S/N	Yes	Level of messages to be generated
SMF=I/W/E/S/N	Yes	Level of messages to be written to SMF
TRACE=xxxxxxx	No	Jobname of traced MAINVIEW SRM actions
DESC='xxxxxxxxxx xxxxxxxx'	No	Description of function

SMFUNCxx is required. MAINVIEW SRM does nothing without defined function parameters.

Parameter Explanations**NAME=**

Purpose: Specifies the name of the function. Function names are assigned within MAINVIEW SRM code.

Syntax: NAME=xxxxxxx
where xxxxxxx is a 1–8 character string of a MAINVIEW SRM function assigned by BMC Software.

Required: Yes

Default: None

FLST=

Purpose: Specifies the suffix of the filter list PARMLIB member (SMFLST xx) for this function. The filter list allows selection of resources that are affected by the function. If no filter list member is specified, no resources are selected for the function.

Syntax: FLST= xx
where xx is any two-character string. A single character is not allowed.

Required: No

Default: None

RLST=

Purpose: Specifies the suffix of the rule list PARMLIB member (SMRLST xx) for this function. The rule list allows specification of how the function is applied to selected resources. If no rule list is specified, the function default processing is applied to all resources selected by the filter list parameters. However, if there is no default processing by the function (that is, an action parameter is required for the function to have affect), a rule list must be specified (SET and INC parameters) for any processing to take place.

Note that the following functions do not use a rule list: FORCECAT, MODDELET, OPENEMPT, SMSMCREN, SUPJSCAT, SUPVOLRF, and TAPEDEFR.

Syntax: RLST= xx
where xx is any two-character string. A single character is not allowed.

Required: Yes, if the function uses a rule list; otherwise, no.

Default: None

ACTIVE=

Purpose: Specifies the status of the function. If **ACTIVE=NO** is specified, the function has no effect, regardless of any specifications in the filter or rule list members. **ACTIVE=YES** must be specified for the function to provide any MAINVIEW SRM services.

Syntax: **ACTIVE=YES/NO**

Required: Yes

Default: None

MSG=

Purpose: Specifies the default message generation option for the function. Information and error messages can be produced, or all messages can be suppressed. Note that the **MSG** option on the filter list **SET** command overrides this option for selected resources.

Syntax: **MSG=I/W/E/S/N**

where

I = Information and error messages

W = Warning messages

E = Error messages only

S = Severe error messages

N = No messages

Required: Yes

Default: None

SMF=

Purpose: Specifies the SMF message generation option for the function. Information and error messages can be written to the SMF data set, or all messages can be omitted from the SMF data set. Note that the **SMF** option on the filter list **SET** command overrides this option for selected resources.

Syntax: **SMF=I/W/E/S/N**

where

I = Information and error messages

W= Warning messages

E = Error messages only

S = Severe error messages

N = No messages

Required: Yes

Default: None

TRACE=

Purpose: Specifies that, for the identified job, all filter and rule list processing for the function is to be traced by writing MAINVIEW SRM messages.

Syntax: TRACE=xxxxxxx

where xxxxxxx is a 1–8 character jobname (including TSO session ID or started task name). Note that the name of the job to be traced must match this parameter value exactly; *name masking does not apply to this parameter.*

Required: No

Default: None

DESC=

Purpose: Specifies a short description for the function. This description appears in the ISPF panel(s) where the function is displayed.

Syntax: DESC='xxxxxxx'

where xxxxxxx is a quoted string up to 46 characters long.

Required: Yes

Default: None

Diagnostic Member Parameters

SMDIAGxx SMDIAGxx aids in diagnosing problems in MAINVIEW SRM modules.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMDIAGxx and brief description of INC/EXC statements used in SMDIAGxx. Since this member is used only when directed to do so by BMC Software Customer Support, the parameters are not described in detail.

Table 2-12 SET Statement Diagnostic Parameters

Parameter	Required	Description
ABEND=YES/NO	No	Forces S0C3 abend when a particular module is entered
DEBUG=YES/NO	No	Optional debugging information from a MAINVIEW SRM module
DUMP=YES/NO	No	Issues SDUMP if program abend occurs
IGNORE=YES/NO	No	Skip this function
MODTRC=YES/NO	No	MAINVIEW SRM module trace
TRACE=YES/NO	No	FLST/RLST trace output

Table 2-13 INC/EXC Statement Diagnostic Parameters

Parameter	Description
FUNCTION=xxxxxxx	A valid MAINVIEW SRM function name (up to eight characters)
JOB=xxxxxxx	A job name (up to eight characters)
MODULE=xxxxxxx	A valid MAINVIEW SRM module name (up to eight characters)
PGM=xxxxxxx	A valid MAINVIEW SRM program name (up to eight characters)
STEP=xxxxxxx	A step name (up to eight characters)

Event Member Parameters

SMEVNTxx SMEVNTxx defines how event notices are to be generated.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMEVNTxx, and a brief description of INC/EXC statements used in SMEVNTxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

Table 2-14 SET Statement Event Parameters

Parameter	Required	Description
DEST=ETS/AOO/(ETS,AOO)	Yes	Routes the event
EVENTID=xxxxx	Yes	Identifies the event
MODE=A/I	Yes	Sets the event to active or inactive
OVERRIDE=Y/N	No	Specifies that default system event parameters are to be replaced
SEV=x	No	Indicates the urgency of the event
TEXT='xxxxx'	Yes	Specifies the text of the event message

Table 2-15 INC/EXC Statement Event Parameters

Parameter	Required	Description
EVENTID=xxxxx	Yes	Specifies the identifier assigned to the user event in SMEVNTxx.

SMEVNTxx is an optional member.

Parameter Explanations

DEST=

Purpose: Routes an event to either a for Product/SeriesName^{®™} Explorer console or an AutoOperator console for central viewing and management.

Syntax: DEST=*ETS/AOO/(ETS,AOO)*

where

ETS routes the event to the for Product/SeriesName^{®™} Explorer console specified on the ETS_ID parameter in SMMSYSxx.

AOO routes the event to all AutoOperator subsystem consoles specified on the AOO_SUBSYS parameter(s) in SMMSYSxx.

(AOO,ETS) routes an event message to both AutoOperator and for Product/SeriesName^{®™} Explorer consoles.

Note: Events routed to AutoOperator may be responded to automatically on receipt if you create AutoOperator rules for them.

Required: Yes

Default: None

EVENTID=

Purpose: Specifies an event identifier. The value specified on this parameter is appended to the characters SVW to form an eight-character header for an event message. If a severity indicator is also specified for an event, the indicator will be appended to SVWxxxxx to form a nine-character message header.

Syntax: EVENTID=*xxxxx*

where *xxxxx* is a 5 character string. User-defined events may not begin with the character I, which is reserved for system messages. It is recommended that user-defined events begin with the character U.

For user events, this value must match the value assigned on the SET statement EVENTID= parameter in the filter list or rule list member that generates the event.

For system events, this value must match the system event ID assigned to the event.

Required: Yes

Default: None

MODE=

Purpose: Sets an event to active or inactive to turn event generation off or on. If the event mode is inactive, event generation will be bypassed when the function that generates it is processed.

Syntax: `MODE=A/I`

Required: Yes

Default: System events are defined as inactive. If you want to activate a system event, you must change the value on this parameter to `MODE=A`.

Note: Refreshing an event member reactivates an event.

OVERRIDE=

Purpose: Allows you to replace default values for system events. When `OVERRIDE=Y` is specified in an entry, the values you specify on the other parameters in the entry replace the system event default values.

Syntax: `OVERRIDE=Y/N`

Required: No

Default: None

SEV=

Purpose: Indicates the urgency of an event. The severity indicator is appended to the end of `SVWxxxx` to form a nine-character header for an event message.

Syntax: `SEV=x`

where *x* is a single alpha or numeric character. It is recommended that you use the following characters.

- I (informational messages)
- W (warning messages)
- E (error messages)
- S (serious error messages)

Required: No

Default: None

TEXT=

Purpose: Specifies the text of the event message.

Syntax: TEXT='xxxxx'

where the text is enclosed in single quotation marks (') and can contain variables from the function generating the event. The total length of the text can be a maximum of 255 bytes once the variables are expanded. If the text is greater than 255 bytes after variable expansion, it is truncated word by word until it is 255 bytes or less.

Variables used on the TEXT= parameter must be based on INC/EXC statement parameters for functions that generate events. A text variable consists of an ampersand (&) followed by an INC/EXC statement parameter name valid for the function that generates the event. When the event is generated, the value of the parameter is passed to the event and replaces the parameter name in the text.

Note: Parameters used as text variables are restricted to INC/EXC statement parameters. You may not use SET statement parameter names as variables.

To continue a line of text to the next line, place a non-blank character in column 72 of the line to be continued. The first character in the next line is appended to the last character in the previous line. If you need a blank space to appear in the text following the character in column 72, place a quotation mark (') in the first position of the new line and a space after the quotation mark.

Required: Yes

Default: None

VTOC Scan Facility Parameters

SMVSCFxx SMVSCFxx defines the VTOC Scan Facility filter criteria.

Parameter Quick Reference

The following tables provide a brief description of SET statements used in SMVSCFxx and a brief description of INC/EXC statements used in SMVSCFxx. Detailed descriptions of each parameter are listed in alphabetical order after the tables.

SMVSCFxx is an optional member.

Table 2-16 SET Statement VTOC Scan Facility Parameters

Parameter	Required	Description
RECORD_TYPE=x	N	Specifies whether to generate data set or volume records
DSN_MASK=xxxxxxxxx	N	Specifies the data set name or mask
DSN_TYPE=x	N	Specifies the data set type
MRG_CATINFO=YES/NO	N	Specifies whether to include catalog information in the collected statistics
MRG_SGCINFO=YES/NO	N	Specifies whether to include SG-Control data in the collected statistics
VOLUME=xxxxxxx	N	Specifies the volser or mask
START_UNIT=nnnn	N	Specifies the starting unit address range
END_UNIT=nnnn	N	Specifies the ending unit address range
MNT_STATUS=xx	N	Specifies the volume mount status
SMS_STATE=xxxxxxxxx	N	Specifies the volume's SMS status
SMS_GROUP=	N	Specifies the SMS group name or mask

Table 2-17 INC/EXC Statement SMVSCFxx Parameter

Parameter	Required	Description
FORSYSID=	No	Specifies user-defined system IDs that can be included or excluded in a sysplex environment

SMVSCFxx is an optional member.

Parameter Explanations**DSN_MASK=**

Purpose: Specifies the data set name or mask.

Syntax: DSN_MASK=xxxxxxxxxx

where xxxxxxxx... is the data set name or data set filter. A forward slash specifies all data set names.

Required: No

Default: DSN_MASK=/

DSN_TYPE=

Purpose: Specifies the data set type.

Syntax: DSN_TYPE=x

where *x* is
A = All
V = VSAM
N = NONVSAM

Required: No

Default: DSN_TYPE=A

END_UNIT=

Purpose: Specifies the ending unit address range.

Syntax: END_UNIT=xxxx

where xxxx is 4 characters

Required: No

Default: END_UNIT=FFFF

MNT_STATUS=

Purpose: Specifies the mount status of the volume.

Syntax: MNT_STATUS=*x*

where *x* is

A = All

P = Public

V = Private

S = Storage

Required: No

Default: MNT_STATUS=A

MRG_CATINFO=

Purpose: Specifies whether to include catalog information in the collection statistics.

Syntax: MRG_CATINFO=*YES/NO*

Required: No

Default: MRG_CATINFO=YES

MRG_SGCINFO=

Purpose: Specifies whether to include SG-Control data in the collected statistics.

Syntax: MRG_SGCINFO=*YES/NO*

Required: No

Default: MRG_SGCINFO=YES

RECORD_TYPE=

Purpose: Specifies the whether to generate the report by data set or volume.

Syntax: RECORD_TYPE=*x*

where *x* is

D = data set and volume statistics records

V = volume statistical records

Required: No

Default: RECORD_TYPE=V

SMS_GROUP=

Purpose: Specifies the volume's SMS group name or mask.

Syntax: SMS_GROUP=xxxxxxxx

where xxxxxxxx is an SMS storage group name or filter. A forward slash specifies all SMS storage group names.

Required: No

Default: SMS_GROUP=/

SMS_STATE=

Purpose: Specifies the SMS status for the volumes.

Syntax: SMS_STATE=x

where x is

A= All
I= Initial
M = Managed
U = Unmanaged

Required: No

Default: SMS_STATE=A

START_UNIT=

Purpose: Specifies the starting unit address range.

Syntax: START=xxxx

where xxxx is 4 characters

Required: No

Default: START=0000

VOLUME=

Purpose: Specifies the volume serial number or volume serial number filter.

Syntax: VOLUME=xxxxxx

where xxxxxx is 1- to 6-character valid volume serial number defined in your environment. A forward slash specifies all volumes.

Required: No

Default: VOLUME=/

Glossary

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

Since this glossary pertains to BMC Software-related products, some of the terms defined may 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.

action	Defined operation, such as modifying a MAINVIEW window, that is performed in response to a command. <i>See</i> object.
active window	Any MAINVIEW window in which data can be refreshed. <i>See</i> 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. <i>See</i> 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. <i>See</i> alternate window.
Alternate Access	<i>See</i> MAINVIEW Alternate Access.
alternate form	View requested through the FORM command that changes the format of a previously displayed view to show related information. <i>See also</i> 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. <i>Contrast with</i> current window. <i>See</i> 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. <i>See</i> 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	<i>See</i> 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.
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	<i>See</i> 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: <ul style="list-style-type: none"> • MAINVIEW AutoOPERATOR • MAINVIEW for CICS • MAINVIEW for DB2 • MAINVIEW for DBCTL • MAINVIEW for IMS Online • MAINVIEW for MQSeries (formerly Command MQ for S/390) • MAINVIEW for VTAM • MAINVIEW VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)
BBPARM	<i>See</i> parameter library.

BBPROC	<i>See</i> procedure library.
BBPROF	<i>See</i> profile library.
BBSAMP	<i>See</i> sample library.
BBV	<i>See</i> 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.
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 image requires a separate CAS. Cross-system communication is established through the CAS using VTAM and XCF communication links.
CFMON	<i>See</i> coupling facility monitoring.
chart	Display format for graphical data. <i>See also</i> 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. <i>See</i> 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.

CMRSTAT

MAINVIEW for CICS data set that stores both CICS operational statistic records, at 5-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 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.

CPO

Customized Product Offering. Delivery and installation technique that allows any combination of BMC Software SMP/E-maintainable products to be distributed on a product tape to a customer and installed quickly. The CPO product tape contains libraries required for product customization and execution, plus SMP distribution libraries and data sets needed for application of SMP maintenance.

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, 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.

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.

DASD ADVISOR

An interactive software tool that diagnoses DASD performance problems and makes recommendations to reduce overall service time. This tool measures and reports on the operational performance of IBM and IBM-compatible devices.

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 services, OS/390 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. <i>See also</i> 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. <i>See also</i> collection interval, sample cycle, total mode.
DFSMS	(Data Facility Storage Management System) Data management, backup, and HSM software from IBM for OS/390 mainframes.
DMR	<i>See</i> MAINVIEW for DB2.
DMS	(Data Management System) <i>See</i> CA-Disk.
DMS2HSM	Component of MAINVIEW SRM that facilitates the conversion of CA-Disk, formerly known as DMS, to HSM.
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.
EasyHSM	Component of MAINVIEW SRM that provides online monitoring and reporting to help storage managers use DFHSM efficiently.
EasyPOOL	Component of MAINVIEW SRM 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.
EasySMS	Component of MAINVIEW SRM 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.
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 <i>field help</i> .
Enterprise Storage Automation	Component of MAINVIEW SRM that integrates powerful event management technology and storage monitoring technology to provide significant storage automation capabilities and solutions. Storage occurrences are defined to generate events in the form of messages that provide an early warning system for storage problems and are routed to MAINVIEW AutoOPERATOR to be viewed.

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. <i>See also</i> 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. <i>Contrast with</i> data collector.
extractor interval	<i>See</i> collection interval.
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. <i>See also</i> 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.
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. <i>See also</i> query, view.
full-screen mode	Display of a MAINVIEW product application or service on the entire screen. There is no window information line. <i>Contrast with</i> windows mode.

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. <i>See also</i> chart.
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. <i>Contrast with</i> 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. <i>See</i> 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	<p>(1) Preset field in a view or an EXPAND line on a display that permits you to</p> <ul style="list-style-type: none"> • Access cursor-sensitive help • Issue commands • Link to another view or display <p>The transfer can be either within a single product or to a related display/view in a different MAINVIEW product. Generally, hyperlinked fields are highlighted. (2) Cursor-activated short path from a topic or term in online help to related information. <i>See also</i> fast path.</p>
Image log	<p>Collection of screen-display records. Image logs may be created for both the BBI-SS PAS and the BBI terminal session (TS).</p> <p>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.</p> <p>The TS Image log is a single data set that wraps around when full.</p>

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 detailed (one event, one record) or summarized (more than one event, one record). A detailed IRUF is created by processing the IMS system log through a program called IMFLEEDIT. A summarized IRUF is created by processing one or more detailed IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program.

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.

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).
- MAINVIEW** BMC Software integrated systems management architecture.
- MAINVIEW Alarm Manager (MV ALARM)**
In conjunction with other MAINVIEW products, notifies you when an exception 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 shows exceptions for all MAINVIEW performance monitors within your OS/390 enterprise.
- MAINVIEW Alternate Access**
Enables MAINVIEW products to be used without TSO by providing access through EXCP and VTAM interfaces.
- 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 Desktop** Version of the MAINVIEW window interface designed to run on OS/2 and Windows workstations.
- 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 (MVDBC)

Product 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 MQSeries (formerly known as Command MQ for S/390)

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 (formerly 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 Selection Menu

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

MAINVIEW Storage Resource Monitor (SRM)

Suite of products that assist in all phases of OS/390 storage management. MAINVIEW SRM consists of components that perform automation, reporting, trend analysis, and error correction for storage management in OS/390.

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; e.g., reports may be for a single target, an OS/390 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.

MV MANAGER for CICS

See MAINVIEW for CICS.

MV MANAGER for DB2

See MAINVIEW for DB2.

MV MANAGER for MVS

See MAINVIEW for OS/390.

MVALARM

See MAINVIEW Alarm Manager.

MVCICS

See MAINVIEW for CICS.

MVDB2

See MAINVIEW for DB2.

MVDBC

See MAINVIEW for DBCTL.

MVIMS

See MAINVIEW for IMS.

MVMQ

See MAINVIEW for MQSeries.

MVMVS

See MAINVIEW for OS/390.

MVSRM

See MAINVIEW Storage Resource Manager (SRM).

MVSRMHSM	<i>See</i> EasyHSM.
MVSRMSGC	<i>See</i> SG-Control.
MVSRMSGD	<i>See</i> StorageGUARD.
MVSRMSGP	<i>See</i> StorageGUARD.
MVUSS	<i>See</i> MAINVIEW for UNIX System Services.
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.
MVVP	<i>See</i> MAINVIEW VistaPoint.
MVVTAM	<i>See</i> MAINVIEW for VTAM.
MVWEB	<i>See</i> MAINVIEW for WebSphere.
nested help	Multiple layers of help pop-up windows. Each successive layer is accessed by hyperlinking from the previous layer.
object	<p>Anything you can manipulate as a single unit. MAINVIEW objects can be any of the following: product, secondary window, view, row, column, or field.</p> <p>You can issue an action against an object by issuing a line command in the line command column to the left of the object. <i>See</i> action.</p>
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 data collectors, including the CMF MONITOR Extractor. Used by MAINVIEW for OS/390, MAINVIEW for USS, and CMF MONITOR products. <i>See</i> PAS.
parameter library	<p>Data set comprised of members containing parameters for specific MAINVIEW products or a support component. There can be several versions:</p> <ul style="list-style-type: none"> • The distributed parameter library, called BBPARM • 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

MVS/SP-defined collection of address spaces. *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)

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.

pop-up window

Window containing help information that, when active, overlays part of the window area. A pop-up panel is displayed when you issue the HELP command.

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 IEAIPSxx member.

procedure library

Data set comprised of members containing 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 comprised of members containing 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.

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.

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.
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 comprised of members each of which contains one of the following: <ul style="list-style-type: none"> • 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 <p>There can be several versions:</p> <ul style="list-style-type: none"> • The distributed sample library, called BBSAMP • A site-specific sample library or libraries <p>These can be</p> <ul style="list-style-type: none"> -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	<i>See</i> 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. <i>See</i> 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	OS/390- or MAINVIEW for OS/390-defined collection of address spaces. If you are running MVS 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, OS/390 creates a performance group workload instead of a service class. <i>See</i> 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. There are no OS/390-related measures of service for started task workloads.
service point	Specification, to MAINVIEW, of the services required to enable a specific product. Services may be actions, selectors, or views. Each target (e.g., 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. <i>See also</i> 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	Component of MAINVIEW SRM that provides early warning notification of storage anomalies and automated responses to those anomalies based on conditions in the storage subsystem.
SG-Control	Component of MAINVIEW SRM that provides real-time monitoring, budgeting, and control of DASD space utilization.

single system image (SSI)

Feature of the MAINVIEW window environment architecture that allows you to 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 images.

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).

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

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

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 system problems as they occur. Accessible from the OS/390 Performance and Control Main Menu. Note that this is also available as a stand-alone product MAINVIEW SYSPROG Services.

system resource *See* object.

target	Entity monitored by one or more MAINVIEW products, such as an OS/390 image, IMS or DB2 subsystem, CICS region, or related workloads across systems. <i>See</i> context, scope, SSI context.
target context	Single target/product combination. <i>See</i> 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. <i>See also</i> service request block.
TCB	<i>See</i> 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	<i>See</i> trace log directory.
threshold	Specified value used to determine whether the data in a field meets specific criteria.
TLDS	<i>See</i> 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. <i>See also</i> 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	<i>See</i> terminal session.
TSO workload	Workload that consists of address spaces running TSO sessions.
UAS	<i>See</i> user address space.
UBBPARM	<i>See</i> parameter library.
UBBPROC	<i>See</i> procedure library.
UBBSAMP	<i>See</i> sample library.
user address space	Runs a MAINVIEW terminal session (TS) in TSO, VTAM, or EXCP mode.
User BBPROF	<i>See</i> profile library.
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. <i>See also</i> 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).
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. <i>See</i> 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, 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 may include measures of performance such as response times and batch turnaround times.

Index

A

ABEND 10-2
action parameters 2-5, 2-6, 3-2
 defined 3-2
 relationships 3-3
actions
 HELP 1-11
activate
 dynamic 4-3
activating system software into MVS 4-2
ACTIVE 6-3, B-77, B-79
ADR 7-3, B-69
ALLTAPE B-60
AOO_SUBSYS B-2, B-15
asterisk
 comment line 3-3
AUTOPROC B-2, B-15

B

BBI-3 architecture 2-10
BBI3_SSID B-2, B-15
BBPARM 2-2
BBSAMP members
 SGPPROC B-35
BCDSn B-2, B-16
BLKINPUT B-2
BLKOLDSR B-2, B-16

C

cached devices B-19, B-57
CAL B-2
calendar
 parameter B-17
 SMCAL parameter B-74
CANCEL 4-18
catalog name B-62
CHECK B-2, B-17
command
 issuing 1-13
 in COMMAND field 1-20
 using menus 1-14
 with hyperlinks 1-13
 TIME 1-29
command line
 HELP 1-12
commands
 entering 4-9
 line action 1-11
 MAINVIEW SRM
 primary action 1-11
 MAINVIEW SRM specific 1-20
 primary action 1-11
common command structure 4-3
communicating with MAINVIEW SRM
 components 4-3
comparison operators 3-4
console
 commands
 DISPLAY 4-19
 REFRESH 4-19, 4-20

- SET 4-19, 4-21
- VSCAN 4-19, 4-22
 - interface 4-18
- CONTEXT command 1-22
- controlling MAINVIEW SRM components 4-3
- controlling secondary reduction B-29
- conventions
 - document xviii
 - syntax statements xix
 - typographical xviii
- CRITLIST B-2, B-17
- customization
 - HELP text 1-10
 - view 1-10

D

- DADSMEX B-2, B-17
- DASDGENR B-2, B-18
- DASDPOOL B-72
- data entry panel 1-17
- data input
 - filtering 1-17
- data sets
 - size 2-4, 5-20
- DATACLAS 2-6
- DATEFMT B-2, B-18
- DCTYPE B-2, B-19
- DD 2-7
- DD statement
 - PROIGN
 - override 10-2
 - suppress jobstep activity B-27
- DD statements
 - PRO\$D\$M\$ 10-9
 - PRODUMP 10-9
 - PROIGN 10-9, B-27
 - PROMTRCE 10-9
 - PROTRACE 10-9
- DEBUG 10-2
- Defining 6-1
- defining pools B-69, B-73
- definition member
 - parameters 2-5
 - SMCAL\$xx 9-1
- definition members 2-2
 - SMACMDxx 2-3

- SMCAL\$xx 2-2
- SMDAIGxx 2-2
- SMEVNTxx 2-3
- SMFUNCxx 2-2, 6-2
- SMPOOLxx 2-2, 7-2
- SMSPOLxx 2-2
- SMVARSxx 2-2, 8-1
- SMVSCFLxx 2-3
- DESC 6-3, B-77, B-80
- DEST B-82, B-83
- devices
 - using SSI to check 1-24
- DFREORGPRC B-2, B-19
- DIAG B-2, B-19
 - IGNORE parameter 10-2
- DIAGMSDD 10-9, B-2, B-20
- diagnostic parameters 10-1, B-81
- DISPLAY 4-19, B-2, B-20
- DMYUNIT B-3, B-20
- document conventions xviii
- documentation
 - related xvi
- DP_RENAME B-3, B-21
- dualcopy B-19
- DUMP 10-2
- DUMP command 4-11
- DUMPDD 10-9, B-3, B-21
- dynamic activate 4-3
 - overview 4-2
- dynamic fields 1-22

E

- Easy Menu 1-14
- Easy Menus
 - described 1-14
- Easy menus
 - EZM390 1-27, 5-2, 5-21, 12-2
- EasyHSM 4-4
- EasyPOOL 4-4
- EasySMS 4-4
- entering commands 4-9
- Enterprise Storage Automation 4-4
- ETS_ID B-21
- EVENTID B-82
- EVNT B-3, B-21
- EXC statements 2-8

execution JCL description 4-16
EXPDT 2-6
EZ menus 1-14
EZcmd menu 1-19
EZcmd menus 1-19

F

fastwrite B-19
FDRIAM B-3, B-22
field
 dynamic 1-22
 HELP 1-11
field-to-field compares 3-9
filter list parameters 11-2
 examples 11-8
filter lists
 defined 2-7
filtering data 1-20
 from menus 1-17
FLST 6-3, B-77, B-78
FORCE
 allow from console 4-17
 command 4-12
 parm on SVOS 4-17
FORCECAT 11-6
FORPLEXNAME 2-11, B-14, B-22
FORSMFID 2-11, B-14, B-22
FORSYSID 2-11, B-14, B-22
FRAGCNTL 11-6
FREE 9-2, B-74, B-75
FUNC B-3
FUNCTION 10-3, B-81
functions
 define and activate 6-1
 definition examples 6-4

G

general syntax rules 3-1
global parameters
 reference B-1
global parameters - SMMSYSxx B-1

H

HELP
 actions 1-11
 command line 1-12
 field 1-11
 line action commands 1-11
 primary action commands 1-11
 text customization 1-10
 view 1-11
help
 field 1-11
 in full-screen mode 1-11
 in windows 1-11
hierarchal structure of PDS members 2-1
HISTDAYS B-3, B-23
HLOGAUTH B-3, B-24
HLOGAUTM B-3, B-24
HLOGCOLL B-3, B-25
HLOGINDX B-3, B-25, B-26
HLOGPRIM B-3
HLOGTASK B-3, B-26
HLOGUNIT B-3
HLOGYDSN B-3, B-26
HSMACTID B-3, B-26
hyperlink
 establishing 1-13
 using 1-13

I

IGNORE 10-2
 parameter 10-2
IGNOREDD 10-9, B-3, B-27
INC statements 2-8
ISPF interface
 console 4-18

J

JCLEXT B-3, B-27
JCLUREQ B-4, B-28
JOB 10-3, B-81
JOBACCT 2-7

L

LIMIT 2-8
limiting secondary reduction B-29
line action commands
 HELP 1-11

M

MAINVIEW SRM
 starting and stopping 4-4
MAINVIEW SRM component identifier
 SVALLOC 4-4
 SVESA 4-4
 SVHSM 4-4
 SVOS 4-4
 SVSGA 4-4
 SVSGD 4-4
 SVSGP 4-4
MAINVIEW SRM Parameters 2-5
masking syntax 3-5
master system definition member
 SMMSYSxx 2-2
master system member parameters B-1
MAXSIZE 2-7
MAXVOL B-4, B-28
MCDSn B-4
menu
 data filtering 1-17
 easy 1-14
 EZ 1-14
 EZcmd 1-19
 EZSRM 1-14
 pop-up 1-18
 using 1-14
messages
 written to the SMF data set 13-1
MODDELET 11-6
MODE 11-2, 11-3
 parameter 11-3
MODIFY
 commands 4-10
MODTRC 10-2
MODTRCDD 10-9, B-4, B-29
MODULE 10-3, B-81
monitor
 performance views 12-7

pool views 12-3
SMS pool views 12-7
SMS storage group views 12-6
volume performance statistics view 12-10
volume performance summary views 12-9
volume performance views 12-4
volume space view 12-6
volumes in a pool view 12-5

MON-SUN 9-2, B-74, B-75
MPOOL 12-3
MPOOLVOL 12-5
MREDUCE B-4, B-29
MSG 2-6, 2-8, 6-3, 11-2, 11-4, B-77, B-79
MSGID B-4, B-30
MSGLVL B-4, B-30
MSGPREF B-4, B-30
MSMSG 12-6
MSMSP 12-7
multiple subsystems
 support 4-3
MVOLPAV 12-10
MVOLPER 12-7
MVOLPERS 12-9
MVOLSPC 12-4
MVOLSSPC 12-6

N

NAME 6-3, B-77, B-78
name masking 2-4, 2-8, 3-5
naming
 conventions for views 1-11
navigation 1-14
NOCATDYN B-4, B-31
NOCATPFX B-4, B-31
NOCATPRG B-4, B-31
NOCATSEC B-4, B-32
NOCATSMS B-4, B-32
NOCATVOL B-4, B-33
NOCATWHEN B-4, B-33
NOVERIFY
 parm on SVOS 4-17

O

OCDS B-5, B-34

OPENEMPT 11-6
OPMHLQ B-5, B-34
ORIGDATA B-5, B-34
override PROIGN DD statement
 IGNORE 10-2

P

parameter description 4-16
parameter relationships 3-3
parameters 6-2
 action 3-2
 calendar settings, SMCALS B-74
 global parameter SMMSYSxx B-1
 masking syntax 3-5
 selection 3-2
 specifying with a view 1-20
 positional 1-20
 values, definition of 3-2
parmlib
 member descriptions 2-3
 SVOS commands 4-9
PASSWORD B-5, B-35
PERFRM_PRC B-5
PGM 10-3
POOL B-5, B-35
pool
 defining parameters with SMPPOOL B-69
POOLNAME 7-3, B-69, B-70, B-73
 description B-69
pools
 examples 7-4
pop-up menus 1-18
primary action commands
 HELP 1-11
PRO\$D\$M\$
 activates WTO message trace 10-9
PROCOLD B-5, B-36
PRODUMP
 produces DUMP 10-9
PROGRAM B-81
PROIGN
 deactivates tracing 10-9
PROMTRCE
 activates entry/exit tracing 10-9
PROTRACE
 activates tracing 10-9

R

RACF 2-8
REFRESH command 4-19, 4-20
REJECT B-5, B-36
related publications xvi
release notes xviii
remote data 1-21
REPLACE 2-8
REQTYPE B-5, B-36
RLST 6-3, B-77, B-78
rule list parameters 11-6
 examples 11-8
rule lists
 defined 2-8

S

SCAT B-5, B-37
secondary reduction
 limitations B-29
selection criteria ordering 3-9
selection parameters 2-6
 comparison operators 3-4
 defined 2-5, 3-2
 syntax 3-10
SET command 4-19, 4-21
 description B-1
SET statement filter list parameters 11-2
SET statements
 defined 3-1
setting the calendar parameter
 SMCAL S B-74
SG_INITPOOL B-5, B-37
SG_INITVOL B-5, B-37
SG_IXFPNTVL B-38
SG_MAXACCT B-5, B-38
SG_MAXPOOL B-5, B-38
SG_MAXSSDSZ B-6, B-39
SG_READNTVL B-6, B-39
SG_RETRYLIM B-6, B-39
SG_SIBSTK B-6
SG_SPACHLDR B-6, B-40
SG_SUBTASKS B-6, B-40
SG_WRITNTVL B-6, B-40
SGA_ENQSCOP B-6, B-41
SGACMD B-6, B-41

SGASCAN B-6, B-41
 SGASIM B-6, B-41
 SG-Auto 4-4
 SGC_ADDEXIT B-6
 SGC_CHKEXIT B-6, B-42
 SGC_DEFEXIT B-6
 SGC_KEYEXIT B-6, B-42
 SGC_SECEXIT B-7, B-43
 SGC_SELEXIT B-7, B-43
 SGC_STOGRP B-7, B-43
 SGC_STORCLS B-7, B-43
 SGCDN B-7, B-44
 SGD_PROCNM B-7, B-44
 SGD_SMFID B-7, B-44
 SGDCOLLECT 7-3, B-7, B-44, B-71
 SGDCOLLECTn B-45, B-69
 SGDPROCCNMn B-45
 SGDPROCNMn B-7
 SGDSMFIDn B-7
 SGEXITACCTn B-7, B-46
 SGEXITPOOLn B-8, B-46
 SGEXITVOLn B-8
 SGINITPOOLn B-8
 SGINITVOLn B-8
 SGMAXACCTn B-8
 SGMAXPOOLn B-8
 SGMAXSSDSZn B-8, B-48
 SGP_EXITBBS B-8, B-49
 SGP_EXITLIB B-8, B-49, B-53
 SGP_MAXCCUS B-8, B-49
 SGP_MAXDIRS B-8, B-50
 SGP_MAXDNS B-8, B-50
 SGP_MAXJOBS B-9, B-50
 SGP_MAXLCUS B-9, B-50
 SGP_MAXPOLs B-9, B-51
 SGP_MAXPTHS B-9, B-51
 SGP_MAXPVLs B-9, B-51
 SGP_MAXRRKS B-9, B-51
 SGP_MAXRSFS B-9, B-52
 SGP_MAXSCLS B-9, B-52
 SGP_MAXVOLS B-9, B-52
 SGP_RDFCOMP B-9, B-52
 SGP_SIBSTK B-9, B-53
 SGP_SMF42 B-9
 SGP_TRACE B-9, B-53
 SGPROCACCTn B-10, B-53
 SGPROCPOOLn B-10, B-54
 SGPROCVLERn B-10, B-54
 SGPROCVOLn B-10, B-54
 SGREADNTVLn B-10, B-55
 SGRETRYLIMn B-10, B-55
 SGSPACHLDRn B-10, B-55
 SGSUBTASKSn B-10, B-56
 SGWRITNTVLn B-10, B-56
 shared parmlib environment 2-10
 SHUTDOWN command 4-12
 single system image (SSI) 1-21
 SIZEISPRIM B-11, B-28, B-56
 SKIP B-11, B-57
 SMACMDxx 2-3
 SMCALSxx 2-2, 2-3, 5-19, 9-2
 parameter for calendar settings B-74
 SMCRTxx 5-20
 SMDIAGxx 2-2, 2-3, 5-19, 10-2
 defining diagnostic parameters B-81
 SMDIAGxx subparameters
 ABEND B-81
 DEBUG B-81
 DUMP B-81
 IGNORE B-81
 MODTRC B-81
 TRACE B-81
 SMEVNTxx 2-3
 SMF 2-3, 5-19, 6-3, 11-2, 11-4, B-77, B-79
 SMF message library 13-2
 SMF record format 13-2
 SMFID B-11, B-58
 SMFLSTxx 2-4, 5-19, 11-2
 SMFUNCxx 2-2, 2-3, 5-19, 6-2, B-77
 SMMSYSxx 2-2, 2-3, 5-19, 6-2, B-1
 global parameters B-1
 SET keyword B-1
 subparameters
 CAL B-1
 FUNC B-1
 MSGID B-1
 MSGPREF B-1
 OPMHLQ B-1
 PASSWORD B-1
 POOL B-1
 SMFID B-1
 usage notes B-14
 SMMSYSxx subparameters B-1
 SMPPOOL
 defining pool parameters B-69, B-73
 SMPPOOLxx 2-2, 2-3, 5-19, 7-2, B-69, B-73

- parameters 7-2
- SMPOOL_{xx} subparameters
 - POOLNAME B-69, B-73
 - TYPE B-69
 - USELIMIT B-69
- SMRLST_{xx} 2-4, 5-19, 5-20, 11-6
- SMRORG_{xx} 5-20
- SMS subpools 7-2
- SMS_ALLOC B-11, B-58
- SMS_EXTEND B-11, B-59
- SMSMANAGED 2-7
- SMSMCREN 11-7
- SMSMF000 13-2
- SMSMFCNV 13-2
- SMSPOL B-73
- SMSPOL_{xx} 2-2, 2-3, 5-19
- SMSPPOOL B-11
- SMVARS_{xx} 2-2, 2-3, 5-19, 8-1, B-76, B-82, B-86
- SMVSCF 5-19
- SMVSCF_{xx} 2-4
- SMVSCF_{xx}S 2-3
- Softworks 4-8
- space reduction
 - limitations B-29
- SPACVOLA B-63, B-72
- START command 4-13
- START/STOP
 - Commands 4-19
- START/STOP commands 4-19
- starting and stopping MAINVIEW SRM 4-4
- starting and stopping MAINVIEW SRM with other products 4-8
- starting MAINVIEW SRM components 4-5
 - EasyHSM 4-5
 - EasyPOOL 4-5
 - EasySMS 4-5
 - Enterprise Storage Automation 4-5
 - historical performance 4-5
 - messages 4-6
 - S SVALLOC 4-5
 - S SVESA 4-5
 - S SVHSM 4-5
 - S SVOS,SUF=00 4-5
 - S SVSGA 4-5
 - S SVSGD 4-5
 - S SVSGP 4-5
 - SG-Auto 4-5
 - StopX37/II 4-5
 - StorageGUARD 4-5
 - SVOS 4-5
- statements and parameters 3-1
- static activate
 - activate
 - static 4-2
- STATUS command 4-14
- STEP 10-3, B-81
- STKSCR B-11, B-59
- STOP command 4-15
- stopping MAINVIEW SRM 4-7
 - with other products 4-8
- stopping MAINVIEW SRM components 4-7
 - /P SVALLOC 4-7
 - /P SVESA 4-7
 - /P SVHSM 4-7
 - /P SVOS 4-7
 - /P SVSGA 4-7
 - /P SVSGD 4-7
 - /P SVSGP 4-7
 - messages 4-7
- stopping SVOS 4-8
- StopX37/II 4-4
- StorageGUARD
 - historical performance 4-4
 - historical space 4-4
- SUBCHAR
 - parm on SVOS 4-17
- SUBSYS
 - parm on SVOS 4-18
- subsystem character
 - modification 4-17
 - using 4-10
- subsystem command character
 - use of 4-9
- SUPJSCAT 11-6
- SUPVOLRF 11-6
- SVOS 4-4
 - console interface 4-18
 - entering commands 4-9
 - execution JCL 4-16
 - multiple subsystem support 4-3
 - operator commands 4-18
 - parameters 4-16
- SVOS command definitions 4-11
- SVOS command syntax 4-9
- SVOS commands

- DUMP 4-11
- FORCE 4-11
- SHUTDOWN 4-11
- START 4-11
- STAT 4-11
- STATUS 4-11
- STOP 4-11
- SVOS Console Interface 4-18
- switching to a cached device B-57
- syntax
 - masking 3-5
 - shared parmlib environment 2-11
- syntax rules 3-1
- syntax statement conventions xix
- SYSLIB B-11, B-60
- SYSLIB2 B-11
- SYSLIB3 B-11
- sysplex
 - defining environment 2-10
 - parameters 2-11
- system
 - monitoring activity 5-18
 - multiple 1-21
- system configuration members 2-2
- system utilities
 - restrictions 1-26

T

- TAPEDEFR 11-7
- TAPEGENR B-11, B-60
- target
 - accessing 1-26
- TIME command
 - examples 1-29
- TRACE 6-3, 10-2, 10-4, B-61, B-77, B-80
 - options 10-9
- TRACEDD 10-9, B-11, B-61
- TRKCYL B-11, B-61
- TRKLEN B-11, B-62
- TSO B-78, B-80
- TYPE 7-3, B-69, B-71, B-81
- typographical conventions xviii

U

- USECAT B-11, B-62
- USELIMIT 7-3, B-69, B-72, B-81
- USERVARS 6-2
 - example 6-7
 - function 6-6
 - parameters 6-6
- using SSI
 - to check devices 1-24
- USRCn 6-6
- USRNy 6-6

V

- VALUE 8-2, B-76
- VAR B-11, B-62
- VARIABLE 8-2, B-76
- variables 2-9, 8-1
 - examples 8-3
 - name masks 3-6
 - parameters B-76
 - usage 8-3
- VASMLIMWARN B-63
- VASMPRIM B-63
- version number
 - SVOS 4-15
- view
 - displaying 1-13
 - from menus 1-14
 - using commands 1-13, 1-20
 - using hyperlinks 1-13
 - with positional parameters 1-20
 - HELP 1-11
 - help 1-12
 - menu-driven 1-14
 - naming conventions 1-12
- view customization 1-10
- views
 - MAINVIEW SRM Performance Monitor
 - options 12-1
- VOL 2-6, 2-7, 7-3, B-69, B-72, B-73, B-74
- VOLSEL B-72
- volume selection B-61, B-62
- volume switch
 - cached devices B-19
 - device characteristics B-19

dualcopy B-19
fastwrite B-19
same device characteristics B-57
shared devices B-19
VSAM B-63
VSAMJCL B-11, B-63
VSAMLIMWARN B-12
VSAMPRIM B-12
VSAMZSEC B-12, B-64
VSCAN_MNTSK B-12, B-64
VSCAN_MXTSK B-12, B-64
VSCAN_OINDX B-12, B-65
VSCAN_OPRI B-12, B-65
VSCAN_OSEC B-12, B-65
VSCAN_OUNIT B-12, B-66
VSCAN_OVOL B-12, B-66
VSCAN_TPRI B-12, B-66
VSCAN_TSEC B-12, B-66
VSCAN_TUNIT B-12, B-67
VSCAN_TVOL B-12, B-67

W

window
 simultaneously displayed 1-21
window interface
 using multiple windows 1-21
WTODC B-12, B-67
WTORC B-13, B-68

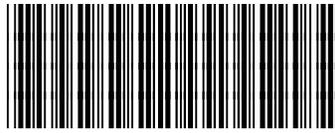
X

X37POOL B-13, B-68

Y

YEAR 9-2, B-74, B-75

Notes



100038105