



SGI® InfiniteStorage System Manager (ISSM)
Enterprise Edition
CLI and Script Commands
Programmer Reference Manual

007-5487-001

COPYRIGHT

© 2008 SGI. All rights reserved; provided portions may be copyright in third parties, as indicated elsewhere herein. No permission is granted to copy, distribute, or create derivative works from the contents of this electronic documentation in any manner, in whole or in part, without the prior written permission of SGI.

LIMITED RIGHTS LEGEND

The software described in this document is “commercial computer software” provided with restricted rights (except as to included open/free source) as specified in the FAR 52.227-19 and/or the DFAR 227.7202, or successive sections. Use beyond license provisions is a violation of worldwide intellectual property laws, treaties and conventions. This document is provided with limited rights as defined in 52.227-14.

The electronic (software) version of this document was developed at private expense; if acquired under an agreement with the USA government or any contractor thereto, it is acquired as “commercial computer software” subject to the provisions of its applicable license agreement, as specified in (a) 48 CFR 12.212 of the FAR; or, if acquired for Department of Defense units, (b) 48 CFR 227-7202 of the DoD FAR Supplement; or sections succeeding thereto. Contractor/manufacturer is SGI, 1140 E. Arques Avenue, Sunnyvale, CA 94085.

TRADEMARKS AND ATTRIBUTIONS

Altix, SGI, and the SGI logo are registered trademarks and SGI ProPack is a trademark of SGI in the United States and/or other countries worldwide.

Java is a trademark or registered trademark of Sun Microsystems, Inc., in the U.S. and other countries. Linux is a registered trademark of Linus Torvalds. Red Hat is a registered trademark of Red Hat, Inc. SUSE LINUX and the SUSE logo are registered trademarks of Novell, Inc. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. Windows and Windows NT are registered trademarks of Microsoft Corporation. The program product(s) described in this book is a licensed product of LSI Logic Corporation. LSI Logic and SYM are registered trademarks and RAID Manager and SYMlicity are trademarks of LSI Logic Corporation. All other trademarks mentioned herein are the property of their respective owners.

Record of Revision

Version	Description
001	June 2008 Original printing.

Contents

	About This Guide.	xxi
1.	About the Command Line Interface	1
	How to Use the Command Line Interface	2
	Usage Notes	3
	CLI Commands	4
	Command Line Terminals	6
	Formatting Considerations	11
	Detailed Error Reporting	12
	Exit Status	13
	Usage Examples	14
2.	About Script Commands.	17
	Script Command Structure	19
	Script Command Synopsis	21
	Recurring Syntax Elements	23
	Usage Guidelines	28
	Adding Comments to a Script File	29
3.	Configuring a Storage Array	31
	Configuration Concepts	32
	Controllers.	33
	Drives	35
	Hot Spares	36
	Volume Groups	37
	Volumes	38
	RAID Levels	40
	Hosts	43
	Host Bus Adapter Host Ports	43

Host Groups 44
Logical Unit Numbers. 44
Configuring a Storage Array 45
Determining What Is on Your Storage Array 45
Clearing the Configuration 49
Using the Auto Configure Command 50
Using the Create Volume Command 53
Modifying Your Configuration 58
Setting the Controller Clocks. 58
Setting the Storage Array Password. 59
Setting the Storage Array Host Type 60
Setting the Storage Array Cache 61
Setting the Modification Priority. 66
Assigning Global Hot Spares. 67
Saving a Configuration to a File 67
4. Using the Snapshot Premium Feature 69
How Snapshot Works 70
Creating a Snapshot Volume 71
Creating a Snapshot Volume with User-Assigned Drives 72
Creating a Snapshot Volume with Software-Assigned Drives 73
Creating a Snapshot Volume by Specifying a Number of Drives 74
User-Defined Parameters. 74
Snapshot Volume Names and Snapshot Repository Volume Names 77
Changing Snapshot Volume Settings 78
Stopping, Restarting, and Deleting a Snapshot Volume 79
5. Using the Remote Volume Mirroring Premium Feature 81
How Remote Volume Mirroring Works 82
Mirror Repository Volumes 83
Mirror Relationships 83
Data Replication 84
Link Interruptions or Secondary Volume Errors 85
Resynchronization. 86

Creating a Remote-Mirror Pair	87
Performance Considerations	87
Enabling the Remote Volume Mirroring Feature	88
Activating the Remote Volume Mirroring Feature	88
Determining Candidates for a Remote-Mirror Pair	92
Creating a Remote-Mirror Pair	92
Changing Remote Volume Mirroring Settings	94
Suspending and Resuming a Mirror Relationship	95
Removing a Mirror Relationship	96
Deleting a Primary Volume or a Secondary Volume	97
Disabling the Remote Volume Mirroring Feature	97
Deactivating the Remote Volume Mirroring Feature	97
Interaction with Other Features	98
SANshare Storage Partitioning	98
Snapshot Volumes	99
Volume Copy	99
Dynamic Volume Expansion	100
6. Using the Volume Copy Premium Feature	101
How Volume Copy Works	101
Source Volume	102
Target Volume	103
Volume Copy and Persistent Reservations	104
Storage Array Performance	104
Restrictions	105
Volume Copy Commands	105
Creating a Volume Copy	106
Enabling the Volume Copy Feature	107
Determining Volume Copy Candidates	107
Creating a Volume Copy	107
Viewing Volume Copy Properties	109
Changing Volume Copy Settings	110
Recopying a Volume	111
Stopping a Volume Copy	112

Removing Copy Pairs	113
Interaction with Other Features	113
SANshare Storage Partitioning	114
Snapshot Volumes.	114
Remote Volume Mirroring	115
7. Maintaining a Storage System	117
Routine Maintenance	117
Running a Media Scan	117
Running a Redundancy Check	119
Resetting a Controller	119
Enabling a Controller Data Transfer.	120
Resetting the Battery Age.	120
Removing Persistent Reservations	120
Synchronizing the Controller Clocks	121
Locating Drives	121
Relocating a Volume Group	121
Performance Tuning	123
Monitoring the Performance	123
Changing the RAID Levels	125
Changing the Segment Size	125
Changing the Cache Parameters.	126
Defragmenting a Volume Group	127
Troubleshooting and Diagnostics.	128
Collecting All Support Data	128
Collecting Drive Data	130
Diagnosing a Controller	130
Running Read Link Status Diagnostics	131
Collecting Switch-on-a-Chip Error Statistics	135
Recovery Operations	137
Setting the Controller Operational Mode	137
Changing the Controller Ownership	138
Initializing a Drive.	139
Reconstructing a Drive	139

Initializing a Volume140
Redistributing Volumes140
Replacing Canisters141
8. Script Commands145
Naming Conventions146
Entering Numerical Names147
Firmware Compatibility Levels148
Examples of Firmware Compatibility Levels148
Formatting Rules for Script Commands149
Commands Listed by Function151
Controller Commands151
Drive Commands152
Host Topology Commands153
Remote Volume Mirroring Commands153
Session Command154
Snapshot Commands154
Storage Array Commands154
Tray Commands156
Uncategorized Commands156
Volume Commands156
Volume Copy Commands157
Volume Group Commands158
Commands Listed Alphabetically159
Activate Remote Volume Mirroring Feature159
Activate Storage Array Firmware162
Autoconfigure Storage Array162
Autoconfigure Storage Array Hot Spares166
Check Remote Mirror Status166
Check Volume Parity167
Clear Drive Channel Statistics169
Clear Storage Array Configuration169
Clear Storage Array Event Log171
Clear Storage Array Firmware Pending Area171

Clear Volume Reservations	172
Clear Volume Unreadable Sectors	173
Create Host.	174
Create Host Group	175
Create Host Port	176
Create RAID Volume (Automatic Drive Select)	177
Create RAID Volume (Free Extent Based Select)	180
Create RAID Volume (Manual Drive Select)	184
Create Remote Mirror	188
Create Snapshot Volume	192
Create Volume Copy	197
Create Volume Group.	198
Deactivate Remote Mirror	201
Delete Host.	202
Delete Host Group.	203
Delete Host Port	204
Delete Volume	205
Delete Volume Group.	206
Diagnose Controller	206
Diagnose Remote Mirror	208
Disable Storage Array Feature	209
Download Drive Firmware	209
Download Environmental Card Firmware	211
Download Storage Array Drive Firmware	212
Download Storage Array Firmware/NVSRAM	213
Download Storage Array NVSRAM.	214
Enable Controller	215
Enable Storage Array Feature Key	216
Recopy Volume Copy	217
Recover RAID Volume	218
Re-create Remote Volume Mirroring Repository Volume	221
Re-create Snapshot	224
Re-create Snapshot Collection	225

Remove Remote Mirror226
Remove Volume Copy227
Remove Volume LUN Mapping228
Repair Volume Parity230
Replace Drive231
Reset Controller232
Reset Storage Array Battery Install Date233
Reset Storage Array Diagnostic Data234
Reset Storage Array RLS Baseline235
Reset Storage Array SOC Baseline235
Reset Storage Array Volume Distribution236
Resume Remote Mirror237
Revive Drive238
Revive Volume Group239
Save Controller NVSRAM240
Save Drive Channel Fault Isolation Diagnostic Status241
Save Drive Log242
Save Storage Array Configuration243
Save Storage Array Diagnostic Data245
Save Storage Array Events246
Save Storage Array Performance Statistics.247
Save Storage Array RLS Counts.248
Save Storage Array SOC Counts249
Save Storage Array State Capture250
Save Storage Array Support Data250
Set Controller251
Set Controller Service Action Allowed Indicator254
Set Drive Channel Status.255
Set Drive Hot Spare256
Set Drive Service Action Allowed Indicator257
Set Drive State.258
Set Foreign Drive to Native258
Set Host259

Set Host Channel	263
Set Host Group.	264
Set Host Port	265
Set Remote Mirror	266
Set Session	268
Set Snapshot Volume	271
Set Storage Array	272
Set Storage Array Redundancy Mode	276
Set Storage Array Time	276
Set Storage Array Tray Positions.	277
Set Tray Alarm.	278
Set Tray Identification.	279
Set Tray Service Action Allowed Indicator	280
Set Volume	282
Set Volume Copy	289
Set Volume Group.	290
Set Volume Group Forced State	291
Show Controller	292
Show Controller NVSRAM	296
Show Drive Channel Stats	297
Show Drive Download Progress.	298
Show Host Ports	299
Show Remote Volume Mirroring Volume Candidates	300
Show Remote Volume Mirroring Volume Synchronization Progress	301
Show Storage Array	302
Show Storage Array Auto Configure	306
Show Storage Array Host Topology.	309
Show Storage Array LUN Mappings	311
Show Storage Array Unreadable Sectors	312
Show String	313
Show Volume Action Progress	314
Show Volume Copy	315
Show Volume Copy Source Candidates	316

Show Volume Copy Target Candidates317
Show Volume Group317
Show Volume Group Export Dependencies319
Show Volume Group Import Dependencies320
Show Volume Performance Statistics321
Show Volume Reservations322
Start Drive Channel Locate322
Start Drive Initialize323
Start Drive Locate324
Start Drive Reconstruction324
Start Remote Volume Mirroring Synchronization325
Start Storage Array Locate325
Start Tray Locate326
Start Volume Group Defragment326
Start Volume Group Export327
Start Volume Group Import328
Start Volume Group Locate329
Start Volume Initialization329
Stop Drive Channel Locate330
Stop Drive Locate330
Stop Snapshot331
Stop Storage Array Drive Firmware Download331
Stop Storage Array Locate332
Stop Tray Locate332
Stop Volume Copy332
Stop Volume Group Locate333
Suspend Remote Mirror333
Show Storage Array Command335
Show Controller NVSRAM351
Show Drive354
Show Drive Channel Status360
Show Volumes362
Configuration Script Example 1368

Configuration Script Example 2	370
Deprecated Commands	373
Deprecated Parameters	378
Description of the Asynchronous Remote Volume Mirroring Utility	381
Operation of the Asynchronous Remote Volume Mirroring Utility	382
Running the Asynchronous Remote Volume Mirroring Utility	383
Configuration Utility	384

Figures

Figure 1	Host and Storage Relationship	33
-----------------	---	----

Tables

Table 1	Command Name Syntax Conventions	4
Table 2	Command Line Terminals.	6
Table 3	Exit Status	13
Table 4	Configuration and Management Operations	17
Table 5	Object Types and Identifiers	20
Table 6	General Form of the Script Commands	21
Table 7	Recurring Syntax Elements	24
Table 8	Host Ports and the Type of Host Interfaces on Controller-Drive Trays.	34
Table 9	Maximum Number of Drives	36
Table 10	Maximum Number of Drives in a Volume Group Based on Capacity	37
Table 11	Volume Specifications by Supported Controller Model	40
Table 12	RAID Level Configurations	41
Table 13	Maximum Number of HBA Host Ports per Controller.	44
Table 14	Default Values for Segment Size and Cache Read Prefetch	52
Table 15	Values for the cacheFlushModifier Parameter	63
Table 16	Components of a Snapshot Volume	70
Table 17	Snapshot Volume Commands	71
Table 18	Snapshot Volume Parameters.	75
Table 19	Maximum Number of Defined Mirrors	82
Table 20	Volume Copy Commands.	105
Table 21	Information About Storage Array Performance	123
Table 22	Support Data for the Storage Array	128
Table 23	RLS Baseline Data for the Storage Array.	132
Table 24	Service Action Not Allowed	142
Table 25	Valid Characters for a CHAP Secret	261
Table 26	Deprecated Commands	373
Table 27	Deprecated Parameters	378

What's New in This Guide

This document includes instructions for installing and running the SGI InfiniteStorage System Manager Enterprise Edition (ISSM-EE) software. This update contains the following new features and software enhancements:

- Supported on SGI InfiniteStorage (IS) 4600, 4500, 4000, and 350
- RAID 6 on IS350 and IS4000
- Full-config RAID 10 and RAID 0 volume groups:
 - IS4600– up to 112+112 RAID 10 or 224-drive RAID 0
 - IS4500– up to 112+112 RAID 10 or 224-drive RAID 0
 - IS4000 – up to 56+56 RAID 10 or 112-drive RAID 0
- Greater than 2 TB volumes (LUNs)
- Full-config RAID 10 and RAID 0 volume groups
- More partitions and logins:
 - IS4500 increased to 512 partitions and 2,048 logins (host ports). Up from 64 and 512.
 - IS350 and IS4000 increased to 128 partitions and 512 logins (host ports). Up from 64 and 256.
- More mirrors per system:
 - IS4500 increased to 128. Up from 64.
 - IS350 and IS4000 increased to 64. Up from 32.
- More snapshots per volume:
 - IS4500 increased to 16. Up from 4.
 - IS350 and IS4000 increased to 8. Up from 4.
- Global Hot Spare usage improvements. Global hot spare can become a permanent member of a volume group or invoke the copy back command once the failed drive has been replaced. Unlimited number (up from 15).

About This Guide

This guide provides information on preparing, installing, configuring, and starting the SGI InfiniteStorage System Manager Enterprise Edition software (ISSM-EE) in a Linux® or Windows® environment.

This is part of a document set that supports ISSM-EE. To get the most out of this book, consult the following documentation first:

- `README` file — Read this text file first. It contains late-breaking installation and user information about the storage management software.

When you have finished the entire installation process, see the following online Help systems, which contain information common to all operating environments.

- Enterprise Management window Help — Use this online Help system to learn more about working with the management domain.
- Array Management window Help — Use this online Help system to learn more about managing storage arrays.

Audience

This guide is intended for system administrators. Use this guide to:

- Gain a basic understanding of the ISSM-EE software.
- Understand the two different storage array management methods.
- Learn about hardware and software requirements.
- Follow procedures to install the ISSM-EE software.

Related Publications

This Administration Guide is part of a product book set. To get the most out of this guide, consult the following documentation before using this guide.

- *SGI InfiniteStorage* hardware storage array documentation for:
 - SGI Total Performance TP9300, TP9400, TP9500, and TP9700
 - InfiniteStorage 350
 - InfiniteStorage 4000
 - InfiniteStorage 4500
 - InfiniteStorage 4600
- *SGI InfiniteStorage System Manager Enterprise Edition (ISSM-EE) Administration Guide*, publication number 007-5478-00x
This guide provides procedures for installing, configuring, and starting the ISSM-EE software in a Linux or Windows environment.
- *SGI InfiniteStorage System Manager Enterprise Edition (ISSM-EE) Software Concepts Guide*, publication number 007-5479-00x
This guide explains the terminology and features of the ISSM-EE storage management software.

Product Support

SGI provides a comprehensive product support and maintenance program for its products. SGI also offers services to implement and integrate Linux applications in your environment.

- Refer to <http://www.sgi.com/support/>
- If you are in North America, contact the Technical Assistance Center at +1 800 800 4SGI or contact your authorized service provider.
- If you are outside North America, contact the SGI subsidiary or authorized distributor in your country.

Reader Comments

If you have comments about the technical accuracy, content, or organization of this document, contact SGI. Be sure to include the title and document number of the manual with your comments. (Online, the document number is located in the front matter of the manual. In printed manuals, the document number is located at the bottom of each page.)

You can contact SGI in any of the following ways:

- Send e-mail to the following address: techpubs@sgi.com
- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.
- Send mail to the following address:

SGI
Technical Publications
1140 East Arques Avenue
Sunnyvale, CA 94085-4602

SGI values your comments and will respond to them promptly.

About the Command Line Interface

The command line interface (CLI) is a software tool that lets storage array installers, developers, and engineers configure and monitor storage arrays. Using the CLI, you can run commands from an operating system prompt, such as the DOS `C:` prompt, or a Linux operating system path.

Each command performs a specific action for managing a storage array or returning information about the status of a storage array. You can enter individual commands, or you can run script files when you need to perform operations more than once. For example, you can run script files when you want to install the same configuration on several storage arrays. The CLI lets you load a script file from a disk and run the script file. The CLI provides a way to run storage management commands on more than one network storage array. You can use the CLI both in installation sites and in development environments.

The CLI gives you direct access to a script engine that is a utility in the SGI InfiniteStorage System Manager (ISSM) Enterprise Edition software (also referred to as the storage management software). The script engine runs commands that let you configure and manage storage arrays. The script engine reads the commands, or runs a script file, from the command line and performs the operations instructed by the commands.

NOTE You can also access the script engine by using the Enterprise Management Window in the storage management software. If you access the script engine by using the Enterprise Management Window, you can edit or run script commands on only one storage array in the script window. You can open a script window for each storage array in your configuration and run commands in each window. By using the CLI, you can run commands on more than one storage array from a single command line.

You can use the command line interface to perform these actions:

- Directly access the script engine and run script commands
- Create script command batch files to be run on multiple storage arrays when you need to install the same configuration on different storage arrays

- Run script commands on an in-band managed storage array, an out-of-band managed storage array, or a combination of both
- Show configuration information about the network storage arrays
- Add storage arrays to and remove storage arrays from the management domain
- Perform automatic discovery of all of the storage arrays that are attached to the local subnet
- Add or delete Simple Network Management Protocol (SNMP) trap destinations and email alert notifications
- Specify the mail server and sender email address or SNMP server for alert notifications
- Show the alert notification settings for storage arrays that are currently configured in the Enterprise Management Window
- Direct the output to a standard command line display or to a named file

How to Use the Command Line Interface

The commands that you run on the CLI provide access to the script engine, specify the storage array to receive the script commands, and set operation environment parameters.

A CLI command consists of these elements:

- The term `smeecli`
- The storage array identifier
- Parameters
- Script commands

A CLI command takes this form:

```
smeecli storageArray parameters script-commands;
```

- `smeecli` invokes the command line interface.
- *storageArray* is the name or the IP address of the storage array.
- *parameters* are CLI parameters that define the environment and the purpose for the command.

- *script-commands* are one or more script commands or the name of a script file that contains script commands. (The script commands are the storage array configuration commands.)

“[Chapter 2, About Script Commands](#),” presents an overview of the script commands.

“[Chapter 8, Script Commands](#),” provides definitions, syntax, and parameters for the script commands.)

Usage Notes

If you enter `smcli` or `smecli` and a storage array name but do not specify CLI parameters, script commands, or a script file, the command line interface runs in interactive mode. Interactive mode lets you run individual commands without prefixing the commands with `smcli` or `smecli`. In interactive mode, you can enter a single command, view the results, and enter the next command without typing the complete `smcli` string. Interactive mode is useful for determining configuration errors and quickly testing configuration changes.

To end an interactive mode session, type the operating system-specific command for terminating a program (such as **Control-C** on the UNIX operating system or the Windows operating system). Typing the termination command (**Control-C**) while in interactive mode turns off interactive mode and returns operation of the command prompt to an input mode that requires you to type the complete `smcli` string.

If you enter an incomplete or inaccurate `smecli` string that does not have the correct syntax, parameter names, options, or terminals, the script engine returns usage information.

CLI Commands

[Table 1](#) lists the conventions used in the command statement syntax to manage the storage array.

Table 1 Command Name Syntax Conventions

Convention	Definition
a b	Alternative (“a” or “b”)
<i>italicized-words</i>	A terminal that needs user input to fulfill a parameter (a response to a variable)
[...] (square brackets)	Zero or one occurrence (square brackets are also used as a delimiter for some command parameters)
{ ... } (curly braces)	Zero or more occurrences
(a b c)	Choose only one of the alternatives
bold	A terminal that needs a command parameter entered to start an action

General forms of the CLI commands follow, which show the terminals and the parameters that are used in each command. [Table 2](#) on page 6 lists definitions for the terminals shown in the general form.

```
smeecli host-name-or-IP-address [host-name-or-IP-address]
[-c "command; {command2};"]
[-n storage-system-name | -w wwID]
[-o outputfile] [-p password] [-e] [-S] [-quick]
```

```
smeecli host-name-or-IP-address [hostname-or-IP-address]
[-f scriptfile]
[-n storage-system-name | -w wwID]
[-o outputfile] [-p password] [-e] [-S] [-quick]
```

```
smeecli (-n storage-system-name | -w wwID)
[-c "command; {command2};"]
[-o outputfile] [-p password] [-e] [-S] [-quick]
```

```
smeecli (-n storage-system-name -w wwID)
[-f scriptfile]
[-o outputfile] [-p password] [-e] [-S] [-quick]
```

```
smeecli -a email: email-address [host-name-or-IP-address1
[host-name-or-IP-address2]]
[-n storage-system-name | -w wwID | -h host-name | -r (host_sa |
direct_sa)]
[-I information-to-include] [-q frequency] [-S]

smeecli -x email: email-address [host-name-or-IP-address1
[host-name-or-IP-address2]]
[-n storage-system-name | -w wwID | -h host-name | -r (host_sa |
direct_sa)] [-S]

smeecli (-a | -x) trap: community, host-name-or-IP-address
[host-name-or-IP-address1 [host-name-or-IP-address2]]
[-n storage-system-name | -w wwID | -h host-name | -r (host_sa |
direct_sa)] [-S]

smeecli -d [-w] [-i] [-s] [-v] [-S]

smeecli -m host-name-or-IP-address -F email-address
[-g contactInfoFile] [-S]

smeecli -A [host-name-or-IP-address [host-name-or-IP-address]]
[-S]

smeecli -X (-n storage-system-name | -w wwID | -h host-name)

smeecli -?
```

Command Line Terminals

Table 2 Command Line Terminals

Terminal	Definition
host-name-or-IP-address	<p>Specifies either the host name or the Internet Protocol (IP) address (<i>xxx . xxx . xxx . xxx</i>) of an in-band managed storage array or an out-of-band managed storage array.</p> <ul style="list-style-type: none"> • If you are managing a storage array by using a host through in-band storage management, you must use the <i>-n</i> terminal or the <i>-w</i> terminal if more than one storage array is connected to the host. • If you are managing a storage array by using out-of-band storage management through the Ethernet connection on each controller, you must specify the <i>host-name-or-IP-address</i> of the controllers. • If you have previously configured a storage array in the Enterprise Management Window, you can specify the storage array by its user-supplied name by using the <i>-n</i> terminal. • If you have previously configured a storage array in the Enterprise Management Window, you can specify the storage array by its World Wide Identifier (WWID) by using the <i>-w</i> terminal.
-A	<p>Adds a storage array to the configuration file. If you do not follow the <i>-A</i> terminal with a <i>host-name-or-IP-address</i>, auto-discovery scans the local subnet for storage arrays.</p>
-a	<p>Adds a Simple Network Management Protocol (SNMP) trap destination or an email address alert destination.</p> <ul style="list-style-type: none"> • When you add an SNMP trap destination, the SNMP community is automatically defined as the community name for the trap, and the <i>host</i> is the IP address or Domain Name Server (DNS) host name of the system to which the trap should be sent. • When you add an email address for an alert destination, the <i>email-address</i> is the email address to which you want the alert message to be sent.

Terminal	Definition
-c	Indicates that you are entering one or more script commands to run on the specified storage array. End each command with a semicolon (;). You cannot place more than one -c terminal on the same command line. You can include more than one script command after the -c terminal.
-d	Shows the contents of the script configuration file. The file content takes this format: <i>storage-system-name host-name1 host-name2</i>
-e	Runs the commands without performing a syntax check first.
-F (uppercase)	Specifies the email address from which all alerts will be sent.
-f (lowercase)	Specifies a file name that contains script commands that you want to run on the specified storage array. (This terminal is similar to the -c terminal in that both terminals are intended for running script commands. The -c terminal runs individual script commands. The -f terminal runs a file of script commands.) By default, any errors that are encountered when running the script commands in a file are ignored, and the file continues to run. To override this behavior, use the <code>set session errorAction=stop</code> command in the script file.
-g	Specifies an ASCII file that contains email sender contact information that will be included in all email alert notifications. The CLI assumes that the ASCII file is text only, without delimiters or any expected format. Do not use this terminal if a <code>userdata.txt</code> file exists.
-h	Specifies the host name that is running the SNMP agent to which the storage array is connected. Use this terminal with the -a terminal and the -x terminal.
-I	Specifies the type of information to be included in the email alert notifications. You can select these values: <ul style="list-style-type: none"> • eventOnly • profile • supportBundle

Terminal	Definition
-i	Shows the IP address of the known storage arrays. Use this terminal with the -d terminal. The file contents takes this format: <i>storage-system-name IP-address1 IPaddress2</i>
-m	Specifies the host name or the IP address of the email server from which email alert notifications are sent.
-n	Specifies the name of the storage array on which you want to run the script commands. This name is optional when you use a <i>host-name-or-IP-address</i> . If you are using the in-band method for managing the storage array, you must use the -n terminal if more than one storage array is connected to the host at the specified address. The storage array name is required when the <i>host-name-or-IP-address</i> is not used. The name of the storage array that is configured for use in the Enterprise Management Window (that is, the name is listed in the configuration file) must not be a duplicate name of any other configured storage array.
-o	Specifies a file name for all output text that is a result of running the script commands when you use this terminal with the -c terminal or the -f terminal. If you do not specify an output file, the output text goes to standard output (stdout). All output from commands that are not script commands is sent to stdout, regardless of whether this terminal is set.
-p	Specifies the password for the storage array on which you want to run commands. A password is not necessary under these conditions: <ul style="list-style-type: none">• A password has not been set on the storage array.• The password is specified in a script file that you are running.• You specify the password by using the -c terminal and the <code>set session password=password</code> command.

Terminal	Definition
-q	<p>Specifies the frequency that you want to include additional profile or support bundle information in the email alert notifications. An email alert notification containing at least the basic event information is always generated for every critical event. If you set the -I terminal to eventOnly, the only valid value for the -q terminal is everyEvent. If you set the -I terminal to either the profile value or the supportBundle value, this information is included with the emails with the frequency specified by the -q terminal. These values are valid frequency values:</p> <ul style="list-style-type: none"> • everyEvent – Information is returned with every email alert notification. • 2 – Information is returned no more than once every two hours. • 4 – Information is returned no more than once every four hours. • 8 – Information is returned no more than once every eight hours. • 12 – Information is returned no more than once every 12 hours. • 24 – Information is returned no more than once every 24 hours.
-quick	<p>Reduces the amount of time that is required to run a single-line operation. An example of a single-line operation is the <code>recreate snapshot volume</code> command. This terminal reduces time by not running background processes for the duration of the command.</p> <p>Do not use this terminal for operations that involve more than one single-line operation. Extensive use of this command can overrun the controller with more commands than the controller can process, which causes operational failure. Also, status updates and configuration updates that are collected usually from background processes will not be available to the CLI. This terminal causes operations that depend on background information to fail.</p>
-r	<p>Sets or changes the alert notifications for all of the storage arrays that are managed by a specific storage management station. Use the -r terminal with the -a terminal or the -x terminal. For storage arrays that are managed out-of-band, use the direct_sa value. For storage arrays that are managed in-band, use the host_sa value.</p>

Terminal	Definition
-S (uppercase)	<p>Suppresses informational messages describing the command progress that appear when you run script commands. (Suppressing informational messages is also called silent mode.) This terminal suppresses these messages:</p> <ul style="list-style-type: none"> • Performance syntax check • Syntax check complete • Executing script • Script execution complete • <code>smcli</code> or <code>smeecli</code> completed successfully
-s (lowercase)	Shows the alert settings in the configuration file when used with the <code>-d</code> terminal.
-v	Show the current global status of the known devices in a configuration file when used with the <code>-d</code> terminal.
-w	Specifies the WWID of the storage array. This terminal is an alternate to the <code>-n</code> terminal. Use the <code>-w</code> terminal with the <code>-d</code> terminal to show the WWIDs of the known storage arrays. The file content take this format: <code>storage-system-name world-wide-ID IPaddress1 IP-address2</code> .
-X (uppercase)	Deletes a storage array from a configuration.
-x (lowercase)	Removes an SNMP trap destination or an email address alert destination. The <code>community</code> is the SNMP community name for the trap, and the <code>host</code> is the IP address or DNS host name of the system to which you want the trap sent.
-?	Shows usage information about the CLI commands.

Formatting Considerations

Double quotation marks (“ ”) that are used as part of a name or label require special consideration when you run the CLI commands and the script commands on a Microsoft Windows operating system. This section describes how to use double quotation marks in names while running CLI commands and script commands on a Windows operating system.

When double quotation marks (“ ”) are part of a name or value, you must insert a backslash (\) before each double quotation mark character. For example:

```
-c "set storageArray userLabel=\"Engineering\";"
```

where “Engineering” is the storage array name. A second example is:

```
-n \"My_Array\"
```

where “My_Array” is the name of the storage array.

You cannot use double quotation marks (“ ”) as part of a character string (also called *string literal*) within a script command. For example, you cannot enter the following string to set the storage array name to “Finance” Array:

```
-c "set storageArray userLabel=\"\"Finance\"Array\";"
```

In the Linux operating system, the delimiters around names or labels are single quotation marks (‘ ’). The Linux versions of the previous examples are as follows:

```
-c `set storageArray userLabel="Engineering";`  
-n "My_Array"
```

In a Windows operating system, if you do not use double quotation marks (“ ”) around a name, you must insert a caret (^) before each special script character. Special characters are ^, |, <, and >.

Insert a caret before each special script character when used with the terminals -n, -o, -f, and -p. For example, to specify storage array CLI>CLIENT, enter this string:

```
-n CLI^>CLIENT
```

Insert one caret (^) before each special script character when used within a string literal in a script command. For example, to change the name of a storage array to FINANCE_ | _PAYROLL, enter the following string:

```
-c "set storageArray userLabel=\"FINANCE_^|_PAYROLL\";"
```

Detailed Error Reporting

Error data collected from an error encountered by the CLI is written to a file. Detailed error reporting under the CLI works as follows:

- If the CLI must abnormally end running CLI commands and script commands, error data is collected and saved before the CLI finishes.
- The CLI saves the error data by writing the data to a standard file name.
- The CLI automatically saves the data to a file. Special command line options are not required to save the error data.
- You are not required to perform any action to save the error data to a file.
- The CLI does not have any provision to avoid over-writing an existing version of the file that contains error data.

For error processing, errors appear as two types:

- Terminal errors or syntax errors that you might enter
- Exceptions that occur as a result of an operational error

When the CLI encounters either type of error, the CLI writes information that describes the error directly to the command line and sets a return code. Depending on the return code, the CLI also might write additional information about which terminal caused the error. The CLI also writes information about what it was expecting in the command syntax to help you identify any syntax errors that you might have entered.

When an exception occurs while a command is running, the CLI captures the error. At the end of processing the command (after the command processing information has been written to the command line), the CLI automatically saves the error information to a file.

The name of the file to which error information is saved is `excp rpt . txt`. The CLI tries to place the `excp rpt . txt` file in the directory that is specified by the system property `devmgr . datadir`. If for any reason the CLI cannot place the file in the directory specified by `devmgr . datadir`, the CLI saves the `excp rpt . txt` file in the same directory from which the CLI is running. You cannot change the file name or the location. The `excp rpt . txt` file is overwritten every time that an exception occurs. If you want to save the information in the `excp rpt . txt` file, you must copy the information to a new file or a new directory.

Exit Status

Table 3 lists the exit statuses that might be returned and the meaning of each status.

Table 3 Exit Status

Status Value	Meaning
0	The command terminated without an error.
1	The command terminated with an error. Information about the error also appears.
2	The script file does not exist.
3	An error occurred while opening an output file.
4	A storage array was not at the specified address.
5	Addresses specify different storage arrays.
6	A storage array name does not exist for the host agent that is connected.
7	The storage array name was not at the specified address.
8	The storage array name was not in the configuration file.
10	A management class does not exist for the storage array.
11	A storage array was not found in the configuration file.
12	An internal error occurred.
13	Invalid script syntax was found.
14	The controller was unable to communicate with the storage array.
15	A duplicate argument was entered.
16	An execution error occurred.
17	A host was not at the specified address.
18	The WWID was not in the configuration file.
19	The WWID was not at the address.
20	An unknown IP address was specified.
21	The Event Monitor configuration file was corrupted.

Status Value	Meaning
22	The storage array was unable to communicate with Event Monitor.
23	The controller was unable to write alert settings.
24	The wrong organizer node was specified.
25	The command was not available.
26	The device was not in the configuration file.
27	An error occurred while updating the configuration file.
28	An unknown host error occurred.
29	The sender contact information file was not found.
30	The sender contact information file could not be read.
31	The <code>userdata.txt</code> file exists.
32	An invalid <code>-I</code> value in the email alert notification was specified.
33	An invalid <code>-f</code> value in the email alert notification was specified.

Usage Examples

The following examples show how to enter CLI commands on a command line. The examples show the syntax, the form, and, in some examples, script commands. Examples are shown for both the Windows operating system and the UNIX operating system. Note that the usage for the `-c` terminal varies depending on your operating system. On Windows operating systems, enclose the script command following the `-c` terminal in double quotation marks (“ ”). On UNIX operating systems, enclose the script command following the `-c` terminal in single quotation marks (‘ ’). (For descriptions of the script commands used in these examples, see “[Chapter 8, Script Commands](#).”)

This example shows how to change the name of a storage array. The original name of the storage array is `Payroll_Array`. The new name is `Finance_Array`.

Windows operating system:

```
smcli ICTSANT -n "Payroll_Array" -c "set storageArray  
userLabel=\"Finance_Array\";"
```

UNIX operating system:

```
smecli ICTSANT -n 'Payroll_Array' -c 'set storageArray
userLabel="Finance_Array";'
```

This example shows how to delete an existing volume and create a new volume on a storage array. The existing volume name is `Stocks_<_Bonds`. The new volume name is `Finance`. The controller host names are `finance1` and `finance2`. The storage array is protected, requiring the password `TestArray`.

Windows operating system:

```
smcli finance1 finance2 -c "set session password=\"TestArray\"; delete volume
[\"Stocks_<_Bonds\"]; create volume driveCount[3] RAIDLEVEL=3 capacity=10
GB userLabel=\"Finance\"; show storageArray healthStatus;"
```

UNIX operating system:

```
smecli finance1 finance2 -c 'set session password="TestArray"; delete volume
["Stocks_<Bonds"]; create volume driveCount[3] RAIDLEVEL=3 capacity=10 GB
userLabel="Finance"; show storageArray healthStatus;'
```

This example shows how to run commands in a script file named `scriptfile.scr` on a storage array named `Example`. The `-e` terminal causes the file to run without checking the syntax. Running a script file without checking the syntax lets the file run more quickly; however, the file might not run correctly because the syntax for a command might be incorrect.

```
[smcli | smecli] -n Example -f scriptfile.scr -e
```

This example shows how to run commands in a script file named `scriptfile.scr` on a storage array named `Example`. In this example, the storage array is protected by the password `MyArray`. Output, as a result of commands in the script file, goes to file `output.txt`.

Windows operating system:

```
smcli -n Example -f scriptfile.scr -p "My_Array" -o output.txt
```

UNIX operating system:

```
smeecli -n Example -f scriptfile.scr -p 'My_Array' -o output.txt
```

This example shows how to show all of the storage arrays in the current configuration. The command in this example returns the host name of each storage array.

```
[smcli | smeecli] -d
```

If you want to know the IP address of each storage array in the configuration, add the `-i` terminal to the command.

```
[smcli | smeecli] -d -i
```

About Script Commands

You can use the script commands to configure and manage a storage array. The script commands are distinct from the command line interface (CLI) commands, and you enter the script commands using the CLI. You can enter individual script commands, or you can run a file of script commands. When you enter an individual script command, you include it as part of a CLI command. When you run a file of script commands, you include the file name as part of a CLI command. The script commands are processed by a script engine that performs the following functions:

- Verifies the command syntax
- Interprets the commands
- Converts the commands to the appropriate protocol-compliant commands
- Passes the commands to the storage array

At the storage array, the storage array controllers run script commands.

The script engine and the script commands support the storage array configuration and management operations that are listed in [Table 4](#).

Table 4 Configuration and Management Operations

Operation	Activities
Cache configuration	Controlling all cache parameters, both at the storage array level and the individual volume level
Volume configuration and volume group configuration	Creating, deleting, and setting the reconstruction priority control; labeling; setting drive composition when creating volumes; setting the segment size; and setting the media scan control
Drive configuration	Assigning hot spares
Controller configuration	Defining volume ownership, changing mode settings, defining network settings, and setting host channel IDs

Operation	Activities
General storage array configuration	Resetting a configuration to defaults, labeling, checking the health status, setting the time of day, clearing the Event Log, and setting the media scan rate
NVS RAM configuration	Downloading and modifying the user configuration region at the bit level and the byte level, showing nonvolatile static random access memory (NVS RAM) values
Product identification	Retrieving the tray profile display data
Battery management	Setting the battery installation date
Firmware management	Downloading controller firmware, the environmental services monitor (ESM) firmware, and the drive firmware

Script Command Structure

All script commands have the following structure:

```
command operand-data (statement-data)
```

where *command* identifies the action to be performed, *operand-data* represents the objects associated with a storage array that you want to configure or manage, and *statement-data* provides the information needed to perform the command.

The syntax for *operand-data* is as follows:

```
(object-type | allobject-types | [qualifier] (object-type [identifier]  
{object-type [identifier]} | object-types [identifier-list]))
```

An object can be identified four ways: object type, all parameter prefix, square brackets, or a list of identifiers. Use an object type when the command is not referencing a specific object. The all parameter prefix means all objects of the specified type in the storage array (for example, allVolumes). To perform a command on a specific object, use square brackets to identify the object (for example, volume [engineering]). Specify a subset of objects with a list of identifiers in square brackets (for example, volumes [sales engineering marketing]). A qualifier is required if you want to include additional information to describe the objects.

Table 5 lists the object type and the identifiers that are associated with each object type.

Table 5 Object Types and Identifiers

Object Type	Identifier
controller	a or b
drive	Tray ID and slot ID
replacementDrive	Tray ID and slot ID
driveChannel	Drive channel identifier
host	User label
hostChannel	Host channel identifier
hostGroup	User label
hostPort	User label
remoteMirror	Primary volume user label
snapshot	Volume user label
storageArray	Not applicable
tray	Tray ID
volume	Volume user label or volume World Wide Identifier (WWID) (<i>set</i> command only)
volumeCopy	Target volume user label and, optionally, the source volume user label
volumeGroup	User label Valid characters are alphanumeric, a hyphen, and an underscore.

Statement data is in the form of:

- Parameter=value (such as `raidLevel=5`)
- Parameter-name (such as `batteryInstallDate`)
- Operation-name (such as `redundancyCheck`)

A user-defined entry (such as user label) is called a variable. In the syntax, it is shown in italic (such as *trayID* or *volumeGroupName*).

Script Command Synopsis

Because you can use the script commands to define and manage the different aspects of a storage array (such as host topology, drive configuration, controller configuration, volume definitions, and volume group definitions), the actual number of commands is extensive. The commands, however, fall into general categories that are reused when you apply the commands to the different aspects of a storage array. [Table 6](#) lists the general form of the script commands and provides a definition of each command.

Table 6 General Form of the Script Commands

Syntax	Description
<code>activate object {statement-data}</code>	Sets up the environment so that an operation can take place or performs the operation if the environment is already set up correctly.
<code>autoConfigure storageArray {statement-data}</code>	Automatically creates a configuration that is based on the parameters that are specified in the command.
<code>check object {statement-data}</code>	Starts an operation to report on errors in the object, which is a synchronous operation.
<code>clear object {statement-data}</code>	Discards the contents of some attribute of an object. This operation is destructive and cannot be reversed.
<code>create object {statement-data}</code>	Creates an object of the specified type.
<code>deactivate object {statement-data}</code>	Removes the environment for an operation.
<code>delete object</code>	Deletes a previously created object.
<code>diagnose object {statement-data}</code>	Runs a test and shows the results.
<code>disable object {statement-data}</code>	Prevents a feature from operating.
<code>download object {statement-data}</code>	Transfers data to the storage array or to the hardware that is associated with the storage array.
<code>enable object {statement-data}</code>	Sets a feature operate.
<code>recopy object {statement-data}</code>	Restarts a volume copy operation by using an existing volume copy pair. You can change the parameters before the operation is restarted.

Syntax	Description
<code>recover object {statement-data}</code>	Re-creates an object from saved configuration data and the statement parameters. (This command is similar to the <code>create</code> command.)
<code>recreate object {statement-data}</code>	Restarts a snapshot operation by using an existing snapshot volume. You can change the parameters before the operation is restarted.
<code>remove object {statement-data}</code>	Removes a relationship from between objects.
<code>repair object {statement-data}</code>	Repairs errors found by the <code>check</code> command.
<code>reset object {statement-data}</code>	Returns the hardware or an object to an initial state.
<code>resume object</code>	Starts a suspended operation. The operation starts where it left off when it was suspended.
<code>revive object</code>	Forces the object from the Failed state to the Optimal state. Use this command only as part of an error recovery procedure.
<code>save object {statement-data}</code>	Writes information about the object to a file.
<code>set object {statement-data}</code>	Changes object attributes. All changes are completed when the command returns.
<code>show object {statement-data}</code>	Shows information about the object.
<code>start object {statement-data}</code>	Starts an asynchronous operation. You can stop some operations after they have started. You can query the progress of some operations.
<code>stop object {statement-data}</code>	Stops an asynchronous operation.
<code>suspend object {statement-data}</code>	Stops an operation. You can then restart the suspended operation, and it continues from the point where it was suspended.

Recurring Syntax Elements

Recurring syntax elements are a general category of variables and options that you can use in one or more script commands. The recurring syntax is used in the general definitions of the script commands that are listed in “[Chapter 8, Script Commands](#).”

Selecting a controller or a RAID level is a choice made in many commands. This section describes syntax elements that repeat themselves. The following example shows the actual command entry that uses recurring syntax. This command creates a 2-GB, RAID 5, six-drive volume, which is owned by controller A and named Volume1.

```
create volume drives=(0,2 0,3 1,4 1,5 2,6 2,7) raidLevel=5
userLabel="Volume1" capacity=2 gb owner=a;
```

The syntax values provide the type or range of information that must be entered to obtain a valid command statement.

[Table 7](#) lists the recurring syntax and the syntax values that you can use with the syntax.

NOTE A command statement appears in a monospace font. The statement variables are in a *monospace italic* font. Options are described separately, and usually the choices are shown as **TRUE** or **FALSE**, as a letter (**a**), a number (**42**), or a number-range (**0-99**) selection.

Table 7 Recurring Syntax Elements

Recurring Syntax	Syntax Value
raid-level	(0 1 3 5 6)
repository-raid-level	(1 3 5 6)
capacity-spec	<i>integer-literal</i> [KB MB GB TB Bytes]
segment-size-spec	<i>integer-literal</i>
boolean	(TRUE FALSE)
user-label	<i>string-literal</i> Valid characters are alphanumeric, the dash, and the underscore.
user-label-list	<i>user-label</i> { <i>user-label</i> }
create-raid-vol-attr-value-list	<i>create-raid-volume-attribute-value-pair</i> { <i>create-raid-volume-attribute-value-pair</i> }
create-raid-volume-attribute-value-pair	capacity= <i>capacity-spec</i> owner=(a b) cacheReadPrefetch=(TRUE FALSE) segmentSize= <i>integer-literal</i> usageHint= <i>usage-hint-spec</i>
noncontroller-trayID	(0-99)
slotID	(1-32)
portID	(0-127)
drive-spec	<i>trayID</i> , <i>slotID</i>
drive-spec-list	<i>drive-spec</i> { <i>drive-spec</i> }
trayID-list	<i>trayID</i> { <i>trayID</i> }
esm-spec-list	<i>esm-spec</i> { <i>esm-spec</i> }
esm-spec	<i>trayID</i> , (left right)
hex-literal	<i>0xhexadecimal-literal</i>
volume-group-number	<i>integer-literal</i>
filename	<i>string-literal</i>
error-action	(stop continue)

Recurring Syntax	Syntax Value
drive-channel-identifier (four drive ports per tray)	(1 2 3 4)
drive-channel-identifier (eight drive ports per tray)	(1 2 3 4 5 6 7 8)
drive-channel-identifier-list	<i>drive-channel-identifier</i> { <i>drive-channel-identifier</i> }
host-channel-identifier (four host ports per tray)	(a1 a2 b1 b2)
host-channel-identifier (eight host ports per tray)	(a1 a2 a3 a4 b1 b2 b3 b4)
drive-type	(fibre SATA SAS PATA)
feature-identifier	(storagePartition2 storagePartition4 storagePartition8 storagePartition16 storagePartition64 storagePartitionMax snapshot remoteMirror32 remoteMirror64 remoteMirror128 volumeCopy goldKey mixedDriveTypes)
repository-spec	<i>instance-based-repository-spec</i> <i>count-based-repository-spec</i>
instance-based-repository-spec	(repositoryRAIDLevel= <i>repository-raid-level</i> repositoryDrives=(<i>drive-spec-list</i>) [repositoryVolumeGroupUserLabel= <i>user-label</i>] [trayLossProtect= <i>boolean</i>]) (repositoryVolumeGroup= <i>user-label</i> [freeCapacityArea= <i>integer-literal</i>]) ² Specify the repositoryRAIDLevel parameter with the repositoryDrives parameter. Do not specify the RAID level or the drives with the volume group. Do not set a value for the trayLossProtect parameter when you specify a volume group.

Recurring Syntax	Syntax Value
count-based-repository-spec	<pre>repositoryRAIDLevel=repository-raid-level repositoryDriveCount=integer-literal [repositoryVolumeGroupUserLabel=user-label] [driveType=drive-type³] [trayLossProtect=boolean⁴]</pre>
wwID	<i>string-literal</i>
gid	<i>string-literal</i>
nvsram-offset	<i>hex-literal</i>
host-type	<i>string-literal</i> <i>integer-literal</i>
nvsram-byte-setting	<pre>nvsram-value = 0hexadecimal integer-literal</pre> <p>The 0hexadecimal value is typically a value from 0x0000 to 0xFFFF.</p>
nvsram-bit-setting	<pre>nvsram-mask, nvsram-value = 0hexadecimal, 0hexadecimal integer-literal</pre> <p>The 0hexadecimal value is typically a value from 0x0000 to 0xFFFF.</p>
ip-address	(0-255) . (0-255) . (0-255) . (0-255)
autoconfigure-vols-attr-value-list	<pre>autoconfigure-vols-attr-value-pair {autoconfigure-vols-attr-value-pair}</pre>
autoconfigure-vols-attr-value-pair	<pre>driveType=drive-type⁵ raidLevel=raid-level volumeGroupWidth=integer-literal volumeGroupCount=integer-literal volumesPerGroupCount=integer-literal⁶ hotSpareCount=integer-literal segmentSize=segment-size-spec cacheReadPrefetch=(TRUE FALSE)</pre>
create-volume-copy-attr-value-list	<pre>create-volume-copy-attr-value-pair {create-volume-copy-attr-value-pair}</pre>
create-volume-copy-attr-value-pair	<pre>copyPriority=(highest high medium low lowest) targetReadOnlyEnabled=boolean</pre>

Recurring Syntax	Syntax Value
recover-raid-volume-attr-value-list	<i>recover-raid-volume-attr-value-pair</i> { <i>recover-raid-volume-attr-value-pair</i> }
recover-raid-volume-attr-value-pair	owner=(a b) cacheReadPrefetch=(TRUE FALSE)
cache-flush-modifier-setting	immediate, 0, .25, .5, .75, 1, 1.5, 2, 5, 10, 20, 60, 120, 300, 1200, 3600, infinite
serial-number	string-literal
usage-hint-spec	usageHint=(multiMedia database fileSystem)
ethernet-port-options	IPv4Address= <i>ip-address</i> IPv4ConfigurationMethod=[(static dhcp)] IPv4GatewayIP= <i>ip-address</i> IPv4SubnetMask= <i>ip-address</i>
test-devices-list	<i>test-devices</i> { <i>test-devices</i> }
test-devices	controller=(a b) esms=(<i>esm-spec-list</i>) drives=(<i>drive-spec-list</i>)

The following considerations apply when using the recurring syntax with script commands:

- 1 For tray loss protection to work, each drive in a volume group must be in a separate tray. If you set the `trayLossProtect` parameter to **TRUE** and you have selected more than one drive from any one tray, the storage array returns an error. If you set `trayLossProtect` parameter to **FALSE**, the storage array performs operations, but the volume group that you create might not have tray loss protection.
- 2 To determine if a free capacity area exists, run the `show volumeGroup` command.
- 3 The default drive (drive type) is **fibre** (Fibre Channel).
- 4 If you set the `trayLossProtect` parameter to **TRUE**, the storage array returns an error if the controller firmware cannot find drives that will enable the new volume group to have tray loss protection. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs the operation even if it means that the volume group might not have tray loss protection.

- 5 The `driveType` parameter is not required if only one type of drive is in the storage array. If you use the `driveType` parameter, you must also use the `hotSpareCount` parameter and the `volumeGroupWidth` parameter. If you do not use the `driveType` parameter, the configuration defaults to Fibre Channel drives.
- 6 The `volumesPerGroupCount` parameter is the number of equal-capacity volumes per volume group.

Usage Guidelines

This list provides guidelines for writing script commands on the command line:

- You must end all commands with a semicolon (;).
- You can enter more than one command on a line, but you must separate each command with a semicolon (;).
- You must separate each base command and its associated primary parameters and secondary parameters with a space.
- The script engine is not case sensitive. You can enter commands by using uppercase letters, lowercase letters, or mixed-case letters.
- Add comments to your scripts to make it easier for you and future users to understand the purpose of the script commands.

NOTE While the CLI commands and the script commands are not case sensitive, user labels (such as for volumes, hosts, or host ports) are case sensitive. If you try to map to an object that is identified by a user label, you must enter the user label exactly as it is defined, or the CLI commands and the script commands will fail.

Adding Comments to a Script File

The script engine looks for certain characters or a command to show comments. You can add comments to a script file in three ways:

- 1 Add text after two forward slashes (//) as a comment until an end-of-line character is reached. If the script engine does not find an end-of-line character in the script after processing a comment, an error message appears, and the script operation is terminated. This error usually occurs when a comment is placed at the end of a script and you have forgotten to press the **Enter** key.

```
// Deletes the existing configuration.  
set storageArray resetConfiguration=true;
```

- 2 Add text between /* and */ as a comment. If the script engine does not find both a starting comment notation and an ending comment notation, an error message appears, and the script operation is terminated.

```
/* Deletes the existing configuration */  
set storageArray resetConfiguration=true;
```

- 3 Use the show statement to embed comments in a script file that you want to appear while the script file is running. Enclose the text that you want to appear by using double quotation marks (“ ”).

```
show "Deletes the existing configuration";  
set storageArray resetConfiguration=true;
```


Configuring a Storage Array

Before using script commands to manage a storage array, you should be familiar with these items:

- Controllers
- Drives
- Hot spares
- Volume groups
- Volumes
- RAID technology
- Hosts
- Host groups
- Host bus adapter (HBA) host ports
- Logical unit numbers (LUNs)

Configuring a RAID storage array requires caution and planning to make sure that you define the correct RAID level and configuration for your storage array. You configure a storage array to create volumes, which are addressable by the hosts, from a collection of drives. The commands described in this chapter help you to set up and run a RAID storage array. Additional commands are also available to provide you with more control and flexibility.

NOTE Many of these commands require a thorough understanding of the firmware as well as an understanding of the network components that need to be mapped. Use the CLI commands and the script commands with caution.

The sections in this chapter show some, but not all, of the CLI commands and the script commands. The commands in this chapter show how you can use the commands to configure a storage array. These presentations do not describe all possible usage and syntax for the commands. (For complete definitions of the commands, including syntax, parameters, and usage notes, see “[Chapter 8, Script Commands.](#)”)

This chapter contains examples of CLI command usage and script command usage. The command syntax that is used in the examples is for a host running a Microsoft operating system. As part of the examples, the complete `C:\` prompt and the DOS path for the commands are shown. Depending on your operating system, the prompt and path construct will vary.

For most commands, the syntax is the same for all Windows operating systems and UNIX operating systems, as well as for a script file. Windows operating systems, however, have an additional requirement when entering names in a command. On Windows operating systems, you must enclose the name between two back slashes (`\ \`) in addition to other delimiters. For example, the following name is used in a command running under a Windows operating system:

```
[\"Engineering\"]
```

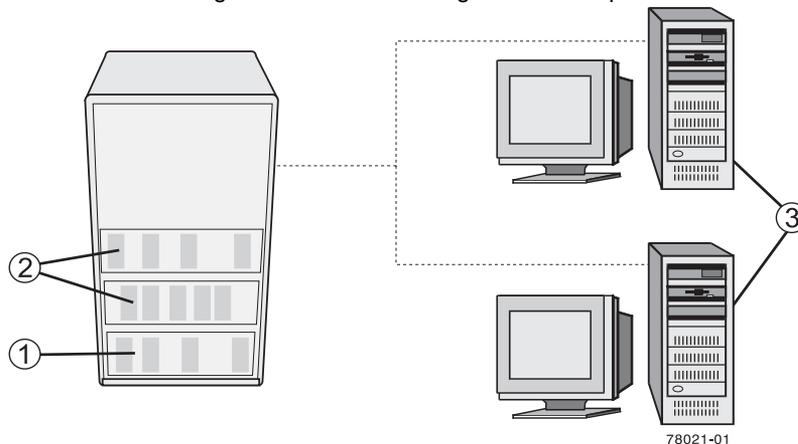
For a UNIX operating system, and when used in a script file, the name appears as follows:

```
[\"Engineering\"]
```

Configuration Concepts

[Figure 1](#) shows a host using a computer system [3], and shows the storage array with a controller tray [1] and two drive trays [2]. The storage array has at least one controller and drives. The drives are grouped together to form a logical structure. The storage array is physically placed in a cabinet as shown on the left in [Figure 1](#).

Figure 1 Host and Storage Relationship



When you configure a storage array, you allocate parts of this logical structure for use by one or more of the hosts that are attached to the storage array. Each drive and drive tray is numbered to allow the storage to be organized into volume groups and volumes. This organization helps to provide storage capacity and data protection so that the hosts can safely store and retrieve data from the storage array.

This section defines the physical components and the logical components that are required to organize and configure a storage array. This section also describes how the components relate to each other.

Controllers

All storage arrays have one or two controllers. The controllers are circuit-board assemblies that manage data and communication between the hosts and the storage array. The controller manages the data flow between the hosts and the drives, keeping track of the logical address of where the data resides. In general, each controller has a processor for performing control operations, NVSRAM for storing the firmware code that operates the storage array, and the buses along which the data flows.

The controllers are located in a controller tray or a controller-drive tray, which provides two positions for controllers: slot A and slot B. The script commands identify each controller by the slot in which the controller is installed. If a storage array has only one controller, the controller must be in slot A. Controllers are identified by using model numbers.

TP9400 and TP9500 controller models used minihubs; two connected to each controller. When viewed from the rear of the controller tray, the host-side minihubs are numbered from left-to-right a1, b1, a2, b2. The script commands identify the host channels by using these identifiers. Minihubs also supported the drive-side, where each minihub represents a single channel to the drives. When viewed from the rear of the controller tray, the drive minihubs are numbered from left to right 4, 3, 2, 1. The script commands use these numbers to identify the drive channels.

TP9300 storage arrays used in an early controller-drive tray that has a slot where either a controller or an environmental services monitor (ESM) can be used. When an ESM is used, the tray is called a drive tray.

Controllers manage the interface by using the controller firmware to transmit and receive commands between the hosts and the drives. Host bus adapters facilitate the communication through whatever interface is selected. Typically, two host bus adapters and two paths are used to optimize redundancy.

These controller-drive trays and controller trays incorporate all host connections and drive tray connections into each controller. The host ports must be identified in your command statements to let you complete their network configurations.

The more recent controllers do not use minihubs. These controllers have host ports that are integrated into the controller circuit boards or auxiliary circuit boards that are directly connected to the controller circuit boards.

[Table 8](#) lists the controller-drive trays that do not use minihubs, the type of host port, and the number of host ports.

Table 8 Host Ports and the Type of Host Interfaces on Controller-Drive Trays

Model	Available Host Ports	Type of Host Interface
IS350/IS4000 controller tray	4	Fibre Channel
IS4000 controller tray	8	Fibre Channel
IS4500 controller tray	8	Fibre Channel
IS4600 controller tray	16	Fibre Channel

The host ports on the IS4000 controller-drive tray are numbered from left-to-right on controller B as Ch 1, Ch 2, Ch 3, and Ch 4. Controller A, which is installed upside-down in the enclosure, is numbered from right-to-left in the same sequence.

The controller in the IS4500 controller tray can have up to eight host channels with one port for each channel; up to four drive channels with two ports per channel (for a total of eight drive ports); and up to two Ethernet ports. In the IS4500 controller tray, the controllers are stacked one above the other. The top controller is A. The bottom controller is B. The controller in the IS4600 controller tray can have up to 16 host channels with one port for each channel; up to 8 drive channels with two ports per channel (for a total of 16 drive ports); and up to two Ethernet ports. In the IS4600 controller tray, the controllers are stacked one above the other. The top controller is A. The bottom controller is B.

For controller A, the host channel identifiers are a1, a2, a3, a4, etc. (you will need this reference for the CLI commands and the script commands). On the controller, the host bus adapter (HBA) host ports are labeled 1, 2, 3, and 4. For controller B, the host channel identifiers are b1, b2, b3, b4, etc. On the controller, the HBA host ports are labeled 1, 2, 3, 4, etc.

Controller A has drive channels 1 and 2. Drive ports labeled 3 and 4 connect to drive channel 1. Drive ports labeled 1 and 2 connect to Drive Channel 2. Controller B has drive channels 3 and 4. Drive ports labeled 1 and 2 connect to drive channel 3. Drive ports labeled 3 and 4 connect to drive channel 4. Each Ethernet port on a controller can have a unique IP address; however, both Ethernet ports share the same gateway IP address, subnet mask, and remote login settings.

Drives

The drives provide the actual storage of the data that is sent to the storage array. The drives are mounted in a drive tray. The drive tray is a tray that, in addition to the drives, contains power supplies, fans, environmental cards, and other supporting components.

Drives are located in a storage array by tray ID and slot ID. Tray ID values are **0** to **99**. In older trays, tray ID values are set during installation by switches on the rear of the trays. In newer trays, tray ID values are set automatically when the power is applied.

The slot ID is the drive position in the drive tray. Slot ID values range from **1** to **32**. A drive tray can contain either 10, 12, 14, or 16 drives. The total number of drives in a storage array depends on the model of the controller and the type of drive tray. [Table 9](#) lists, by controller number and drive tray capacity, the maximum number of drives in a storage array.

Table 9 Maximum Number of Drives

Controller Model	10-Drive Drive Tray	14-Drive Drive Tray	16-Drive Drive Tray
TP9300 Simplex		14	
TP9300 Duplex		112	
IS350/IS4000 4 FC ports		112	112
IS4000 4 FC ports		112	112
TP9400 8 FC ports	220	224	
TP9500 8 FC ports	220	224	224
TP9700 8 FC ports		224	224
IS4600			226

NOTE A maximum of seven drive trays can be on a channel when mixing 14-slot drive trays and 16-slot drive trays.

Hot Spares

A hot spare is a drive that acts as a standby in the event that a drive containing data fails. The hot spare is a drive that has not been assigned to a particular volume group and, as such, can be used in any volume group. You can use the hot spare feature with RAID 1, RAID 3, RAID 5, or RAID 6. If a drive in a volume group fails, the controllers automatically replace the failed drive with a hot spare. The controllers use redundancy data to reconstruct the data from the failed drive onto the hot spare. To be most effective, the drive that you assign as a hot spare must have a capacity equal to or greater than the capacity of the largest drive in the storage array. The hot spare must be the same type of drive as the drive that failed (for example, a Serial Advanced Technology Attachment [SATA] hot spare cannot replace a Fibre Channel hot spare).

You can assign drives to act as hot spares manually or have the script commands automatically assign hot spares. If you manually assign a drive to be a hot spare, you must identify the drive by tray ID and slot ID. When you let the script commands automatically assign hot spares, you must enter the number of hot spares that you want in the storage array.

Volume Groups

A volume group is a set of drives that are logically grouped together by the controllers in a storage array. After you create a volume group, you can create one or more volumes in the volume group. A volume group is identified by a sequence number that is defined by the controller firmware when you created the volume group.

To create a volume group, you must define the capacity and the RAID level.

Capacity is the size of the volume group. Capacity is determined by the number of drives that you assign to the volume group. You can use only unassigned drives to create a volume group. (In this programming guide, the storage space on unassigned drives constitutes the *unconfigured capacity* of a storage array.)

Free capacity is a contiguous region of unassigned capacity in a designated volume group. Before you create a new volume in a volume group, you need to know the free capacity space so that you can determine the size of the volume.

The number of drives you can include in a volume group is constrained by the capacity of each drive. [Table 10](#) relates the capacity of individual drives to the number of drives that you can include in a volume group.

Table 10 Maximum Number of Drives in a Volume Group Based on Capacity

Capacity of the Drive	Maximum Number of Drives in the Volume Group
18 GB	30
36 GB	30
73 GB	28
180 GB	11

NOTE Some of the drive capacity contains metadata information, so the actual usable capacity is somewhat less. Also some operating systems and some storage arrays limit the size of the volume group to 2 TB.

You can determine the size of the volume group by multiplying the maximum number of drives in the volume group by the capacity of the smallest drive in the volume group.

NOTE Some storage arrays permit different drive types in the same tray; however, you cannot have a combination of different drives in the same volume group.

The RAID level is the level of data protection that you want to define for your storage array. The RAID level that you choose affects storage capacity. When you configure your storage array, you must consider this compromise between data protection and storage capacity. In general, the more protection that you need, the less storage capacity is available in your storage array.

Volumes

A volume is a logical component (object) that is the basic structure that is created on the storage array to store data. A volume is a contiguous subsection of a volume group that is configured to meet application needs for data availability and I/O performance. The ISSM-EE software administers a volume as if the volume is one “drive” for data storage. Volumes are identified by names or labels that users choose. The volume names can be any combination of alphanumeric characters, hyphens (-), and underscores (_). The maximum length of a volume name is 30 characters.

The script commands support the following types of volumes:

- **Standard volume** – A logical structure that is the principal type of volume for data storage. A standard volume is the most common type of volume in a storage array.
- **Access volume** – A factory-configured volume in a storage area network (SAN) environment that is used for communication between the ISSM-EE software and the storage array controller. The access volume uses a logical unit number (LUN) address and consumes 20 MB of storage space. The 20 MB of access volume storage space is not available for data storage.

NOTE You must use the access volume only for in-band-managed storage arrays.

- **Snapshot volume** – A logical point-in-time image of another volume. A snapshot volume is the logical equivalent of a complete physical copy; however, it is not an actual, physical copy. Instead, the firmware tracks only the data blocks that are overwritten and copies those blocks to a snapshot repository volume.
- **Snapshot repository volume** – A special volume in the storage array that is created as a resource for a snapshot volume. A snapshot repository volume contains snapshot data and copy-on-write data for a particular snapshot volume.
- **Base volume** – A standard volume from which you create a snapshot volume. The term “base volume” is used only to show the relationship between a standard volume from which you are taking the point-in-time image and a snapshot volume.

- **Primary volume** – A standard volume in a Remote Volume Mirroring relationship. The primary volume accepts host data transfers and stores application data. When you first create the mirror relationship, data from the primary volume is copied in its entirety to the associated secondary volume.
- **Secondary volume** – A standard volume in a Remote Volume Mirroring relationship that maintains a mirror (or copy) of the data from its associated primary volume. The secondary volume remains unavailable to host applications while mirroring is underway. In the event of a disaster or a catastrophic failure of the primary site, a system administrator can promote the secondary volume to a primary role.
- **Mirror repository volume** – A special volume in a Remote Volume Mirroring configuration that is created as a resource for each controller in both the local storage array and the remote storage array. The controller stores mirroring information on this volume, including information about remote writes that are not yet complete. A controller can use this information to recover from controller resets and accidental power shutdown of the storage arrays.

NOTE Snapshot Volume and Remote Volume Mirroring are premium features that you must activate before you can use them. For more information about snapshot volumes, see [“Chapter 4, Using the Snapshot Premium Feature.”](#) For more information about Remote Volume Mirroring, see [“Chapter 5, Using the Remote Volume Mirroring Premium Feature.”](#)

The number and capacity of the volumes in your storage array depends on the type of controller in the storage array. [Table 11](#) lists the maximum values for the volumes in a storage array by specific controllers.

Table 11 Volume Specifications by Supported Controller Model

Specification	IS4000	IS4500
Maximum number of volumes per storage array	1024	2048
Maximum number of volumes per volume group	256	256
Maximum volume size (less 12 GB for overhead)	2 TB	2 TB
Maximum number of drives per volume group using RAID 5	30	30
Maximum number of remote mirrors	64	128

NOTE Addressable capacities greater than 2 TB are possible with some operating systems. Currently these operating systems are Windows, Solaris 9, Solaris 10, AIX 5.2, AIX 5.3, HP-UX 11.23, and Linux 2.6.

A Windows operating system and a Linux 2.6 operating system using 500-GB drives can have a maximum volume size of 112 TB (224 x 500).

RAID Levels

The RAID level defines a storage architecture in which the storage capacity on the drives in a volume group is separated into two parts: part of the capacity stores the user data, and the remainder stores redundant or parity information about the user data. The RAID level that you choose determines how user data is written to and retrieved from the drives. Using the script commands, you can define five RAID levels: RAID 0, RAID 1, RAID 3, RAID 5, and RAID 6. Each level provides different performance and protection features.

RAID 0 provides the fastest storage access but does not provide any redundant information about the stored data. RAID 1, RAID 3, RAID 5, and RAID 6 write redundancy information to the drives to provide fault tolerance. The redundancy information might be a copy of the data or an error-correcting code that is derived from the data. In RAID 1, RAID 3, RAID 5, or RAID 6 configurations, if a drive fails, the redundancy information can be used to reconstruct the lost data. Regardless of the RAID level that you choose, you can configure only one RAID level

across each volume group. All redundancy information for a volume group is stored within the volume group. [Table 12](#) lists the RAID levels and describes the configuration capabilities of each level.

Table 12 RAID Level Configurations

RAID Level	Configuration
0	<p>Non-redundant striping mode – Use this level for high-performance needs. RAID 0 does not provide any data redundancy. RAID 0 stripes data across all of the drives in the volume group. If a single drive fails, all of the associated volumes fail and all data is lost. RAID 0 is suited for noncritical data. It is not recommended for high-availability needs.</p>
1	<p>Striping mirroring mode – RAID 1 uses drive mirroring to create an exact copy from one drive to another drive. A minimum of two drives are required; one for the user data, and one for the mirrored data. RAID 1 offers high performance and the best data availability.</p> <p>Data is written to two drives simultaneously. If one drive in a pair fails, the system can instantly switch to the other drive without any loss of data or service. Only half of the drives in the volume group are available for user data. If a single drive fails in a RAID 1 volume group, all of the associated volumes become degraded, but the mirror drive provides access to the data. RAID 1 can survive multiple drive failures as long as no more than one failure occurs per mirrored pair. If a drive pair fails, all of the associated volumes fail, and all data is lost.</p>

RAID Level	Configuration
3	<p>High-bandwidth mode – RAID 3 stripes both user data and redundancy data (in the form of parity) across the drives. The equivalent of the capacity of one drive is used for the redundancy data. RAID 3 works well for large data transfers in applications, such as multimedia or medical imaging, that write and read large sequential chunks of data.</p> <p>If a single drive fails in a RAID 3 volume group, all of the associated volumes become degraded, but the redundancy data lets the data be reconstructed. If two or more drives fail, all of the associated volumes fail, and all data is lost.</p>
5	<p>High I/O mode – RAID 5 stripes both user data and redundancy data (in the form of parity) across the drives. The equivalent of the capacity of one drive is used for the redundancy data. RAID 5 works well for multiuser environments, such as databases or file system storage, where typical I/O size is small, and a high proportion of read activity exists.</p> <p>If a single drive fails in a RAID 5 volume group, all of the associated volumes become degraded, and the redundancy data permits the data to be reconstructed. If two or more drives fail, all of the associated volumes fail, and all data is lost.</p>
6	<p>Data protection or continuous access mode – RAID 6 stripes both user data and redundancy data (in the form of parity) across the drives. A minimum of five drives are required for a RAID 6 volume group. The equivalent capacity of two drives is used for the redundancy data. Two different algorithms calculate redundancy data, which are in the form of both a P parity and a Q parity.</p> <p>RAID 6 works well for larger drive sizes. Recovery from a second drive failure in the same volume group is possible. If two drives fail in a RAID 6 volume group, all of the associated volumes become degraded, but the redundancy data permits the data to be reconstructed. If three or more drives fail, all of the associated volumes fail, and all data is lost.</p>

NOTE RAID 6 is only available to those controllers that are capable of supporting the P+Q calculation. IS4000 and IS350 controllers are supported. See “[Chapter 8, Set Volume Group](#)” on page 290 to set your volume group to RAID 6.

Hosts

A host is a computer that is attached to the storage array for accessing the volumes in the storage array. The host is attached to the storage array through HBA host ports, which are connectors on host bus adapter circuit boards. You can define specific volume-to-LUN mappings to an individual host or assign the host to a host group that shares access to one or more volumes. Hosts are identified by names or labels that users choose. The host name can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the host name is 30 characters.

In addition to a host name, some script commands require you to identify a host by its “type.” A host type identifies the operating system under which the host is running (such as Windows or Linux). Specifying the host type lets the controllers in the storage array adapt their behavior (such as LUN reporting and error conditions) to the operating system of the host that is sending the information. Host types are identified by a label or an index number that is generated by the controller firmware.

Host Bus Adapter Host Ports

A host bus adapter (HBA) provides the physical connection from the host to the storage array. The host port is a physical connector on an HBA. The HBA is a circuit board that is installed in the host. The HBA can have one or more host ports. Each host port is identified by a unique, 16-byte World Wide Identifier (WWID). If the HBA has more than one host port, each host port has a unique ID.

When you first turn on the power to a storage array, the ISSM-EE software automatically detects the HBA host ports. Initially, all detected host ports belong to a default group. You can use script commands to identify the WWIDs on a storage array and, if you choose, change them. If you move an HBA host port, you must remap any volume-to-LUN mappings. Access to your data is lost until you remap the volumes.

The maximum number of HBA host ports that you can logically define for your storage array depends on the type of controller in the storage array. [Table 13](#) lists the maximum number of HBA host ports that you can define.

Table 13 Maximum Number of HBA Host Ports per Controller

Controller Models	Maximum Number of Host Ports
TP9300 Simplex	256
TP9300 Duplex	256
IS350/IS4000	256
TP9400	512
TP9500	512
TP9700	512
IS4500	512
IS4600	512

Host Groups

A host group is a topological element that you can define if you want to designate a collection of hosts that will share access to the same volumes. A host group is a logical entity. Host groups are identified by names or labels that users choose. The host group name can be any combination of alphanumeric characters with a maximum length of 30 characters.

Logical Unit Numbers

In the context of the CLI commands and the script commands, a logical unit number (LUN) is a unique value that identifies the volumes in a storage array. The hosts identify the volumes that they want to access using the LUN values. When you create a volume, the firmware assigns the LUN values, or you can assign LUN values when you enable the Storage Partitioning premium feature. A volume can have only one LUN and can be mapped to only one host or host group. Each host has unique addressing capability. That is, when more than one host accesses a storage array, each host might use the same LUN to access different volumes. The LUNs might be the same, but the volumes are different. If you are mapping to a host group, the LUN that you specify must be available on every host in the host group.

Configuring a Storage Array

When you configure a storage array, you want to maximize the data availability by making sure that the data is quickly accessible while maintaining the highest level of data protection possible. The speed by which a host can access data is affected by these items:

- The RAID level for the volume group
- The settings for the segment size and the cache size
- Whether the cache read prefetch capability is turned on or turned off

Data protection is determined by the RAID level, hardware redundancy (such as global hot spares), and software redundancy (such as the Remote Volume Mirroring premium feature and the Snapshot Volume premium feature).

In general, you configure a storage array by defining a volume group and its associated RAID level, defining the volumes, and defining which hosts have access to the volumes. This section explains how to use the script commands to perform the general steps to create a configuration from an array of drives.

Determining What Is on Your Storage Array

Even when you create a configuration on a storage array that has never been configured, you still need to determine the hardware features and software features that are to be included with the storage array. When you configure a storage array that has an existing configuration, you must make sure that your new configuration does not inadvertently alter the existing configuration, unless you are reconfiguring the entire storage array. For example, consider the case where you want to create a new volume group on unassigned drives. Before you create a new volume group, you must determine which drives are available. The commands that are described in this section help you to determine the components and the features in your storage array.

The command that returns general information about the storage array is the `show storageArray` command. This command returns information about the components and properties of your storage array, including these items:

- A detailed profile of the components and features in the storage array
- The age of the battery
- The default host type (which is the current host type)

- Other host types that you can select
- The hot spare locations
- The identifiers for enabled features
- The logical component profiles and the physical component profiles
- The time to which both controllers are set
- The controller that currently owns each volume in the storage array

To return the most information about the storage array, run the `show storageArray` command with the `profile` parameter. This example shows the complete CLI command and script command running on a Windows operating system:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storageArray profile;"
```

This example identifies the storage array by the example IP address 123.45.67.89. You can also identify the storage array by name.

The `show storageArray profile` command returns detailed information about the storage array. The information appears in several display screens. You might need to increase the size of your display buffer to see all of the information. Because this information is so detailed, you might want to save the output to a file. To save the output to a file, enter the command as shown in this example:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storageArray profile;" -o  
c:\folder\storagearrayprofile.txt
```

In this example, the name `folder` is the folder in which you choose to place the profile file, and `storagearrayprofile.txt` is the name of the file. You can choose any folder and any file name.

ATTENTION **Possible loss of data** – When you are writing information to a file, the script engine does not check to determine if the file name already exists. If you choose the name of a file that already exists, the script engine writes over the information in the file without warning.

“, [Examples of Information Returned by the Show Commands](#),” shows the type of information returned. When you save the information to a file, you can use the information as a record of your configuration and as an aid during recovery.

To return a brief list of the storage array features and components, use the summary parameter. The command looks like this example:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storageArray summary;"
```

Following is the type of information that is returned by the show storageArray command with the summary parameter.

```
PROFILE FOR STORAGE ARRAY: example9 (X/XX/XX 9:58:32 AM)
SUMMARY-----
Number of controllers: 1
Number of volume groups: 2
Total number of volumes (includes an access volume): 3 of 2048 used
    Number of standard volumes: 2
    Number of access volumes: 1
    Number of snapshot repositories: 0
Number of snapshot volumes: 0
Number of copies: 0
Number of drives: 9
Supported drive types: Fibre (9)
Total hot spare drives: 0
    Standby: 0
    In use: 0
Access volume: LUN 7 (see Mappings section for details)
Default host type: Linux (Host type index 6)
Current configuration
    Firmware version: PkgInfo 96.10.21.00
    NVSRAM version: N4884-610800-001
Pending configuration
    Staged firmware download supported?: No
    Firmware version: Not applicable
    NVSRAM version: Not applicable
    Transferred on: Not applicable
NVSRAM configured for batteries?: Yes
Start cache flushing at (in percentage): 80
Stop cache flushing at (in percentage): 80
Cache block size (in KB): 4
Media scan frequency (in days): Disabled
Failover alert delay (in minutes): 5
Feature enable identifier: 1234567891011121314151617181ABCD
```

The summary information is also returned as the first section of information when you use the profile parameter.

The show commands return information about the specific components of a storage array. The information returned by each of the show commands is the same as the information returned by the show storageArray profile command, but the information is constrained to the specific component. For information about a command, refer to the page number shown in parentheses next to the command.

- `show controller` (page [292](#))
- `show drive` (page [354](#))
- `show driveChannels stats` (page [297](#))
- `show storageArray hostTopology` (page [309](#))
- `show storageArray lunMappings` (page [311](#))
- `show allVolumes` (page [362](#))
- `show volumeGroup` (page [317](#))
- `show volume reservations` (page [322](#))

In addition, these commands return information about a storage array:

- `show controller NVSRAM` (page [296](#))
- `show remoteMirror candidates` (page [300](#))
- `show storageArray autoConfigure` (page [306](#))
- `show storageArray unreadableSectors` (page [312](#))
- `show volumeCopy sourceCandidates` (page [316](#))
- `show volumeCopy targetCandidates` (page [317](#))
- `show volume performanceStat` (page [321](#))

There are other commands that can help you learn about your storage array. To see a list of the commands, see “[Chapter 8, Commands Listed Alphabetically](#)” on page [159](#). These commands are organized by the storage array activities that the commands support, such as volume commands, host commands, tray commands, and others.

Clearing the Configuration

If you want to create a completely new configuration on a storage array that already has an existing configuration, use the `clear storageArray configuration` command. This command deletes all of the existing configuration information, including all of the volume groups, volumes, and hot spare definitions from the controller memory.

ATTENTION Possible damage to the storage array configuration – As soon as you run this command, the existing storage array configuration is deleted.

The command takes this form:

```
c:\...\smee\client>smcli 123.45.67.89 -c "clear storageArray configuration;"
```

This command has two parameters that you can use to limit the amount of configuration information removed:

- `all` – Removes the entire configuration of the storage array, including security information and identification information. Removing all of the configuration information returns the storage array to its initial state.
- `volumeGroups` – Removes the volume configuration and the volume group configuration, but leaves the rest of the configuration intact.

If you want to create new volume groups and volumes within the storage array, you can use the `clear storageArray configuration` command with the `volumeGroups` parameter to remove existing volume groups in a pre-existing configuration. This action destroys the pre-existing configuration. Use the `clear storageArray configuration` command only when you create a new configuration.

Using the Auto Configure Command

The `autoConfigure storageArray` command creates the volume groups on a storage array, the volumes in the volume groups, and the hot spares for the storage array. When you use the `autoConfigure storageArray` command, you define these parameters:

- The type of drives (Fibre, SATA, SAS, or Parallel Advanced Technology Attachment [PATA])
- The RAID level
- The number of drives in a volume group
- The number of volume groups
- The number of volumes in each volume group
- The number of hot spares
- The size of each segment on the drives
- A read ahead multiplier

After you define these parameters, the ISSM-EE software creates the volume groups, the volumes, and the hot spares. The controllers assign volume group numbers and volume numbers as they are created. After the ISSM-EE software creates the initial configuration, you can use the `set volume` command to define volume labels.

Before you run the `autoConfigure storageArray` command, run the `show storageArray autoConfigure` command. The latter command returns a list of parameter values that the ISSM-EE software uses to automatically create a storage array. If you would like to change any of the parameter values, you can do so by entering new values for the parameters when you run the `autoConfigure storageArray` command. If you are satisfied with the parameter values that the `show storageArray autoConfiguration` command returns, run the `autoConfigure storageArray` command without new parameter values.

The `autoConfigure storageArray` command takes this form:

```
autoConfigure storageArray
[driveType=(fibre | SATA | SAS | PATA)
raidLevel=(0 | 1 | 3 | 5 | 6) |
volumeGroupWidth=numberOfDrives |
volumeGroupCount=numberOfVolumeGroups |
volumesPerGroupCount=numberOfVolumesPerGroup |
hotSpareCount=numberOfHotSpares |
segmentSize=segmentSizeValue |
cacheReadPrefetch=(TRUE | FALSE)]
```

When you use the `autoConfigure storageArray` command, two symbol functions (`getAutoConfigCandidates` and `createAutoConfig`) are used that let the client retrieve default settings for the various automatic configuration parameters, change the settings, query what the results of those changes would be and, finally, apply the desired parameters to create a configuration. The configurability portion of this feature provides enhancements to the automatic volume group creation algorithms, which produce volume groups with improved performance and more information about drive and volume attributes so the user can make better choices when configuring volumes manually.

The `volumeGroupWidth` parameter defines the number of unassigned drives that you want to use for each new volume group.

The `volumeGroupCount` parameter defines the number of new volume groups that you want in the storage array.

The `volumesPerGroupCount` parameter defines the number of volumes that you want in each volume group.

The `hotSpareCount` parameter defines the number of hot spares that you want in each volume group.

The `segmentSize` parameter defines the amount of data, in KB, that the controller writes on a single drive in a volume before writing data on the next drive. The smallest units of storage are data blocks. A data block stores 512 bytes of data. The size of a segment determines how many data blocks that it contains. An 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

IMPORTANT For optimal performance in a multiuser database or file system storage environment, set the segment size to minimize the number of drives that are needed to satisfy an I/O request.

Using a single drive for a single request leaves other drives available to simultaneously service other requests. Valid segment size values are **8, 16, 32, 64, 128, 256, and 512**.

NOTE If you set the cache block size to **16**, you cannot create a volume with a segment size of **8**.

If the volume is for a single user with large I/O requests (such as multimedia), performance is maximized when a single I/O request can be serviced with a single data stripe. A data stripe is the segment size multiplied by the number of drives in the volume group that are used for data storage. In this environment, multiple drives are used for the same request, but each drive is accessed only once.

The `cacheReadPrefetch` parameter turns on or turns off the ability of the controller to read additional data blocks into the cache. When you turn on cache read prefetch, the controller copies additional data blocks into the cache while it is reading requested data blocks from a drive into the cache. This action increases the chance that a future request for data can be fulfilled from the cache, which improves the speed with which data is accessed. The number of additional data blocks that the controller reads into the cache is determined by the configuration settings for the storage array that you use. Cache read prefetch is important for applications that use sequential I/O, such as multimedia applications.

Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**. If you want to turn on cache read prefetch, set the `cacheReadPrefetch` parameter to **TRUE**. If you want to turn off cache read prefetch, set the `cacheReadPrefetch` parameter to **FALSE**.

Table 14 lists the default values for the segment size and cache read prefetch settings for different storage array uses.

Table 14 Default Values for Segment Size and Cache Read Prefetch

Storage Array Use	Segment Size (KB)	Cache Read Prefetch
File system	128	TRUE
Database	128	TRUE
Multimedia	256	TRUE

After you have finished creating the volume groups and the volumes by using the `autoConfigure storageArray` command, you can further define the properties of the volumes in a configuration by using the `set volume` command. (For a description about how to further define your configuration, see “[Chapter 3, Modifying Your Configuration](#)” on page 58.)

Example of the Auto Configuration Command

```
c:\...\smee\client>smcli 123.45.67.89 -c "autoConfigure
storageArray driveType=fibre raidLevel=5 volumeGroupWidth=8
volumeGroupCount=3 volumesPerGroupCount=4 hotSpareCount=2
segmentSize=8 cacheReadPrefetch=TRUE;"
```

The command in this example creates a storage array configuration by using Fibre Channel drives set to RAID level 5. Three volume groups are created, and each volume group consists of eight drives, which are configured into four volumes. The storage array has two hot spares. The segment size for each volume is 8 KB. The cache read prefetch is turned on, which causes additional data blocks to be written into the cache.

Using the Create Volume Command

Use the `create volume` command to create new storage array volumes in three ways:

- Create a new volume while simultaneously creating a new volume group to which you assign the drives
- Create a new volume while simultaneously creating a new volume group to which the ISSM-EE software assigns the drives
- Create a new volume in an existing volume group

You must have unassigned drives in the volume group. You do not need to assign the entire capacity of the volume group to a volume.

Creating Volumes with User-Assigned Drives

When you create a new volume and assign the drives you want to use, the ISSM-EE software creates a new volume group. The controller firmware assigns a volume group number to the new volume group. The command takes this form:

```
create volume drives=(trayID1,slotID1...trayIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=volumeName
[capacity=volumeCapacity] [owner=(a | b)]
```

```
cacheReadPrefetch=(TRUE | FALSE)  
segmentSize=segmentSizeValue]  
trayLossProtect=(TRUE | FALSE)]
```

NOTE The `capacity` parameter, the `owner` parameter, the `cacheReadPrefetch` parameter, the `segmentSize` parameter, and the `trayLossProtect` parameter are optional parameters (indicated by the items inside the square brackets). You can use one or all of the optional parameters as needed to define your configuration. If you choose not to use any of the optional parameters, the default values of the parameters are used for your configuration.

The `userLabel` parameter is the name that you want to give to the volume. The volume name can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the volume name is 30 characters. You must enclose the volume name with double quotation marks (“ ”).

The `drives` parameter is a list of the drives that you want to use for the volume group. Enter the tray ID and the slot ID of each drive that you want to use. Enclose the list in parentheses, separate the tray ID value and the slot ID value of a drive with a comma, and separate each tray ID and slot ID pair with a space. This example shows you how to enter tray ID values and slot ID values:

```
(1,1 1,2 1,3 1,4 1,5)
```

The `capacity` parameter defines the size of the volume. You do not need to assign the entire capacity of the drives to the volume. Later, you can assign any unused space to another volume.

The `owner` parameter defines the controller to which you want to assign the volume. If you do not specify a controller, the controller firmware determines the volume owner.

The `cacheReadPrefetch` parameter and the `segmentSize` parameter are the same as those described for the `autoConfigure storageArray` command.

The `trayLossProtect` parameter defines tray loss protection for the volume group. (For a description of how tray loss protection works, see “[Chapter 3, Tray Loss Protection](#)” on page 57.)

Example of Creating Volumes with User-Assigned Drives c:\...\smee\client>smcli
 123.45.67.89 -c "create volume
 drives=(1,1 1,2 1,3 2,1 2,2 2,3) raidLevel=5
 userLabel=\"Engineering_1\" capacity=20 GB
 owner=a cacheReadPrefetch=TRUE segmentSize=128;"

The command in this example automatically creates a new volume group and a volume with the name `Engineering_1`. The volume group has a RAID level of 5 (RAID 5). The command uses six drives to construct the volume group. The drives have a total volume capacity of 20 GB. If each drive has a capacity of 18 GB, the total capacity of all the assigned disks is 108 GB.

18 GB x 6 drives = 108 GB

Because only 20 GB is assigned to the volume, 88 GB remains available (as unconfigured capacity) for other volumes that a user can add to this volume group later.

108 GB - 20 GB volume group size = 88 GB

Cache read prefetch is turned on, which causes additional data blocks to be written into the cache. The segment size for each volume is 128 KB. Tray loss protection is set to **TRUE**, which prevents any operations to drives in the drive tray if the drive tray fails. Hot spares are not created for this new volume group. You must create hot spares after you run this command.

Creating Volumes with Software-Assigned Drives

If you choose to let the ISSM-EE software assign the drives when you create the volume, you need only to specify the number of drives that you want to use. The ISSM-EE software then assigns the drives. The controller firmware assigns a volume group number to the new volume group. To manually create volume groups and volumes, use the `create volume` command:

```
create volume driveCount=numberOfDrives
raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=volumeName
[driveType=(fibre | SATA | SAS | PATA)]
[capacity=volumeCapacity | owner=(a | b) |
cacheReadPrefetch=(TRUE | FALSE) |
segmentSize=segmentSizeValue]
[trayLossProtect=(TRUE | FALSE)]
```

This command is similar to the previous `create volume` command in which users assign the drives. The difference between this command and the previous one is that this version of the command requires only the number and the type of drives you want to use in the volume group. You do not need to enter a list of drives. All of the other parameters are the same. Tray loss

protection is performed differently when the ISSM-EE software assigns the drives than when a user assigns the drives. (For a description of the difference, see “[Chapter 3, Tray Loss Protection](#)” on page 57.)

Example of Creating Volumes with Software Assigned Drives

```
c:\...\smee\client>smcli 123.45.67.89 -c "create volume
driveCount=6 raidLevel=5 userLabel=\"Engineering_1\"
capacity=20 GB owner=a cacheReadPrefetch=TRUE segmentSize=128;"
```

The command in this example creates the same volume as the example for the previous `create volume` command in which a user assigns the drives. The difference is that a user does not know which drives are assigned to this volume group.

Creating Volumes in an Existing Volume Group

If you want to add a new volume to an existing volume group, use this command:

```
create volume volumeGroup=volumeGroupName
userLabel=volumeName
[freeCapacityArea=freeCapacityIndexNumber |
capacity=volumeCapacity | owner=(a | b) |
cacheReadPrefetch=(TRUE | FALSE) |
segmentSize=segmentSizeValue]
```

NOTE Parameters wrapped in square brackets or curly brackets are optional. You can use one or all of the optional parameters as needed to define your configuration. If you choose not to use any of the optional parameters, the default values of the parameter are provided for your configuration.

The `volumeGroup` parameter is the number of the volume group in which you want to create a new volume. If you do not know the volume group numbers on the storage array, you can use the `show allVolumes summary` command to get a list of the volumes and the volume groups to which the volumes belong.

The `userLabel` parameter is the name that you want to give to the volume. The volume name can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the volume name is 30 characters. You must enclose the volume name with double quotation marks (“ ”).

The `freeCapacityArea` parameter defines the free capacity area to use for the volume. If a volume group has several free capacity areas, you can use this parameter to identify which free capacity area to use for volume creation. You do not have to assign the entire capacity of the drives to the volume. Later, you can assign any unused space to another volume.

The usage of the `capacity` parameter, the `owner` parameter, the `cacheReadPrefetch` parameter, and the `segmentSize` parameter is the same as described in the previous examples of the `create volume` command.

Tray Loss Protection

The `trayLossProtect` parameter is a boolean switch that you set to turn on or turn off tray loss protection. For tray loss protection to work, each drive in a volume group must be on a separate tray. The way in which tray loss protection works depends on the method that you choose to assign the drives for a volume group.

When you assign the drives, if you set `trayLossProtect=TRUE` and have selected more than one drive from any one tray, the storage array returns an error. If you set `trayLossProtect=FALSE`, the storage array performs operations, but the volume group that you create does not have tray loss protection.

When the controller firmware assigns the drives, if `trayLossProtect=TRUE` the storage array posts an error if the controller firmware cannot provide drives that result in the new volume group having tray loss protection. If `trayLossProtect=FALSE`, the storage array performs the operation even if it means that the volume group might not have tray loss protection.

Tray loss protection is not valid when creating volumes on existing volume groups.

Modifying Your Configuration

For most configurations, after you have created your initial configuration by using the `autoConfigure storageArray` command or the `create volume` command, you must modify the properties of your configuration to make sure that it performs to meet the requirements for data storage. Use the `set` commands to modify a storage array configuration. This section describes how to modify these properties:

- The controller clocks
- The storage array password
- The storage array host type
- The storage array cache
- The global hot spares

Setting the Controller Clocks

To synchronize the clocks on the controllers with the host, use the `set storageArray time` command. Run this command to make sure that event time stamps that are written by the controllers to the Event Log match the event time stamps that are written to the host log files. The controllers stay available during synchronization. This example shows the command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set storageArray time;"
```

Setting the Storage Array Password

Use the `set storageArray` command to define a password for a storage array. The command takes this form:

```
set storageArray password="password"
```

The `password` parameter defines a password for the storage array. Passwords provide added security to a storage array to help reduce the possibility of implementing destructive commands.

ATTENTION Possible data corruption or data loss – Implementing destructive commands can cause serious damage, including data loss.

Unless you define a password for the storage array, you can run all of the script commands. A password protects the storage array from any command that the controllers consider destructive. A destructive command is any command that can change the state of the storage array, such as volume creation; cache modification; or reset, delete, rename, or change commands.

If you have more than one storage array in a storage configuration, each storage array has a separate password. Passwords can have a maximum length of 30 alphanumeric characters. You must enclose the password in double quotation marks (" "). This example shows how to use the `set storageArray` command to define a password:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set storageArray  
password=\"1a2b3c4d5e\";"
```

Setting the Storage Array Host Type

Use the `set storageArray` command to define the default host type. The command takes this form:

```
set storageArray defaultHostType=(hostTypeName | hostTypeIdentifier)
```

The `defaultHostType` parameter defines how the controllers in the storage array will communicate with the operating system on undefined hosts that are connected to the storage array SAN. This parameter defines the host type only for data I/O activities of the storage array. This parameter does not define the host type for the management station. The operating system can be Windows or Linux.

For example, if you set the `defaultHostType` parameter to Linux, the controller communicates with any undefined host if the undefined host is running a Linux operating system. Typically, you would need to change the host type only when you are setting up the storage array. The only time that you might need to use this parameter is if you need to change how the storage array behaves relative to the hosts that are connected to it.

Before you can define the default host type, you need to determine what host types are connected to the storage array. To return information about host types that are connected to the storage array, use the `show storageArray` command with the `defaultHostType` parameter or the `hostTypeTable` parameter. This command returns a list of the host types with which the controllers can communicate. This command does not return a list of the hosts. These examples show the use of the `show storageArray` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storageArray  
defaultHostType;"
```

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storageArray hostTypeTable;"
```

This example shows how to define a specific default host type:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set storageArray  
defaultHostType=11;"
```

The value **11** is the host type index value from the host type table that appears after entering the previous command.

Setting the Storage Array Cache

The cache is high-speed memory that holds data that is either written to the drives or read by the host. A controller has two memory areas used for intermediate storage of read data and write data. The read cache contains data that has been read from the drives but not yet transferred to the host. The write cache contains data from the host but not yet written to the drives.

The cache acts as a buffer so that data transfers between the host and the drive do not need to be synchronized. In read caching, the data for a read operation from the host might already be in the cache from a previous operation, which eliminates the need to access the drives. The data stays in the read cache until it is flushed. For write caching, a write operation stores data from the host in cache until it can be written to the drives.

The script command set provides two commands to define cache properties:

- `set storageArray`
- `set volume`

Use the `set storageArray` command to change the cache block size, the cache flush start value, and the cache stop value. The command takes this form:

```
set storageArray cacheBlockSize=cacheBlockSizeValue |  
cacheFlushStart=cacheFlushStartSize |  
cacheFlushStop=cacheFlushStopSize
```

You can enter one, two, or all three of the parameters on the command line.

The cache block size value defines the size of the data block that is used by the controller in transferring data into or out of the cache. You can set the cache block size to either **4KB**, **8KB**, or **16KB**. The value that you use applies to the entire storage array and all of the volumes in the storage array. For redundant controller configurations, this value includes all volumes owned by both controllers. Use smaller cache block sizes for systems that require transaction processing requests or I/O streams that are typically small and random. Use larger cache block sizes for large I/O, sequential, high-bandwidth applications. The choice of block size affects read/write performance. Large data transfers take longer in 4-KB block sizes than 16-KB block sizes. This example shows how to set the `cacheBlockSize` parameter:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set storageArray  
cacheBlockSize=16;"
```

To prevent data loss or corruption, the controller periodically writes cache data to the drives (flushes the cache) when the amount of unwritten data in the cache reaches a predefined level, called a start percentage. The controller also writes cache data to the drives when data has been in the cache for a predetermined amount of time. The controller writes data to the drives until the amount of data in the cache drops to a stop percentage level. Use the `set storageArray` command to set the start value and the stop value as percentages of the filled capacity of the cache. For example, you can specify that the controller start flushing the cache when it reaches 80-percent full and stop flushing the cache when it reaches 16-percent full. This example shows how to set these parameters:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set storageArray
cacheFlushStart=80 cacheFlushStop=16;"
```

Low start percentages and low stop percentages provide for maximum data protection. For both low start percentages and low stop percentages, the chance that data requested by a read command is not in the cache is increased. When the data is not in the cache, the cache hit percentage for writes and I/O requests decreases. Low start values and low stop values also increase the number of writes that are necessary to maintain the cache level. Increasing the number of writes increases the system overhead and further decreases performance.

Use the `set volume` command to change settings for the cache flush modifier, cache without batteries enabled, mirror cache enabled, the read ahead multiplier, read cache enabled, and write cache enabled. Use this command to set properties for all of the volumes or for a specific volume in a volume group. The command takes this form:

```
set (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN]
volume <wwID>) |
cacheFlushModifier=cacheFlushModifierValue |
cacheWithoutBatteryEnabled=(TRUE | FALSE) |
mirrorCacheEnabled=(TRUE | FALSE) |
readCacheEnabled=(TRUE | FALSE) |
writeCacheEnabled=(TRUE | FALSE) |
cacheReadPrefetch=(TRUE | FALSE)
```

The `cacheFlushModifier` parameter defines the amount of time that data stays in the cache before it is written to the drives. [Table 15](#) lists the values for the `cacheFlushModifier` parameter.

Table 15 Values for the `cacheFlushModifier` Parameter

Value	Meaning
Immediate	Data is flushed as soon as it is placed into the cache.
250	Data is flushed after 250 ms.
500	Data is flushed after 500 ms.
750	Data is flushed after 750 ms.
1	Data is flushed after 1 s.
1500	Data is flushed after 1500 ms.
2	Data is flushed after 2 s.
5	Data is flushed after 5 s.
10	Data is flushed after 10 s.
20	Data is flushed after 20 s.
60	Data is flushed after 60 s (1 min.).
120	Data is flushed after 120 s (2 min.).
300	Data is flushed after 300 s (5 min.).
1200	Data is flushed after 1200 s (20 min.).
3600	Data is flushed after 3600 s (1 hr.).
Infinite	Data in cache is not subject to any age or time constraints. The data is flushed based on other criteria managed by the controller.

This example shows how to set this parameter value for all of the volumes in the storage array:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set allvolumes
cacheFlushModifier=10;"
```

IMPORTANT Do not set the value of the `cacheFlushModifier` parameter above 10 seconds. An exception is for testing purposes. After running any tests in which you have set the values of the `cacheFlushModifier` parameter above 10 seconds, return the value of the `cacheFlushModifier` parameter to 10 or less seconds.

The `cacheWithoutBatteryEnabled` parameter turns on or turns off the ability of a host to perform write caching without backup batteries in a controller. To enable write caching without batteries, set this parameter to **TRUE**. To disable write caching without batteries, set this parameter to **FALSE**. If you set this parameter to **TRUE**, write caching continues, even when the controller batteries are completely discharged, not fully charged, or not present. If you do not have an uninterruptible power supply (UPS) and you enable this parameter, you can lose data if power to the storage array fails. This example shows how to set this parameter value:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume  
[\"Engineering\"] cacheWithoutBatteryEnabled=FALSE;"
```

The `mirrorCacheEnabled` parameter turns on or turns off write caching with mirroring. Write caching with mirroring permits cached data to be mirrored across redundant controllers that have the same cache size. Data written to the cache memory of one controller is also written to the cache memory of the second controller. If one controller fails, the second controller can complete all outstanding write operations. To use this option, these conditions must exist:

- The controller pair must be an active/active pair.
- The controllers must have the same size cache.

To enable write caching with mirroring, set this parameter to **TRUE**. To disable write caching with mirroring, set this parameter to **FALSE**. This example shows how to set this parameter:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume  
[\"Accounting\"] mirrorCacheEnabled=TRUE;"
```

The `readCacheEnabled` parameter turns on or turns off the ability of the host to read data from the cache. Read caching enables read operations from the host to be stored in controller cache memory. If a host requests data that is not in the cache, the controller reads the needed data blocks from the drives and places them in the cache. Until the cache is flushed, all of the other requests for this data are fulfilled with cache data rather than from a read, which increases throughput. To enable read caching, set this parameter to **TRUE**. To disable read caching, set this parameter to **FALSE**. This example shows how to set this parameter:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume  
[\"Balance_04\"] readCacheEnabled=TRUE;"
```

The `writeCacheEnabled` parameter turns on or turns off the ability of the host to write data to the cache. Write caching enables write operations from the host to be stored in cache memory. The volume data in the cache is automatically written to the drives every 10 seconds. To enable write caching, set this parameter to **TRUE**. To disable write caching, set this parameter to **FALSE**. This example shows how to set this parameter:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set allVolumes  
writeCacheEnabled=TRUE;"
```

The `cacheReadPrefetch` parameter turns on or turns off the ability of the controller to read additional data blocks into cache. When you turn on cache read prefetch, the controller copies additional data blocks into cache while it is reading requested data blocks from a drive into cache. This action increases the chance that a future request for data can be fulfilled from the cache, which improves the speed with which data is accessed. The number of additional data blocks that the controller reads into cache is determined by the storage array configuration settings that you use. Cache read prefetch is important for applications that use sequential I/O, such as multimedia applications.

Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**. If you want to turn on cache read prefetch, set the `cacheReadPrefetch` parameter to **TRUE**. If you want to turn off cache read prefetch, set the `cacheReadPrefetch` parameter to **FALSE**. This example shows how to set this parameter:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume  
[\"Engineering_1\" \"Engineering_2\"] cacheReadPrefetch=TRUE;"
```

Setting the Modification Priority

Modification priority defines how much processing time is allocated for volume modification operations. Time allocated for volume modification operations affects system performance. Increases in volume modification priority can reduce read/write performance. The modification priority affects these operations:

- Copyback
- Reconstruction
- Initialization
- Changing the segment size
- Defragmentation of a volume group
- Adding free capacity to a volume group
- Changing the RAID level of a volume group

The lowest priority rate favors system performance, but the modification operation takes longer. The highest priority rate favors the modification operation, but the system performance might be degraded.

Use the `set volume` command to define the modification priority for a volume. The command takes this form:

```
set (allVolumes | volume [volumeName] |  
volumes [volumeName1 ... volumeNameN] volume <wwID> |  
accessVolume)  
modificationPriority=(highest | high | medium | low | lowest)
```

This example shows how to use this command to set the modification priority for volumes named `Engineering_1` and `Engineering_2`:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume [\"Engineering_1\"  
\"Engineering_2\" modificationPriority=lowest;"
```

The modification rate is set to **lowest** so that system performance is not significantly reduced by modification operations.

Assigning Global Hot Spares

You can assign or unassign global hot spares by using the `set drive` command. To use this command, you must identify the location of the drives by the tray ID and the slot ID. Then, you set the `hotSpare` parameter to **TRUE** to enable the hot spare or **FALSE** to disable an existing hot spare. The command takes this form:

```
set (drive [trayID,slotID] | drives [trayID1,slotID1 ... trayIDn,slotIDn])
hotSpare=(TRUE | FALSE)
```

This example shows how to set hot spare drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set drives [1,2 1,3]
hotSpare=TRUE;"
```

Enter the tray ID and the slot ID of each drive that you want to use. Enclose the list in square brackets, separate the tray ID value and the slot ID value of a drive with a comma, and separate each tray ID and slot ID pair with a space.

Saving a Configuration to a File

After you have created a new configuration or if you want to copy an existing configuration for use on other storage arrays, you can save the configuration to a file by using the `save storageArray configuration` command. Saving the configuration creates a script file that you can run on the command line. The command takes this form:

```
save storageArray configuration file="filename"
[(allconfig | globalSettings=(TRUE | FALSE)) |
volumeConfigAndSettings=(TRUE | FALSE) |
hostTopology=(TRUE | FALSE) | lunMappings=(TRUE | FALSE)]
```

ATTENTION **Possible loss of data** – When information is written to a file, the script engine does not check to determine if the file name already exists. If you choose the name of a file that already exists, the script engine writes over the information in the file without warning.

You can choose to save the entire configuration or specific configuration features. This example shows how to set this parameter value:

```
c:\...\smee\client>smcli 123.45.67.89 -c "save storagearray configuration  
file=\"c:\\folder\\storagearrayconfig1.scr\";"
```

In this example, the name `folder` is the folder in which you choose to place the profile file and `storagearrayconfig1.scr` is the name of the file. You can choose any folder and any file name. The file extension for a configuration file is `.scr`. The ISSM-EE software uses this extension when it creates the configuration file.

Using the Snapshot Premium Feature

The Snapshot premium feature creates a snapshot volume that you can use as a backup of your data. A snapshot volume is a logical point-in-time image of a standard volume. Because it is not a physical copy, a snapshot volume is created more quickly than a physical copy and requires less storage space on the drive. Typically, you create a snapshot volume so that an application, such as a backup application, can access the snapshot volume and read the data while the base volume stays online and user accessible. You can also create several snapshot volumes of a base volume and write data to the snapshot volumes to perform testing and analysis.

Snapshot volumes provide these capabilities:

- Create a complete image of the data on a base volume at a particular point in time
- Use only a small amount of storage space
- Provide for quick, frequent, non-disruptive backups, or testing new versions of a database system without affecting real data
- Provide for snapshot volumes to be read, written, and copied
- Use the same availability characteristics of the base volume (such as RAID protection and redundant path failover)
- Mapping the snapshot volume and making it accessible to any host on a storage area network (SAN). You can make snapshot data available to secondary hosts for read access and write access by mapping the snapshot to the hosts
- Creating up to four snapshots per volume (the maximum number of snapshot volumes is one-half of the total number of volumes that are supported by the controller)
- Increasing the capacity of a snapshot volume

How Snapshot Works

Three components comprise a snapshot volume: the base volume, the snapshot volume, and the snapshot repository volume. [Table 16](#) lists the components and briefly describes what they do.

Table 16 Components of a Snapshot Volume

Component	Description
Base volume	A standard volume from which the snapshot is created
Snapshot volume	A logical point-in-time image of a standard volume
Snapshot repository volume	A volume that contains snapshot metadata and copy-on-write data for a particular snapshot volume

Based on information that you provide through the script commands, the storage management software creates an empty snapshot repository volume and defines the mapping from a base volume to the snapshot repository volume. The snapshot repository volume holds changed data that a host writes to the base volume. When the snapshot repository volume is first created, it holds only the metadata about the snapshot volume with which it is associated.

NOTE When you first create a snapshot repository volume, briefly stop all of the write operations to the base volume so that a stable image of the base volume is available.

When the host writes to the base volume, the new data is also copied to the snapshot repository volume. This action is called *copy-on-write*. A snapshot is constructed by combining the updated data in the snapshot repository volume with data in the base volume that has not been altered. This action creates a complete copy of the base volume at a specific point in time. The snapshot appears as a volume that contains the original data at the time of creation, but the snapshot is actually an image that is the combination of the snapshot repository volume and the original base volume. The snapshot repository volume, which houses original data that has been changed, is the only additional drive space that is needed for the snapshot volume. The additional drive space is typically 10 percent to 20 percent of the drive space of the base volume and varies depending on the amount of changes to the data. The longer a snapshot volume is active, the larger the snapshot repository volume must be. The default size of the snapshot repository volume is 20 percent of the base volume; however, you can set the size of the snapshot repository volume to other values.

You can read, write, and copy a snapshot volume. Data written by a host to the snapshot volume is handled in the snapshot repository volume. When a write occurs to the base volume of a snapshot volume, the new data also overwrites the appropriate snapshot repository volume data.

Table 17 Snapshot Volume Commands

Command	Description
<code>create snapshotVolume</code>	This command creates a snapshot volume.
<code>recreate snapshot</code>	This command starts a fresh copy-on-write operation by using an existing snapshot volume.
<code>recreate snapshot collection</code>	This command restarts multiple snapshot volumes as one batch operation using one or many existing snapshot volumes.
<code>set (snapshotVolume)</code>	This command defines the properties for a snapshot volume and lets you rename a snapshot volume.
<code>stop snapshot</code>	This command stops a copy-on-write operation.

Creating a Snapshot Volume

The `create snapshotVolume` command provides three methods for defining the drives for your snapshot repository volume:

- Defining the drives for the snapshot repository volume by their tray IDs and their slot IDs
- Defining a volume group in which the snapshot repository volume resides. In addition, you can define the capacity of the snapshot repository volume.
- Defining the number of drives, but not specific drives, for the snapshot repository volume.

When you use the `create snapshotVolume` command to create a snapshot volume, the minimum information that you need to provide is the standard volume that you want to use for the base volume. When you create a snapshot volume by using minimum information, the storage management software provides default values for the other property parameters that are required for a completely defined snapshot volume.

Creating a Snapshot Volume with User-Assigned Drives

Creating a snapshot volume by assigning the drives provides flexibility in defining your configuration by letting you choose from the available drives in your storage array. When you choose the drives for your snapshot volume, you automatically create a new volume group. You can specify which drives to use and the RAID level for the new volume group. The command takes this form:

```
create snapshotVolume baseVolume="baseVolumeName"
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
(repositoryDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failSnapShot)]
[trayLossProtect=(TRUE | FALSE)]
```

This example shows a command in which users assign the drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create snapshotVolume
baseVolume=\"Mars_Spirit_4\" repositoryRAIDLevel=5 repositoryDrives=(1,1
1,2 1,3 1,4 1,5);"
```

The command in this example creates a new snapshot of the base volume `Mars_Spirit_4`. The snapshot repository volume consists of five drives that form a new volume group. The new volume group has a RAID level of 5. This command also takes a snapshot of the base volume, which starts the copy-on-write operation.

This example shows how to use the command in a script file:

```
create snapshotVolume baseVolume="Mars_Spirit_4" repositoryRAIDLevel=5
repositoryDrives=(1,1 1,2 1,3 1,4 1,5);
```

This example shows a minimal version of the command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create snapshotVolume
baseVolume=\"Mars_Spirit_4\";"
```

The command in this example creates a new snapshot for the base volume `Mars_Spirit_4`. The snapshot repository volume is created in the same volume group as the base volume, which means that the snapshot repository volume has the same RAID level as the base volume. This command starts the copy-on-write operation.

This example shows how to use the command in a script file:

```
create snapshotVolume baseVolume="Mars_Spirit_4";
```

Creating a Snapshot Volume with Software-Assigned Drives

With this version of the `create snapshotVolume` command, you choose an existing volume group in which to place the snapshot repository volume. The storage management software determines which drives to use. You can also define how much space to assign to the snapshot repository volume. Because you are using an existing volume group, the RAID level for the snapshot volume defaults to the RAID level of the volume group in which you place it. You cannot define the RAID level for the snapshot volume. The command takes this form:

```
create snapshotVolume baseVolume="baseVolumeName"
[repositoryVolumeGroup=volumeGroupName
freeCapacityArea=freeCapacityIndexNumber
userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel=repositoryName
repositoryFullPolicy=(failBaseWrites | failSnapShot)]
[trayLossProtect=(TRUE | FALSE)]
```

This example shows a command in which the storage management software assigns the drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create
snapshotVolume baseVolume=\"Mars_Spirit_4\"
repositoryVolumeGroup=2 freeCapacityArea=2;"
```

The command in this example creates a new snapshot repository volume in volume group 2. The base volume is `Mars_Spirit_4`. The size of the snapshot repository volume is 4 GB. This command also takes a snapshot of the base volume, starting the copy-on-write operation.

When you define the capacity of a snapshot repository volume, specify a size that is 20 percent of the size of the base volume. In the previous example, the size of the snapshot repository volume is set to 4 GB. The underlying assumption is that the base volume size is 20 GB (0.2 x 20 GB= 4 GB).

This example shows how to use the command in a script file:

```
create snapshotVolume baseVolume="Mars_Spirit_4"
repositoryVolumeGroup=2 freeCapacityArea=2;
```

Creating a Snapshot Volume by Specifying a Number of Drives

With this version of the `create snapshotVolume` command, you must specify the number of drives and the RAID level that you want for the snapshot repository volume. This version of the `create snapshotVolume` command creates a new volume group. You must have drives in the storage array that are not assigned to a volume group for this command to work.

```
create snapshotVolume baseVolume="baseVolumeName"
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
driveType=(fibre | SATA | SAS | PATA)
userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failSnapShot)]
[trayLossProtect=(TRUE | FALSE)]
```

This example shows how to use a command in which users specify the number of drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create snapshotVolume
baseVolume=\"Mars_Spirit_4\" repositoryRAIDLevel=5 repositoryDriveCount=3;"
```

The command in this example creates a new snapshot repository volume that consists of three drives. Three drives comprise a new volume group that has a RAID level of 5. This command also takes a snapshot of the base volume, which starts the copy-on-write operation.

This example shows how to use the command in a script file:

```
create snapshotVolume baseVolume= "Mars_Spirit_4"
repositoryRAIDLevel=5 repositoryDriveCount=3;
```

User-Defined Parameters

Use the parameters in the `create snapshotVolume` command to define the snapshot volume to suit the requirements of your storage array. [Table 18](#) lists the parameters and briefly describes what the parameters do.

Table 18 Snapshot Volume Parameters

Parameter	Description
driveType	The type of drive that you want to use for the snapshot repository volume. The choice is fibre (Fibre Channel), SATA , SAS , or PATA . This parameter works only with the count-based repository method of defining a snapshot volume.
repositoryVolumeGroup	The volume group in which you want to build the snapshot repository volume. The default value is to build the snapshot repository volume in the same volume group as the base volume.
freeCapacityArea	The amount of storage space that you want to use for the snapshot repository volume. Free storage space is defined in units of bytes, KB, MB, GB, or TB.
userLabel	<p>The name that you want to give to the snapshot volume. If you do not choose a name for the snapshot volume, the software creates a default name by using the base volume name. For example, with a base volume name of Mars_Spirit_4:</p> <ul style="list-style-type: none"> • When the base volume does not have a snapshot volume, the default snapshot volume name is Mars_Spirit_4-1. • When the base volume already has $n-1$ number of snapshot volumes, the default name is Mars_Spirit_4-n.
repositoryUserLabel	The name that you want to give to the snapshot repository volume. If you do not choose a name for the snapshot repository volume, the software creates a default name by using the base volume name. For example, if the base volume name is Mars_Spirit_4 and does not have an associated snapshot repository volume, the default snapshot repository volume name is Mars_Spirit_4-R1. If the base volume already has $n-1$ number of snapshot repository volumes, the default name is Mars_Spirit_4-R n .

Parameter	Description
warningThresholdPercent	The percentage of the capacity that you will permit the snapshot repository volume to get before you receive a warning that the snapshot repository volume is nearing full. The warning value is a percentage of the total capacity of the snapshot repository volume. The default value is 50 , which represents 50 percent of the total capacity. (You can change this value later by using the <code>set snapshotVolume</code> command.)
repositoryPercentOfBase	The size of the snapshot repository volume as a percentage of the base volume size. The default value is 20 , which represents 20 percent of the base volume size.
repositoryFullPolicy	The type of snapshot processing that you want to continue if the snapshot repository volume is full. You can choose to fail writes to the base volume (failBaseWrites) or fail writes to the snapshot volume (failSnapShot). The default value is failSnapShot .

This example shows the `create snapshotVolume` command that includes user-defined parameters:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create
snapshotVolume baseVolume=\"Mars_Spirit_4\"
repositoryRAIDLevel=5 repositoryDriveCount=5
driveType=fibre userLabel=\"Mars_Spirit_4_snap1\"
repositoryUserLabel=\"Mars_Spirit_4_repl\"
warningThresholdPercent=75 repositoryPercentOfBase=40
repositoryFullPolicy=failSnapShot;"
```

This example shows how to use the command in a script file:

```
create snapshotVolume baseVolume="Mars_Spirit_4"
repositoryRAIDLevel=5 repositoryDriveCount=5 driveType=fibre
userLabel="Mars_Spirit_4_snap1"
repositoryUserLabel="Mars_Spirit_4_repl"
warningThresholdPercent=75 repositoryPercentOfBase=40
repositoryFullPolicy=failSnapShot;
```

Snapshot Volume Names and Snapshot Repository Volume Names

The snapshot volume names and the snapshot repository volume names can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the volume names is 30 characters. You must enclose the volume name in double quotation marks. The character string cannot contain a new line. Make sure that you use unique names; if you do not use unique names, the controller firmware returns an error.

One technique for naming the snapshot volume and the snapshot repository volume is to add a hyphenated suffix to the original base volume name. The suffix distinguishes between the snapshot volume and the snapshot repository volume. For example, if you have a base volume with a name of Engineering Data, the snapshot volume can have a name of Engineering Data-S1, and the snapshot repository volume can have a name of Engineering Data-R1.

If you do not choose a unique name for either the snapshot volume or the snapshot repository volume, the controllers create a default name by using the base volume name. These examples are snapshot volume names that the controllers might create:

- If the base volume name is *aaa* and does not have a snapshot volume, the default snapshot volume name is *aaa1*.
- If the base volume already has $n-1$ number of snapshot volumes, the default name is *aaa-n*.
- If the base volume name is *aaa* and does not have a snapshot repository volume, the default snapshot repository volume name is *aaa-R1*
- If the base volume already has $n-1$ number of snapshot repository volumes, the default name is *aaa-Rn*.

In the examples from the previous section, the user-defined snapshot volume name was `Mars_Spirit_4_snap1`, and the user-defined snapshot repository volume name was `Mars_Spirit_4_rep1`. The default name that was provided by the controller for the snapshot volume was `Mars_Spirit_4-1`. The default name that was provided by the controller for the snapshot repository volume was `Mars_Spirit_4-R1`.

Changing Snapshot Volume Settings

Use the `set (snapshot) volume` command to change these property settings for a snapshot volume:

- The snapshot volume name
- The warning threshold percent
- The snapshot repository full policy

This example shows how to change a snapshot volume name.

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume [\"Mars_Spirit_4-1\"]
userLabel= \"Mars_Odyssey_3-2\";"
```

This example shows how to use the command in a script file:

```
set volume ["Mars_Spirit_4-1"] userLabel="Mars_Odyssey_3-2";
```

When you change the warning threshold percent and the snapshot repository full policy, you can apply the changes to one or several snapshot volumes with this command. This example shows how to use the `set (snapshot) volume` command to change these properties on more than one snapshot volume:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volume
[\"Mars_Spirit_4-1\" \"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"
warningThresholdPercent=50
repositoryFullPolicy=failBaseWrites;"
```

This example shows how to use the command in a script file:

```
set volume ["Mars_Spirit_4-1" "Mars_Spirit_4-2"
"Mars_Spirit_4-3"] warningThresholdPercent=50
repositoryFullPolicy=failBaseWrites;
```

Stopping, Restarting, and Deleting a Snapshot Volume

When you create a snapshot volume, copy-on-write starts running immediately. As long as a snapshot volume is enabled, storage array performance is impacted by the copy-on-write operations to the associated snapshot repository volume.

If you no longer want copy-on-write operations to run, you can use the `stop snapshot volume` command to stop the copy-on-write operations. When you stop a snapshot volume, the snapshot volume and the snapshot repository volume are still defined for the base volume. Only copy-on-write has stopped. This example shows how to stop a snapshot volume:

```
c:\...\smee\client>smcli 123.45.67.89 -c "stop snapshot volumes  
[\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];"
```

This example shows how to use the command in a script file:

```
stop snapshot volumes [\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];
```

When you stop the copy-on-write operations for a specific snapshot volume, only that snapshot volume is disabled. All of the other snapshot volumes stay in operation.

When you want to restart a copy-on-write operation, use the `recreate snapshot volume` command or the `recreate snapshot collection` command. The `recreate snapshot volume` command starts a fresh copy-on-write operation by using an existing snapshot volume.

NOTE The snapshot volume must be in either an Optimal state or a Disabled state.

When you restart a snapshot volume, these actions occur:

- All copy-on-write data previously on the snapshot repository volume is overwritten.
- Snapshot volume parameters and snapshot repository volume parameters stay the same as the previously disabled snapshot volume and the previously disabled snapshot repository volume. You can also change the `userLabel` parameter, the `warningThresholdPercent` parameter, and the `repositoryFullPolicy` parameter when you restart the snapshot volume.
- The original names for the snapshot repository volume are retained.

This example shows how to restart a snapshot volume:

```
c:\...\smee\client>smcli 123.45.67.89 -c "recreate snapshot volumes  
[\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];"
```

This example shows how to use the command in a script file:

```
recreate snapshot volumes ["Mars_Spirit_4-2" "Mars_Spirit_4-3"];
```

If you do not intend to use a snapshot volume again, you can delete the snapshot volume by using the `delete volume` command. When you delete a snapshot volume, the associated snapshot repository volume also is deleted.

Using the Remote Volume Mirroring Premium Feature

The Remote Volume Mirroring premium feature provides for online, real-time replication of data between storage arrays over a remote distance. In the event of a disaster or a catastrophic failure on one storage array, you can promote the second storage array to take over responsibility for computing services. Remote Volume Mirroring is designed for extended storage environments in which the storage arrays that are used for Remote Volume Mirroring are maintained at separate sites. Volumes on one storage array are mirrored to volumes on another storage array across a fabric SAN. Data transfers can be synchronous or asynchronous. You choose the method when you set up the remote-mirror pair. The data transfers occur at Fibre Channel speeds to maintain data on the different storage arrays. Because Remote Volume Mirroring is storage based, it does not require any server overhead or application overhead.

You can use Remote Volume Mirroring for these functions:

- **Disaster recovery** – Remote Volume Mirroring lets you replicate data from one site to another site, which provides an exact mirror duplicate at the remote (secondary) site. If the primary site fails, you can use mirrored data at the remote site for failover and recovery. You can then shift storage operations to the remote site for continued operation of all of the services that are usually provided by the primary site.
- **Data vaulting and data availability** – Remote Volume Mirroring lets you send data off site where it can be protected. You can then use the off-site copy for testing or to act as a source for a full backup to avoid interrupting operations at the primary site.
- **Two-way data protection** – Remote Volume Mirroring provides the ability to have two storage arrays back up each other by mirroring critical volumes on each storage array to volumes on the other storage array. This action lets each storage array recover data from the other storage array in the event of any service interruptions.

How Remote Volume Mirroring Works

When you create a remote-mirror pair, the remote-mirror pair consists of a *primary volume* on a local storage array and a *secondary volume* on a storage array at another site. A standard volume might only be included in one mirrored volume pair.

Table 19 Maximum Number of Defined Mirrors

Controller Model	Maximum Number of Defined Mirrors
TP9300	32
IS350, IS4000	64
TP9400	64
TP9500	64
TP9700	128
IS4500	128
IS4600	128

The primary volume is the volume that accepts host I/O activity and stores application data. When the mirror relationship is first created, data from the primary volume is copied in its entirety to the secondary volume. This process is known as a *full synchronization* and is directed by the controller owner of the primary volume. During a full synchronization, the primary volume remains fully accessible for all normal I/O operations.

The controller owner of the primary volume initiates remote writes to the secondary volume to keep the data on the two volumes synchronized.

The secondary volume maintains a mirror (or copy) of the data on its associated primary volume. The controller owner of the secondary volume receives remote writes from the controller owner of the primary volume but will not accept host write requests. Hosts are able to read from the secondary volume, which appears as read-only.

In the event of a disaster or a catastrophic failure at the primary site, you can perform a role reversal to promote the secondary volume to a primary role. Hosts then are able to read from and write to the newly promoted volume, and business operations can continue.

Mirror Repository Volumes

A mirror repository volume is a special volume in the storage array that is created as a resource for the controller owner of the primary volume in a remote-mirror pair. The controller stores mirroring information on this volume, including information about remote writes that are not yet complete. The controller can use this information to recover from controller resets and the accidental powering down of the storage arrays.

When you activate the Remote Volume Mirroring premium feature on the storage array, you create two mirror repository volumes, one for each controller in the storage array. An individual mirror repository volume is not needed for each remote mirror.

When you create the mirror repository volumes, you specify the location of the volumes. You can either use existing free capacity, or you can create a volume group for the volumes from unconfigured capacity and then specify the RAID level.

Because of the critical nature of the data being stored, do not use RAID 0 as the RAID level of mirror repository volumes. The required size of each volume is 128 MB, or 256 MB total for both mirror repository volumes of a dual-controller storage array. In previous versions of the Remote Volume Mirroring feature, the mirror repository volumes required less disk storage space and needed to be upgraded to use the maximum amount of mirror relationships.

Mirror Relationships

Before you create a mirror relationship, you must enable the Remote Volume Mirroring premium feature on both the primary storage array and the secondary storage array. You must also create a secondary volume on the secondary site if one does not already exist. The secondary volume must be a standard volume of equal or greater capacity than the associated primary volume.

When secondary volumes are available, you can establish a mirror relationship in the storage management software by identifying the primary volume and the storage array that contains the secondary volume.

When you first create the mirror relationship, a full synchronization automatically occurs, with data from the primary volume copied in its entirety to the secondary volume.

Data Replication

The controllers manage data replication between the primary volume and the secondary volume. This process is transparent to host machines and applications. This section describes how data is replicated between the storage arrays that are participating in Remote Volume Mirroring. This section also describes the actions taken by the controller owner of the primary volume if a link interruption occurs between storage arrays.

Write Modes

When the controller owner of the primary volume receives a write request from a host, the controller first logs information about the write to a mirror repository volume, and then writes the data to the primary volume. The controller then initiates a remote write operation to copy the affected data blocks to the secondary volume at the secondary storage array.

The Remote Volume Mirroring feature provides two write mode options that affect when the I/O completion indication is sent back to the host: Synchronous and Asynchronous.

Synchronous Write Mode

Synchronous write mode provides the highest level security for full data recovery from the secondary storage array in the event of a disaster. Synchronous write mode does reduce host I/O performance. When this write mode is selected, host write requests are written to the primary volume and then copied to the secondary volume. After the host write request has been written to the primary volume and the data has been successfully copied to the secondary volume, the controller removes the log record on the mirror repository volume. The controller then sends an I/O completion indication back to the host system. Synchronous write mode is selected as the default value and is the recommended write mode.

Asynchronous Write Mode

Asynchronous write mode offers faster host I/O performance but does not guarantee that a copy operation has successfully completed before processing the next write request. When you use Asynchronous write mode, host write requests are written to the primary volume. The controller then sends an “I/O complete” indication back to the host system, without acknowledging that the data has been successfully copied to the secondary (remote) storage array.

When using Asynchronous write mode, write requests are not guaranteed to be completed in the same order on the secondary volume as they are on the primary volume. If the order of write requests is not retained, data on the secondary volume might become inconsistent with the data on the primary volume. This event could jeopardize any attempt to recover data if a disaster occurs on the primary storage array.

Write Consistency Mode

When multiple mirror relationships exist on a single storage array and have been configured to use Asynchronous write mode and to preserve consistent write order, they are considered to be an interdependent group that is in the *Write consistency mode*. The data on the secondary, remote storage array cannot be considered fully synchronized until all of the remote mirrors that are in the Write consistency mode are synchronized.

If one mirror relationship in the group becomes unsynchronized, all of the mirror relationships in the group become unsynchronized. Any write activity to the remote, secondary storage arrays is prevented to protect the consistency of the remote data set.

Link Interruptions or Secondary Volume Errors

When processing write requests, the primary controller might be able to write to the primary volume, but a link interruption might prevent communication with the remote (secondary) controller.

In this case, the remote write operation cannot be completed to the secondary volume, and the primary volume and the secondary volume are no longer correctly mirrored. The primary controller transitions the mirrored pair into an Unsynchronized state and sends an I/O completion to the primary host. The primary host can continue to write to the primary volume, but remote writes do not take place.

When communication is restored between the controller owner of the primary volume and the controller owner of the secondary volume, a resynchronization takes place. This resynchronization happens automatically, or it must be started manually, depending on which write mode you chose when setting up the mirror relationship. During the resynchronization, only the blocks of data that have changed on the primary volume during the link interruption are copied to the secondary volume. After the resynchronization starts, the mirrored pair transitions from an Unsynchronized status to a Synchronization in Progress status.

The primary controller also marks the mirrored pair as unsynchronized when a volume error on the secondary side prevents the remote write from completing. For example, an offline secondary volume or a failed secondary volume can cause the remote mirror to become unsynchronized. When the volume error is corrected (the secondary volume is placed online or recovered to an Optimal status), then synchronization is required. The mirrored pair then transitions to a Synchronization in Progress status.

Resynchronization

Data replication between the primary volume and the secondary volume in a mirror relationship is managed by the controllers and is transparent to host machines and applications. When the controller owner of the primary volume receives a write request from a host, the controller first logs information about the write to a mirror repository volume. The controller then writes the data to the primary volume. The controller then initiates a write operation to copy the affected data to the secondary volume on the remote storage array.

If a link interruption or a volume error prevents communication with the secondary storage array, the controller owner of the primary volume transitions the mirrored pair into an Unsynchronized status. The controller owner then sends an I/O completion to the host sending the write request. The host can continue to issue write requests to the primary volume, but remote writes to the secondary volume do not take place.

When connectivity is restored between the controller owner of the primary volume and the controller owner of the secondary volume, the volumes must be resynchronized by copying the blocks of data that changed during the interruption to the secondary volume. Only the blocks of data that have changed on the primary volume during the link interruption are copied to the secondary volume.

ATTENTION **Possible loss of data access** – Any communication disruptions between the primary storage array and the secondary storage array while resynchronization is underway could result in a mix of new data and old data on the secondary volume. This condition would render the data unusable in a disaster recovery situation.

Creating a Remote-Mirror Pair

Before you create any mirror relationships, volumes must exist at both the primary site and the secondary site. The volume that resides on the local storage array is the primary volume. Similarly, the volume that resides on the remote storage array is the secondary volume. If neither the primary volume or the secondary volume exist, you must create these volumes. When you create the secondary volume, you must consider these items:

- The secondary volume must be of equal or greater size than the primary volume.
- The RAID level of the secondary volume does not have to be the same as the primary volume.

Use these steps to create the volume.

- 1 Enable the Remote Volume Mirroring feature.
- 2 Activate the Remote Volume Mirroring feature.
- 3 Determine candidates for a remote-mirror pair.
- 4 Create the remote-mirror relationship.

Performance Considerations

Consider these performance issues when you create mirror relationships:

- The controller owner of a primary volume performs a full synchronization in the background while processing local I/O writes to the primary volume and associated remote writes to the secondary volume. Because the full synchronization diverts controller processing resources from I/O writes, full synchronization can have a performance impact to the host application.
- To reduce the performance impact, you can set the synchronization priority level to determine how the controller owner will prioritize the full synchronization relative to other I/O activity. To set the synchronization priority level, consider these guidelines:
 - A full synchronization at the lowest synchronization priority level takes approximately eight times as long as a full synchronization at the highest synchronization priority level.
 - A full synchronization at the low synchronization priority level takes approximately six times as long as a full synchronization at the highest synchronization priority level.

- A full synchronization at the medium synchronization priority level takes approximately three-and-a-half times as long as a full synchronization at the highest synchronization priority level.
- A full synchronization at the high synchronization priority level takes approximately twice as long as a full synchronization at the highest synchronization priority level.
- When the mirrored volume pair is in a Synchronization in Progress state, all host write data is copied to the remote system. Both controller I/O bandwidth and I/O latency can affect host write performance. Host read performance is not affected by the mirroring relationship.
- The time that it takes for data to be copied from the primary volume to the secondary volume might impact overall performance. This impact is primarily caused by the delay and system resource required for copying data to the remote mirror. Some delay might also occur because of the limit to the number of simultaneous writes.

Enabling the Remote Volume Mirroring Feature

The first step in creating a remote mirror is to make sure that the Remote Volume Mirroring feature is enabled on both storage arrays. Because Remote Volume Mirroring is a premium feature, you need a Feature Key file to enable the feature. The command for enabling the Feature Key file is as follows:

```
enable storageArray feature file="filename"
```

where the `file` parameter is the complete file path and file name of a valid Feature Key file. Enclose the file path and the file name in double quotation marks (" "). Valid file names for Feature Key files end with a `.key` extension.

Activating the Remote Volume Mirroring Feature

Activating the Remote Volume Mirroring feature prepares the storage arrays to create and configure mirror relationships. After you activate the feature, the secondary ports for each controller are reserved and dedicated to remote mirror use. In addition, a mirror repository volume is automatically created for each controller in the storage array. As part of the activation process, you can decide where the mirror repository volumes will reside, free capacity on an existing volume group or in a newly created volume group, and the RAID level for the mirror repository volumes.

The free capacity that you select for the mirror repository volume must have a total of 256 MB of capacity available. Two mirror repository volumes are created on this capacity, one for each controller. If you enter a value for the repository storage space that is too small for the mirror repository volumes, the firmware returns an error message that gives the amount of space needed for the mirror repository volumes. The command does not try to activate the Remote Volume Mirroring feature. You can re-enter the command using the value from the error message for the repository storage space value.

The RAID level that you choose for the mirror repository volume has these constraints:

- **RAID 0** – You cannot use RAID 0.
- **RAID 1** – The number of drives must be an even number. If you select an odd number of drives, the controller firmware returns an error.
- **RAID 3 or RAID 5** – You must have a minimum of three drives in the volume group.
- **RAID 6** – You must have a minimum of five drives in the volume group.

To activate the Remote Volume Mirroring feature, use this command:

```
activate storageArray feature=remoteMirror
```

The `activate storageArray feature=remoteMirror` command provides three methods for defining the drives for your mirror repository volume:

- You define each drive for the mirror repository volume by its tray ID and its slot ID.
- You define a volume group in which the mirror repository volume resides. You can optionally define the capacity of the mirror repository volume.
- You define the number of drives, but not specific drives, for the mirror repository volume.

Activating the Remote Volume Mirroring Feature with User-Assigned Drives

Activating the Remote Volume Mirroring feature by assigning the drives provides flexibility in defining your configuration by letting you choose from the available drives in your storage array. Choosing the drives for your remote mirror automatically creates a new volume group. You can specify which drives to use and the RAID level for the new volume group.

The command takes this form:

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
trayLossProtect=(TRUE | FALSE)
```

This example shows a command in which you assign the drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "activate
storageArray feature=remoteMirror repositoryRAIDLevel=5
repositoryDrives=(1,1 1,2 1,3 1,4 1,5);"
```

The command in this example creates a new mirror repository volume consisting of five drives that forms a new volume group. The new volume group has a RAID level of 5.

This example shows how to use the command in a script file:

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=5
repositoryDrives=(1,1 1,2 1,3 1,4 1,5);
```

Activating the Remote Volume Mirroring Feature with Software-Assigned Drives

With this version of the `activate storageArray feature=remoteMirror` command, you choose an existing volume group in which to place the mirror repository volume. The storage management software then determines which drives to use. You can also define how much space to assign to the mirror repository volume. Because you are using an existing volume group, the RAID level for the mirror repository volume defaults to the RAID level of the volume group in which you place it. You cannot define the RAID level for the mirror repository volume.

The command takes this form:

```
activate storageArray feature=remoteMirror
repositoryVolumeGroup=volumeGroupName
[freeCapacityArea=freeCapacityIndexNumber]
```

This example shows a command in which the software assigns the drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "activate storageArray
feature=remoteMirror repositoryVolumeGroup=2 freeCapacityArea=2;"
```

The command in this example creates a new mirror repository volume in volume group 2 using the second free capacity area.

This example shows how to use the command in a script file:

```
activate storageArray feature=remoteMirror repositoryVolumeGroup=2
freeCapacityArea=2;
```

Activating the Remote Volume Mirroring Feature by Specifying a Number of Drives

With this version of the `activate storageArray feature=remoteMirror` command, you must specify the number of drives and the RAID level that you want for the mirror repository volume. This version of the command creates a new volume group. For this command to work, you must have drives in the storage array that are not assigned to a volume group.

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
[driveType=(fibre | SATA | SAS | PATA)]
[trayLossProtect=(TRUE | FALSE)]
```

This example shows a command in which you specify the number of drives:

```
c:\...\smee\client>smcli 123.45.67.89 -c "activate storageArray
feature=remoteMirror repositoryRAIDLevel=5 repositoryDriveCount=5
driveType=fibre;"
```

The command in this example creates a new mirror repository volume by using five software-selected drives for the mirror repository volume. The mirror repository volume has a RAID level of 5. The type of drive for the mirror repository volume is Fibre Channel.

This example shows how to use the command in a script file:

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=5 repositoryDriveCount=5
driveType=fibre;
```

Determining Candidates for a Remote-Mirror Pair

All of the volumes and drives on the remote storage array might not be available for use as secondary volumes. To determine which volumes on a remote storage array that you can use as candidates for secondary volumes, use the `show remoteMirror candidates` command. This command returns a list of the volumes that you can use when creating a remote mirror.

The command takes this form:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show
remoteMirror candidates primary=\"volumeName\"
remoteStorageArrayName=\"storageArrayName\";"
```

where *volumeName* is the name of the volume that you want to use for the primary volume, and *storageArrayName* is the remote storage array that contains possible candidates for the secondary volume. Enclose both the volume name and the storage array name in double quotation marks (" ").

Creating a Remote-Mirror Pair

When you create a new remote mirror, you must define which volumes that you want to use for the primary (local) volume and the secondary (remote) volume. You define the primary volume by the name of the volume. You define the secondary volume by name with either the name or the World Wide Identifier (WWID) of the storage array on which the secondary volume resides. The primary volume name, the secondary volume name, and the remote storage array name (or WWID) are the minimum information that you need to provide. Using this command, you can also define synchronization priority, write order, and write mode.

The command takes this form:

```
create remoteMirror primary="primaryVolumeName"
secondary="secondaryVolumeName"
(remoteStorageArrayName="storageArrayName" |
remoteStorageArrayWwn="wwID") remotePassword=password
syncPriority=(highest | high | medium | low | lowest)
writeOrder=(preserved | notPreserved)
writeMode=(synchronous | asynchronous)
```

NOTE You can use the optional parameters as needed to help define your configuration.

This example shows the `create remoteMirror` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create
remoteMirror primary=\"Jan_04_Account\"
secondary=\"Jan_04_Account_B\" remoteStorageArrayName=\"Tabor\"
remotePassword=\"jdw2ga05\" syncPriority=highest
writeMode=synchronous;"
```

The command in this example creates a remote mirror in which the primary volume is named Jan_04_Account on the local storage array. The secondary volume is named Jan_04_Account_B on the remote storage array that is named Tabor. The names used in this example are similar, but that is not a requirement for the volume names in a remote-mirror pair. In this example, the remote storage array has a password that you must enter when making any change to the storage array configuration. Creating a remote-mirror pair is a significant change to a storage array configuration. Setting the write mode to **synchronous** and the synchronization priority to **highest** means that host write requests are written to the primary volume and then immediately copied to the secondary volume. These actions help to make sure that the data on the secondary volume is as accurate a copy of the data on the primary volume as possible. The highest synchronization priority does, however, use more system resources, which can reduce system performance.

This example shows how to use the command in a script file:

```
create remoteMirror primary="Jan_04_Account"
secondary="Jan_04_Account_B" remoteStorageArrayName="Tabor"
remotePassword="jdw2ga05" syncPriority=highest
writeMode=synchronous;
```

After you have created a remote mirror, you can see the progress of data synchronization between the primary volume and the secondary volume by running the `show remoteMirror synchronizationProgress` command. This command shows the progress as a percentage of data synchronization that has completed.

Changing Remote Volume Mirroring Settings

The `set remoteMirror` command lets you change the property settings for a remote-mirror pair. Use this command to change these property settings:

- The volume role (either primary or secondary)
- The synchronization priority
- The write order
- The write mode

You can apply the changes to one or several remote-mirror pairs by using this command. Use the primary volume name to identify the remote-mirror pairs for which you are changing the properties.

This example shows how to use the `set remoteMirror` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set remoteMirror
localVolume [Jan_04_Account] syncPriority=medium writeOrder=notpreserved
writeMode=asynchronous;"
```

This example shows how to use the command in a script file:

```
set remoteMirror localVolume [Jan_04_Account]
syncPriority=medium writeOrder=notpreserved
writeMode=asynchronous;
```

Suspending and Resuming a Mirror Relationship

Use the `suspend remoteMirror` command to stop data transfer between a primary volume and a secondary volume in a mirror relationship without disabling the mirror relationship. Suspending a mirror relationship lets you control when the data on the primary volume and data on the secondary volume are synchronized. Suspending a mirror relationship helps to reduce any performance impact to the host application that might occur while any changed data on the primary volume is copied to the secondary volume. Suspending a mirror relationship is particularly useful when you want to run a backup of the data on the secondary volume.

When a mirror relationship is in a suspended state, the primary volume does not make any attempt to contact the secondary volume. Any writes to the primary volume are persistently logged in the mirror repository volumes. After the mirror relationship resumes, any data that is written to the primary volume is automatically written to the secondary volume. Only the modified data blocks on the primary volume are written to the secondary volume. Full synchronization is not required.

IMPORTANT If you suspend a remote mirror that is set up in the Write consistency mode, you suspend all remote-mirror pairs within the group. You can then resume mirror operations for any of the individual remote-mirror pairs that are in the group.

This example shows the `suspend remoteMirror` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "suspend remoteMirror primary
Jan_04_Account writeConsistency=false;"
```

The `writeConsistency` parameter defines whether the volumes identified in this command are in a write-consistency group or are separate. For the volumes in a write-consistency group, set this parameter to **TRUE**. For the volumes that are not in a write-consistency group, set this parameter to **FALSE**.

This example shows how to use the command in a script file:

```
suspend remoteMirror volume Jan_04_Account writeConsistency=false;
```

The mirror relationship remains suspended until you use the `resume remoteMirror` command to restart synchronization activities. This command restarts data transfers between a primary volume and a secondary volume in a mirror relationship after the mirror has been suspended or unsynchronized.

This example shows the `resume remoteMirror` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "resume
remoteMirror volume Jan_04_Account writeConsistency=false;"
```

The `writeConsistency` parameter in this command operates the same as in the previous command.

This example shows how to use the command in a script file:

```
resume remoteMirror volume Jan_04_Account
writeConsistency=false;
```

Removing a Mirror Relationship

Use the `remove remoteMirror` command to remove the link between a primary volume and a secondary volume. (Removing a mirror relationship is similar to deleting a mirror relationship.) Removing the link between a primary volume and a secondary volume does not affect any of the existing data on either volume. The link between the volumes is removed, but the primary volume still continues normal I/O operations. Later, you can establish the mirror relationship between the two volumes and resume normal mirror operations. You can remove the mirror relationship for one or several remote-mirror pairs with this command.

This example shows the `remove remoteMirror` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "remove remoteMirror localVolume
[Jan_04_Account];"
```

When you run this command, use the name of the primary volume of the remote-mirror pair.

This example shows how to use the command in a script file:

```
remove remoteMirror localVolume [Jan_04_Account];
```

To re-establish the link between a primary volume and a secondary volume, use the `create remoteMirror` command.

Deleting a Primary Volume or a Secondary Volume

Use the `delete volume` command to remove a primary volume or a secondary volume from a storage array. Deleting a volume in a mirror relationship removes the mirror relationship and completely deletes the volume from the storage array. You cannot redefine the mirror relationship until you create a new volume or choose an alternate volume to replace the deleted volume.

ATTENTION **Possible loss of data access** – Deleting a primary volume or a secondary volume permanently removes the data from the storage array.

Disabling the Remote Volume Mirroring Feature

You disable the Remote Volume Mirroring feature to prevent the new mirror relationship from being created. When you disable the Remote Volume Mirroring feature, the feature is in a Disabled/Active state. In this state, you can maintain and manage previously existing mirror relationships; however, you cannot create new relationships. To disable the Remote Volume Mirroring feature, use this command:

```
disable storageArray feature=remoteMirror
```

Deactivating the Remote Volume Mirroring Feature

If you no longer require the Remote Volume Mirroring feature and you have removed all of the mirror relationships, you can deactivate the feature. Deactivating the feature re-establishes the normal use of dedicated ports on both storage arrays and deletes both mirror repository volumes. To deactivate the Remote Volume Mirroring feature, use this command:

```
deactivate storageArray feature=remoteMirror
```

Interaction with Other Features

You can run the Remote Volume Mirroring feature while running these premium features:

- SANshare Storage Partitioning
- Snapshot Volume
- Volume Copy

When you run the Remote Volume Mirroring feature with other premium features, you must consider the requirements of the other premium features to help make sure that you set up a stable storage array configuration.

In addition to running with the premium features, you can also run the Remote Volume Mirroring feature while running Dynamic Volume Expansion (DVE).

SANshare Storage Partitioning

SANshare Storage Partitioning is a premium feature that lets hosts share access to volumes in a storage array. You create a storage partition when you define any of these logical components in a storage array:

- A host
- A host group
- A volume-to-LUN mapping

The volume-to-LUN mapping lets you define which host group or host has access to a particular volume in the storage array.

When you create storage partitions, define the partitions after you have created the primary volume and the secondary volume in a Remote Volume Mirroring configuration. The storage partition definitions for the primary storage array and the secondary storage array are independent of each other. If these definitions are put in place while the volume is in a secondary role, the administrative effort associated with the site recovery is reduced if it becomes necessary to promote the volume to a primary role.

Snapshot Volumes

A snapshot volume is a point-in-time image of a volume. Typically, it is created so that an application, such as a backup application, can access the snapshot volume and read the data while the base volume stays online and is accessible to hosts.

The volume for which the point-in-time image is created is known as the base volume and must be a standard volume in the storage array. The snapshot repository volume stores information about all data that changed since the snapshot was created.

In this version of the storage management software, you can create snapshot volumes based on the primary volume or secondary volume of a remote mirror.

Volume Copy

The Volume Copy premium feature copies data from one volume (the source volume) to another volume (the target volume) within a single storage array. You can use this feature to perform these functions:

- Copy data from volume groups that use smaller-capacity drives to volume groups that use larger-capacity drives
- Back up data
- Restore snapshot volume data to the base volume.

You can use a primary volume in a remote mirror as a source volume or a target volume in a volume copy. You cannot use a secondary volume as a source volume or a target volume.

NOTE If you start a role reversal during a copy-in-progress, the copy fails and cannot be restarted.

Dynamic Volume Expansion

A Dynamic Volume Expansion (DVE) is a modification operation that increases the capacity of a standard volume or a snapshot repository volume. The increase in capacity is achieved by using the free capacity that is available in the volume group of the standard volume or the snapshot repository volume.

This modification operation is considered to be “dynamic” because you can continually access data on volume groups, volumes, and drives throughout the entire operation.

A DVE operation can be performed on a primary volume or a secondary volume of a mirror relationship.

NOTE Although the storage management software indicates that the volume has increased capacity, its usable capacity is the size of the smaller of the primary volume or the secondary volume.

You cannot perform a DVE operation on a mirror repository volume.

Using the Volume Copy Premium Feature

The Volume Copy premium feature lets you copy data from one volume (the source) to another volume (the target) in a single storage array. You can use this feature to perform these tasks:

- Back up data
- Copy data from volume groups that use smaller-capacity drives to volume groups using greater-capacity drives
- Restore snapshot volume data to the associated base volume

How Volume Copy Works

When you create a volume copy, you create a copy pair that consists of a source volume and a target volume. Both the source volume and the target volume are located on the same storage array. During a volume copy, the controllers manage copying the data from the source volume to the target volume. The volume copy is transparent to the host machines and applications, except that users cannot write to the source volume during a volume copy.

While a volume copy is In Progress, the same controller must own both the source volume and the target volume. If one controller does not own both the source volume and the target volume before creating the volume copy, ownership of the target volume is automatically transferred to the controller that owns the source volume. When the volume copy is finished or stopped, ownership of the target volume is restored to its preferred controller. If ownership of the source volume changes while a volume copy is running, ownership of the target volume also changes.

Source Volume

The source volume is the volume that accepts host I/O and stores data. When you start a volume copy, data from the source volume is copied in its entirety to the target volume. While a volume copy has a status of In Progress, Pending, or Failed, the source volume is available only for read activity.

After the volume copy completes, the source volume becomes available to host applications for write requests. The target volume automatically becomes read only to hosts, and write requests to the target volume are rejected.

The following are valid source volumes:

- A standard volume
- A snapshot volume
- The base volume of a snapshot volume
- A primary volume that is participating in a remote-mirror pair

The following are not valid source volumes:

- A secondary volume that is participating in a remote-mirror pair
- A snapshot repository volume
- A mirror repository volume
- A failed volume
- A missing volume
- A volume currently in a modification operation
- A volume that is holding a Small Computer System Interface-2 (SCSI-2) reservation or a persistent reservation
- A volume that is a source volume or a target volume in another volume copy that has a status of In Progress, Pending, or Failed

Target Volume

A target volume contains a copy of the data from the source volume. When a volume copy is started, data from the source volume is copied in its entirety to the target volume.

ATTENTION Possible loss of data access – A volume copy overwrites data on the target volume. Before you start a new operation, make sure that you no longer need the old data, or you have backed up the old data on the target volume.

While the volume copy has a status of In Progress, Pending, or Failed, the controllers reject read and write requests to the target volume. After the volume copy operation is finished, the target volume automatically becomes read only to the hosts, and write requests to the target volume are rejected. You can change the Read-Only attribute after the volume copy has completed or has been stopped. (For more information about the Read-Only attribute, see “[Chapter 6, Viewing Volume Copy Properties](#)” on page 109.)

The following volumes are valid target volumes:

- A standard volume
- The base volume of a disabled snapshot volume or failed snapshot volume
- A primary volume that is participating in a remote-mirror pair

The following volumes are not valid target volumes:

- The base volume of an active snapshot volume
- A snapshot volume
- A mirror repository volume
- A snapshot repository volume
- A secondary volume in a remote-mirror pair
- A failed volume
- A missing volume
- A volume with a status of Degraded
- A volume that is currently in a modification operation

- A volume that is holding a SCSI-2 reservation or a persistent reservation
- A volume that is a source volume or a target volume in another volume copy that has a status of In Progress, Pending, or Failed

Volume Copy and Persistent Reservations

You cannot use volumes that hold persistent reservations for either a source volume or a target volume. Persistent reservations are configured and managed through the server cluster software and prevent other hosts from accessing the reserved volume. Unlike other types of reservations, a persistent reservation reserves host access to the volume across multiple HBA host ports, which provides various levels of access control.

To determine which volumes have reservations, run the `show (volume) reservations` command. To remove a reservation, run the `clear (volume) reservations` command.

Storage Array Performance

During a volume copy operation, the resources of the storage array might be diverted from processing I/O activity to completing a volume copy, which might affect the overall performance of the storage array.

These factors contribute to the performance of the storage array:

- The I/O activity
- The volume RAID level
- The volume configuration (number of drives in the volume groups and cache parameters)
- The volume type (snapshot volumes might take more time to copy than standard volumes)

When you create a new volume copy, you define the copy priority to determine how much controller processing time is allocated for a volume copy compared with I/O activity.

Copy priority has five relative settings ranging from highest to lowest. The highest priority rate supports the volume copy, but I/O activity might be affected. The lowest priority rate supports I/O activity, but the volume copy takes longer. You define the copy priority when you create the volume copy pair. You can redefine the copy priority later by using the `set volumeCopy` command. You can also redefine the volume copy priority when you recopy a volume.

Restrictions

These restrictions apply to the source volume, the target volume, and the storage array:

- While a volume copy operation has a status of In Progress, Pending, or Failed, the source volume is available for read activity only. After the volume copy finishes, read activity from and write activity to the source volume is permitted.
- A volume can be selected as a target volume for only one volume copy at a time.
- The maximum allowable number of volume copies per storage array depends upon the storage array configuration.
- A volume that is reserved by the host cannot be selected as a source volume or as a target volume.
- A volume with a status of Failed cannot be used as a source volume or as a target volume.
- A volume with a status of Degraded cannot be used as a target volume.
- You cannot select a volume that is participating in a modification operation as a source volume or as a target volume. Modification operations include Dynamic Capacity Expansion (DCE), Dynamic RAID Level Migration (DRM), Dynamic Segment Sizing (DSS), Dynamic Volume Expansion (DVE), and defragmenting a volume group.

Volume Copy Commands

[Table 20](#) lists the Volume Copy commands and briefly describes what the commands do.

Table 20 Volume Copy Commands

Command	Description
<code>create volumeCopy</code>	Creates a volume copy and starts the volume copy operation.
<code>disable storageArray featurevolumeCopy</code>	Turns off the current volume copy operation.
<code>enable storageArray feature</code>	Activates the Volume Copy premium feature.
<code>recopy volumeCopy</code>	Re-initiates a volume copy operation using an existing volume copy pair.

Command	Description
<code>remove volumeCopy</code>	Removes a volume copy pair.
<code>set volumeCopy</code>	Defines the properties for a volume copy pair.
<code>show volumeCopy</code>	Returns information about volume copy operations. You can retrieve information about a specific volume copy pair, or all of the volume copy pairs in the storage array.
<code>show volumeCopy sourceCandidates</code>	Returns information about the candidate volumes that you can use as the source for a volume copy operation.
<code>show volumeCopy targetCandidates</code>	Returns information about the candidate volumes that you can use as the target for a volume copy operation.
<code>stop volumeCopy</code>	Stops a volume copy operation.

Creating a Volume Copy

Before you create a volume copy, make sure that a suitable target volume exists on the storage array, or create a new target volume specifically for the volume copy. The target volume that you use must have a capacity equal to or greater than the source volume.

You can have a maximum of eight volume copies with a status of In Progress at one time. Any volume copy greater than eight has a status of Pending until one of the volume copies with a status of In Progress has completed the volume copy process.

To create a volume copy, perform these general steps:

- 1 Enable the Volume Copy premium feature.
- 2 Determine the candidates for a volume copy.
- 3 Create the target volume and the source volume for the volume copy.

Enabling the Volume Copy Feature

The first step in creating a volume copy is to make sure that the feature is enabled on the storage array. Because Volume Copy is a premium feature, you need a Feature Key file to enable the feature. This command enables the Feature Key file:

```
enable storageArray feature file="filename"
```

where the `file` parameter is the complete file path and file name of a valid Feature Key file. Enclose the file path and file name in double quotation marks (" "). Valid file names for Feature Key files usually end with a `.key` extension.

Determining Volume Copy Candidates

All volumes and drives might not be available for use in volume copy operations. To determine which candidate volumes on the storage array that you can use as a source volume, use the `show volumeCopy sourceCandidates` command. To determine which candidate volumes on the storage array that you can use as a target volume, use the `show volumeCopy targetCandidates` command. These commands return a list of the drive tray, slot, and capacity information for the source volume candidates and the target volume candidates. You can use the `show volumeCopy sourceCandidates` command and the `show volumeCopy targetCandidates` command only after you have enabled the Volume Copy premium feature.

Creating a Volume Copy

ATTENTION **Possible loss of data access** – A volume copy overwrites data on the target volume. Make sure that you no longer need the data or have backed up the data on the target volume before you start a volume copy operation.

When you create a volume copy, you must define which volumes that you want to use for the source volume and the target volume. You define the source volume and the target volume by the name of each volume. You can also define the copy priority and choose whether you want the target volume to be read only after the data is copied from the source volume.

The command takes this form:

```
create volumeCopy
source="sourceName" target="targetName"
[copyPriority=(highest | high | medium | low | lowest)
targetReadOnlyEnabled=(TRUE | FALSE)]
```

Before you run the `create volumeCopy` command, perform these actions:

- Stop all I/O activity to the source volume and the target volume.
- Dismount any file systems on the source volume and the target volume, if applicable.

This example shows the `create volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "create
volumeCopy source=\"Jaba_Hut\" target=\"Obi_1\"
copyPriority=medium targetReadOnlyEnabled=TRUE;"
```

The command in this example copies the data from the source volume named `Jaba_Hut` to the target volume named `Obi_1`. Setting the copy priority to **medium** provides a compromise between how quickly the data is copied from the source volume to the target volume and the amount of processing resources that are required for data transfers to other volumes in the storage array. Setting the `targetReadOnlyEnabled` parameter to **TRUE** means that write requests cannot be made to the target volume, making sure that the data on the target volume stays unaltered.

This example shows how to use the command in a script file:

```
create volumeCopy source="Jaba_Hut" target="Obi_1"
copyPriority=medium targetReadOnlyEnabled=TRUE;
```

After the volume copy operation is completed, the target volume automatically becomes read only to hosts. Any write requests to the target volume are rejected, unless you disable the Read-Only attribute by using the `set volumeCopy` command.

To view the progress of a volume copy, use the `show volume actionProgress` command. This command returns information about the volume action, the percentage completed, and the time remaining until the volume copy is complete.

Viewing Volume Copy Properties

Use the `show volumeCopy` command to view information about one or more selected source volumes or target volumes. This command returns these values:

- The role
- The copy status
- The start time stamp
- The completion time stamp
- The copy priority
- The Read-Only attribute setting for the target volume
- The source volume World Wide Identifier (WWID) or the target volume WWID

If a volume is participating in more than one volume copy (it can be a source volume for one volume copy operation and a target volume for another volume copy operation), the details are repeated for each associated copy pair.

The command takes this form:

```
show volumeCopy (allVolumes | source [sourceName] |
target [targetName])
```

This example shows the `show volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show volumeCopy source
[\"JabaHut\"];"
```

The command in this example is requesting information about the source volume `Jaba_Hut`. If you wanted information about all of the volumes, you would use the `allVolumes` parameter. You can also request information about a specific target volume.

This example shows how to use the command in a script file:

```
show volumeCopy source [\"Jaba_Hut\"];
```

Changing Volume Copy Settings

The `set volumeCopy` command lets you change these property settings for a volume copy pair:

- The copy priority
- The target volume read/write permission

Copy priority has five relative settings ranging from highest to lowest. The highest priority supports the volume copy, but I/O activity might be affected. The lowest priority supports I/O activity, but the volume copy takes longer. You can change the copy priority at these times:

- Before the volume copy operation starts
- While the volume copy operation has a status of In Progress
- After the volume copy operation has completed when re-creating a volume copy operation by using the `recopy volumeCopy` command

When you create a volume copy pair and after the original volume copy has completed, the target volume is automatically defined as read-only to the hosts. The read-only status of the target volume helps to make sure that the copied data on the target volume is not corrupted by additional writes to the target volume after the volume copy is created. You want to maintain the read-only status when you are performing these tasks:

- Using the target volume for backup purposes
- Copying data from one volume group to a larger volume group for greater accessibility
- Planning to use the data on the target volume to copy back to the base volume in case of a disabled snapshot volume or failed snapshot volume

At other times, you might want to write additional data to the target volume. You can use the `set volumeCopy` command to reset the read/write permission for the target volume.

NOTE If you have set the volume copy parameters to enable host writes to the target volume, the read request and the write request to the target volume are rejected while the volume copy operation has a status of In Progress, Pending, or Failed.

The command takes this form:

```
set volumeCopy target [targetName] [source [sourceName]]
copyPriority=(highest | high | medium | low | lowest)
targetReadOnlyEnabled=(TRUE | FALSE)
```

NOTE You can use the parameters as needed to help define your configuration.

This example shows the `set volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "set volumeCopy
target [\"Obi_1\"] copyPriority=highest
targetReadOnlyEnabled=FALSE;"
```

This example shows how to use the command in a script file:

```
set volumeCopy target ["Obi_1"] copyPriority=highest
targetReadOnlyEnabled=FALSE;
```

Recopying a Volume

Use the `recopy volumeCopy` command to create a new volume copy for a previously defined copy pair that has a status of Stopped, Failed, or Completed. You can use the `recopy volumeCopy` command to create backups of the target volume. Then, you can copy the backup to tape for off-site storage. When you use the `recopy volumeCopy` command to make a backup, you cannot write to the source volume while the recopy operation is running. The recopy operation might take a long time.

When you run the `recopy volumeCopy` command, the data on the source volume is copied in its entirety to the target volume.

ATTENTION Possible loss of data access – The `recopy volumeCopy` command overwrites existing data on the target volume and makes the target volume read-only to hosts. The `recopy volumeCopy` command fails all of the snapshot volumes that are associated with the target volume, if any exist.

You can also reset the copy priority by using the `recopy volumeCopy` command if you want to change the copy priority for the recopy operation. The higher priorities allocate storage array resources to the volume copy at the expense of storage array performance.

The command takes this form:

```
recopy volumeCopy target [targetName] [source [sourceName]
copyPriority=(highest | high | medium | low | lowest)
targetReadOnlyEnabled=(TRUE | FALSE)]
```

NOTE You can use the optional parameters as needed to help define your configuration.

This example shows the `show volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "recopy volumeCopy target
[\"Obi_1\"] copyPriority=highest;"
```

The command in this example copies data from the source volume that is associated with the target volume `Obi_1` to the target volume again. The copy priority is set to the highest value to complete the volume copy as quickly as possible. The underlying consideration for using this command is that you have already created the volume copy pair, which has already created one volume copy. By using this command, you are copying the data from the source volume to the target volume with the assumption that the data on the source volume has changed since the previous copy was made.

This example shows you how to use the command in a script file:

```
recopy volumeCopy target ["Obi_1"] copyPriority=highest;
```

Stopping a Volume Copy

The `stop volumeCopy` command lets you stop a volume copy that has a status of `In Progress`, `Pending`, or `Failed`. After you have stopped a volume copy, you can use the `recopy volumeCopy` command to create a new volume copy by using the original volume copy pair. After you stop a volume copy operation, all of the mapped hosts will have write access to the source volume.

The command takes this form:

```
stop volumeCopy target [targetName] [source [sourceName]]
```

This example shows the `show volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "stop volumeCopy target
[\"Obi_1\"];"
```

This example shows how to use the command in a script file:

```
stop volumeCopy target ["Obi_1"];
```

Removing Copy Pairs

The `remove volumeCopy` command lets you remove a volume copy pair from the storage array configuration. All of the volume copy-related information for the source volume and the target volume is removed from the storage array configuration. The data on the source volume or the target volume is not deleted. Removing a volume copy from the storage array configuration also removes the Read-Only attribute for the target volume.

IMPORTANT If the volume copy has a status of In Progress, you must stop the volume copy before you can remove the volume copy pair from the storage array configuration.

The command takes this form:

```
remove volumeCopy target [targetName] [source [sourceName]]
```

This example shows the `remove volumeCopy` command:

```
c:\...\smee\client>smcli 123.45.67.89 -c "remove volumeCopy target  
[\"Obi_1\"];"
```

This example shows how to use the command in a script file:

```
remove volumeCopy target [\"Obi_1\"];
```

Interaction with Other Features

You can run the Volume Copy premium feature while running the following premium features:

- SANshare Storage Partitioning
- Snapshot Volume
- Remote Volume Mirroring

When you are running the Volume Copy feature with other premium features, you must consider the requirements of other premium features to help make sure that you set up a stable storage array configuration.

In addition to the premium features, you also can run the Volume Copy feature while running Dynamic Volume Expansion (DVE).

SANshare Storage Partitioning

SANshare Storage Partitioning is a premium feature that lets hosts share access to volumes in a storage array. You create a storage partition when you define any of these logical components in a storage array:

- A host
- A host group
- A volume-to-LUN mapping

The volume-to-LUN mapping lets you define which host group or host has access to a particular volume in the storage array.

After you create a volume copy, the target volume automatically becomes read only to hosts to make sure that the data is preserved. Hosts that have been mapped to a target volume do not have write access to the volume, and any attempt to write to the read-only target volume results in a host I/O error.

If you want hosts to have write access to the data on the target volume, use the `set volumeCopy` command to disable the Read-Only attribute for the target volume.

Snapshot Volumes

A snapshot volume is a point-in-time image of a volume. It is usually created so that an application, such as a backup application, can access the snapshot volume and read the data while the base volume stays online and is accessible to hosts.

The volume for which the point-in-time image is created is known as the base volume and must be a standard volume in the storage array. The snapshot repository volume stores information about all of the data that changed since the snapshot was created.

You can select snapshot volumes as the source volumes for a volume copy. This selection is a good use of this feature, because it performs complete backups without significant impact to the storage array I/O. Some I/O processing resources are lost to the copy operation.

IMPORTANT If you choose the base volume of a snapshot volume as your target volume, you must disable all of the snapshot volumes that are associated with the base volume before you can select it as a target volume.

When you create a snapshot volume, a snapshot repository volume is automatically created. The snapshot repository volume stores information about the data that has changed since the snapshot volume was created. You cannot select a snapshot repository volume as a source volume or a target volume in a volume copy.

You can use the Snapshot Volume premium feature with the Volume Copy premium feature to back up data on the same storage array and to restore the data on the snapshot volume back to its original base volume.

Remote Volume Mirroring

The Remote Volume Mirroring premium feature provides for online, real-time replication of data between storage arrays over a remote distance. In the event of a disaster or a catastrophic failure of one storage array, you can promote a secondary storage array to take over responsibility for data storage.

When you create a remote mirror, a remote-mirror pair is created, which consists of a primary volume at the primary storage array and a secondary volume at a remote storage array.

The primary volume is the volume that accepts host I/O and stores data. When the mirror relationship is initially created, data from the primary volume is copied in its entirety to the secondary volume. This process is known as a full synchronization and is directed by the controller owner of the primary volume. During a full synchronization, the primary volume remains fully accessible for all normal I/O activity.

The controller owner of the primary volume starts remote writes to the secondary volume to keep the data on the two volumes synchronized. Whenever the data on the primary volume and the secondary volume becomes unsynchronized, the controller owner of the primary volume starts a resynchronization, where only the data that changed during the interruption is copied.

The secondary volume maintains a mirror of the data on its associated primary volume. The controller owner of the secondary volume receives remote writes from the controller owner of the primary volume but does not accept host write requests.

The secondary volume stays available to host applications as read-only while mirroring is underway. In the event of a disaster or a catastrophic failure at the primary site, you can perform a role reversal to promote the secondary volume to a primary role. Hosts are then able to access the newly promoted volume, and business operations can continue.

You can select a primary volume that is participating in a remote-mirror pair to be used as the source volume or a target volume for a volume copy. A secondary volume that is participating in a remote-mirror pair cannot be selected as a source volume or a target volume.

Role Reversals

A role reversal is the act of promoting the secondary volume to be the primary volume of the remote-mirror pair, and demoting the primary volume to be the secondary volume.

In the event of a disaster at the storage array that contains the primary volume, you can fail over to the secondary site by performing a role reversal to promote the secondary volume to the primary volume role. This action lets hosts continue to access data, and business operations can continue.

Trying a role reversal in which the original primary volume is the source volume for an active volume copy (the status is In Progress or Pending) causes the volume copy to fail. The failure occurs when the original primary volume becomes the new secondary volume. You cannot restart the volume copy until you return the roles of the volumes back to their original state; that is, the volume that was originally the primary volume is set once again to be the primary volume.

If the primary storage array is recovered but is unreachable due to a link failure, a forced promotion of the secondary volume will result in both the primary volume and the secondary volume viewing themselves in the primary volume role (dual-primary condition). If this condition occurs, the volume copy in which the primary volume is participating is unaffected by the role change.

You can perform a role reversal by using the `set remoteMirror` command.

- To change a secondary volume to a primary volume, use this command, which promotes the selected secondary volume to become the primary volume of the remote-mirror pair. Use this command after a catastrophic failure has occurred.

```
set remoteMirror role=primary
```

- To change a primary volume to a secondary volume, use this command, which demotes the selected primary volume to become the secondary volume. Use this command after a catastrophic failure has occurred.

```
set remoteMirror role=secondary
```

Maintaining a Storage System

Maintenance covers a broad spectrum of activity with the goal of keeping a storage array operational and available to all hosts. This chapter provides descriptions of commands you can use to perform storage array maintenance. The commands are organized into four sections:

- Routine maintenance
- Performance tuning
- Troubleshooting and diagnostics
- Recovery operations

The organization is not a rigid approach, and you can use the commands as appropriate for your storage array. The commands listed in this chapter do not cover the entire array of commands you can use for maintenance. Other commands, particularly the `set` commands, can provide diagnostic or maintenance capabilities.

Routine Maintenance

Routine maintenance involves those tasks that you might perform periodically to make sure that the storage array is running as well as possible or to detect conditions before they become problems.

Running a Media Scan

Media scan provides a way of detecting drive media errors before they are found during a normal read from or write to the drives. Any media scan errors that are detected are reported to the Event Log. The Event Log provides an early indication of an impending drive failure and reduces the possibility of encountering a media error during host operations. A media scan is performed as a background operation and scans all data and redundancy information in defined user volumes.

A media scan runs on all of the volumes in the storage array that have these conditions:

- Has Optimal status
- Has no modification operations in progress
- Has media scan enabled

Errors that are detected during a scan of a user volume are reported to the Major Event Log (MEL) and handled as follows:

- **Unrecovered media error** – The drive could not read the requested data on its first try or on any subsequent retries. The result of this action is that for volumes with redundancy protection, the data is reconstructed, rewritten to the drive, and verified, and the error is reported to the Event Log. For volumes without redundancy protection, the error is not corrected, but it is reported to the Event Log.
- **Recovered media error** – The drive could not read the requested data on its first attempt. The result of this action is that the data is rewritten to the drive and verified. The error is reported to the Event Log.
- **Redundancy mismatches** – Redundancy errors are found, and a media error is forced on the block stripe so that it is found when the drive is scanned again. If redundancy is repaired, this forced media error is removed. The result of this action is that the first 10 redundancy mismatches found on a volume are reported to the Event Log.
- **Unfixable error** – The data could not be read, and parity information or redundancy information could not be used to regenerate it. For example, redundancy information cannot be used to reconstruct data on a degraded volume. The result of this action is that the error is reported to the Event Log.

The script command set provides two commands to define media scan properties:

- `set volume`
- `set storageArray`

The `set volume` command enables a media scan for the volume. The command takes this form:

```
set (allVolumes | volume [volumeName] | volumes [volumeName1 ... volumeNameN]
| volume <wwID>) mediaScanEnabled=(TRUE | FALSE)
```

The `set storageArray` command defines how frequently a media scan is run on a storage array. The command takes this form:

```
set storageArray mediaScanRate=(disabled | 1-30)
```

The `mediaScanRate` values define the number of days over which the media scan runs. Valid values are **disabled**, which turns off the media scan; or **1** day to **30** days, where **1** day is the fastest scan rate, and **30** days is the slowest. A value other than what is shown will not allow the media scan to function.

Running a Redundancy Check

Redundancy checks are performed when media scans are run. (For a description about how to set up and run media scans, see “[Chapter 7, Running a Media Scan](#)” on page 117.) During a redundancy check, all of the data blocks in a volume are scanned, and, depending on the RAID level, deteriorated data is corrected. Correction is performed as follows:

- For RAID 3, RAID 5, or RAID 6 volumes, redundancy is checked and repaired.
- For RAID 1 volumes, the data is compared between the mirrored drives and data inconsistencies are repaired.
- RAID 0 volumes have no redundancy.

Before you can run a redundancy check, you must enable redundancy checking by using the `set volume` command. The command takes this form:

```
set (allVolumes | volume [volumeName] | volumes [volumeName1 ... volumeNameN]
 | volume <wwID>) redundancyCheckEnabled=(TRUE | FALSE)
```

Resetting a Controller

IMPORTANT When you reset a controller, the controller is no longer available for I/O operations until the reset is complete. If a host is using volumes that are owned by the controller being reset, the I/O that is directed to the controller is rejected. Before resetting the controller, either make sure that the volumes that are owned by the controller are not in use, or make sure that a multi-path driver is installed on all of the hosts that are using these volumes.

Resetting a controller is the same as rebooting the controller processors. To reset a controller, use this command:

```
reset controller [(a | b)]
```

Enabling a Controller Data Transfer

At times, a controller might become quiescent while running diagnostics. If this condition occurs, the controller might become unresponsive. To revive a controller that has become quiescent while running diagnostics, use this command:

```
enable controller [(a | b)] dataTransfer
```

Resetting the Battery Age

After you have replaced the batteries in the storage array, you must reset the age of the battery. You can reset either the batteries for an entire storage array or a battery in a specific controller. To reset the age of the batteries to zero days, use this command:

```
reset storageArray batteryInstallDate [controller=(a | b)]
```

Removing Persistent Reservations

Persistent reservations preserve volume registrations, and they prevent hosts, other than the host defined for the volume, from accessing the volume. You must remove persistent reservations before you make these changes to your configuration:

- Change or delete LUN mappings on a volume holding a reservation
- Delete volume groups or volumes that have any reservations

To determine which volumes have reservations, use this command:

```
show (allVolumes | volume [volumeName] | volumes [volumeName1 ...  
volumeNameN]) reservations
```

To clear persistent volume reservations, use this command:

```
clear (allVolumes | volume [volumeName] | volumes [volumeName1 ...  
volumeNameN]) reservations
```

Synchronizing the Controller Clocks

To synchronize the clocks on both controllers in a storage array with the host clock, use this command:

```
set storageArray time
```

Locating Drives

At times, you might need to locate a specific drive. In very large storage array configurations, this task can sometimes be awkward. If you need to locate a specific drive, you can do so by turning on the indicator light on the front of the drive. To locate a drive, use this command:

```
start drive [trayID,slotID] locate
```

To turn off the indicator light after locating the drive, use this command:

```
stop drive locate
```

Relocating a Volume Group

Volume group relocation describes the action of moving drives within the same storage array. This is a supported capability; however, any relocation of storage array components must be completed under the guidance of a Customer and Technical Support representative.

This section describes the commands that you use to remove a set of drives and then reinstall them into a different storage array.

Hot and Cold Volume Group Relocation

There are two methods you can use to move volume groups: hot volume group relocation and cold volume group relocation.

- Hot volume group relocation lets you add or move storage without reconfiguring the storage array and, in some cases, without rebooting. During hot volume group relocation, the storage array power is not turned off.
- Cold volume group relocation requires that the power to the source storage array and the destination storage array be turned off before moving the volume groups from one storage array to another. Then the power to the storage arrays can be turned on.

To ensure that any volume group being moved to a different destination storage array is correctly recognized and managed by the new storage array, use hot volume group relocation whenever possible.

ATTENTION Possible loss of data access – You must move a single volume group at a time, and it must go into a storage array with the same level of controller firmware.

Basic Process Steps

Relocating a volume group includes these procedures:

- 1 Verifying the status of the storage array
- 2 Locating the drives in the volume group
- 3 Placing the volume group offline
- 4 Removing drives from the storage array
- 5 Replacing a volume group into the new storage array

To perform these steps, you must be familiar with the following CLI commands. The command syntax is provided to assist in your use of these new commands.

Volume Group Relocation Commands

Use the following command to place a specific storage array into an exported state so that its drives may be removed.

```
start volumeGroup [user-label] export
```

At this point you are allowed to remove the drives that comprise the volume group, and physically reinstall them into a different storage array.

Use the following command to logically move a specific storage array from an exported state to the complete state.

```
start volumeGroup [user-label] import
```

Your relocated volume group is now available for use.

For additional information, refer to these commands:

- “[Chapter 8, Show Volume Group Export Dependencies](#)” on page 318
- “[Chapter 8, Show Volume Group Import Dependencies](#)” on page 319
- “[Chapter 8, Start Volume Group Export](#)” on page 326
- “[Chapter 8, Start Volume Group Import](#)” on page 327

Performance Tuning

Over time, as a storage array exchanges data between the hosts and the drives, its performance can degrade. You can monitor the performance of a storage array and make adjustments to the operational settings on the storage array to help improve performance.

Monitoring the Performance

You can monitor the performance of a storage array by using the `save storageArray performanceStats` command. This command saves performance information to a file that you can review to help determine how well the storage array is running. [Table 21](#) lists the performance information that is saved to the file.

Table 21 Information About Storage Array Performance

Type of Information	Description
Devices	These devices are included in the file: <ul style="list-style-type: none"> • Controllers – The controller in slot A or slot B and a list of the volumes that are owned by the controller • Volumes – A list of the volume names • Storage array totals – A list of the totals for both controllers in an active/active controller pair, regardless if one, both, or neither are selected for monitoring
Total I/Os	The number of total I/Os performed since the storage array was started
Read Percentage	The percentage of total I/Os that are read operations (calculate the write percentage by subtracting the read percentage from 100 percent)

Type of Information	Description
Cache Hit Percentage	The percentage of reads that are fulfilled by data from the cache rather than requiring an actual read from a drive
Current KB per second	The current transfer rate in kilobytes per second (current means that the number of kilobytes per second since the last time that the polling interval elapsed, causing an update to occur)
Maximum KB per second	The highest data transfer value that is achieved in the current kilobyte-per-second statistic block
Current I/O per second (IOPS)	The current number of I/Os per second (current means the number of I/Os per second since the last time that the polling interval elapsed, causing an update to occur)
Maximum I/O per second	The highest number of I/Os achieved in the current I/O-per-second statistic block

The command takes this form:

```
save storageArray performanceStats file="filename"
```

where *filename* is the name of the file in which you want to save the performance statistics. You can use any file name that your operating system can support. The default file type is `.csv`. The performance information is saved as a comma-delimited file.

Before you use the `save storageArray performanceStats` command, run the `set session performanceMonitorInterval` command and the `set session performanceMonitorIterations` command to specify how often statistics are collected.

Changing the RAID Levels

When you create a volume group, you can define the RAID level for the volumes in that volume group. You can change the RAID level later to improve performance or provide more secure protection for your data.

To change the RAID level, use this command:

```
set volumeGroup [volumeGroupName]  
raidLevel=(0 | 1 | 3 | 5 | 6)
```

where *volumeGroupName* is the number of the volume group for which you want to change the RAID level.

Changing the Segment Size

When you create a new volume, you can define the segment size for that volume. In addition, you can change the segment size later to optimize performance. In a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are needed to satisfy an I/O request. Use larger values for the segment size. Using a single drive for a single request leaves other drives available to simultaneously service other requests. If the volume is in a single-user large I/O environment, performance is maximized when a single I/O request is serviced with a single data stripe; use smaller values for the segment size. To change the segment size, use this command:

```
set volume ([volumeName] | <wwID>) segmentSize=segmentSizeValue
```

where *segmentSizeValue* is the new segment size that you want to set. Valid segment size values are **8**, **16**, **32**, **64**, **128**, **256**, and **512**. You can identify the volume by name or by WWID.

Changing the Cache Parameters

The script command `set` provides two commands that you can use to change cache parameter settings:

- `set storageArray`
- `set volume`

The `set storageArray` command lets you change settings for these items:

- The cache block size
- The cache flush start percentage
- The cache flush stop percentage

The `set volume` command lets you change settings for these items:

- The cache flush modifier
- The cache without batteries enabled or disabled
- The mirror cache enabled or disabled
- The read cache enabled or disabled
- The write cache enabled or disabled
- The read ahead multiplier
- The redundancy check enabled or disabled

Defragmenting a Volume Group

When you defragment a volume group, you consolidate the free capacity in the volume group into one contiguous area. Defragmentation does not change the way in which the data is stored on the volumes. As an example, consider a volume group with five volumes. If you delete volume 1 and volume 3, your volume group is configured as follows:

space, volume 2, space, volume 4, volume 5, original unused space

When you defragment this volume group, the space (free capacity) is consolidated into one contiguous location after the volumes. After being defragmented, the volume group appears as follows:

volume 2, volume 4, volume 5, consolidated unused space

To defragment a volume group, use this command:

```
start volumeGroup [volumeGroupName] defragment
```

where *volumeGroupName* is the identifier for the volume group.

Troubleshooting and Diagnostics

If a storage array exhibits abnormal operation or failures, you can use the commands that are described in this section to help determine the cause of the problem.

Collecting All Support Data

To gather the most comprehensive information about a storage array, run the `save storageArray supportData` command. This command collects data for remote troubleshooting and analysis of problems with the storage management software. All of the files gathered are compressed into a single archive in a zipped file format. [Table 22](#) lists the type of support data that is collected.

Table 22 Support Data for the Storage Array

Type of Data	Description and File Name
Storage array profile	A list of all components and properties of a storage array. storageArrayProfile.txt
Major Event Log	A detailed list of errors that occur on the storage array. The list is stored in reserved areas on the drives in the storage array. The list records configuration events and failures with storage array components. majorEventLog.txt
Read link status	A detailed list of errors that have been detected in the traffic flow between the devices on the Fibre Channel loop. A file of historical read link status data might also be included in the archive. readLinkStatus.csv
Switch-on-a-chip (SOC) error statistics	Information from the loop-switch ports that are connected to Fibre Channel devices. socStatistics.csv
NVSRAM	A controller file that specifies the default settings for the controllers. NVSRAMdata.txt

Type of Data	Description and File Name
Performance statistics	A detailed description of how a storage array is performing. Collected data includes the I/O activity of specific controllers or volumes, the transfer rate of the controller, the current I/Os per second, and the maximum I/Os per second. performanceStatistics.csv
Persistent reservations and persistent registrations	A detailed list of volumes on the storage array and persistent reservations and persistent registrations. persistentRegistrations.txt
Object bundle	A detailed description of the status of the storage array and its components, which was valid at the time that the file was generated. The object bundle file is a binary file and does not contain human-readable information. objectBundle
Drive diagnostic data	A detailed list of log sense data from all of the drives in the storage array. driveDiagnosticData.txt
Recovery profile	A detailed description of the latest recovery profile record and historical data. recoveryProfile.csv
Unreadable sectors	A detailed list of all of the unreadable sectors that have been logged to the storage array. badBlocksData.txt
State capture data	A detailed description of the current state of the storage array. stateCaptureData.dmp
Storage array	A detailed listing of the hardware components and the software components that comprise the storage array configuration. storageArrayConfiguration.cfg

Collecting Drive Data

To gather information about all of the drives in a storage array, use the `save allDrives` command. This command collects sense data and saves the data to a file. The sense data consists of statistical information that is maintained by each of the drives in the storage array.

Diagnosing a Controller

The `diagnose controller` command provides these tests that help you make sure that a controller is functioning correctly:

- The read test
- The write test
- The data-loopback test

The read test initiates a read command as it would be sent over an I/O data path. The read test compares data with a known, specific data pattern, and the read test checks for data integrity and errors. If the read command is unsuccessful or the data compared is not correct, the controller is considered to be in error and is placed offline.

The write test initiates a write command as it would be sent over an I/O data path to the diagnostics region on a specified drive. This diagnostics region is then read and compared to a specific data pattern. If the write fails or the data compared is not correct, the controller is considered to be in error, and it is failed and placed offline.

Run the data-loopback test only on controllers that have connections between the controller and the drives. The test passes data through each controller drive-side channel, the minihub, out onto the loop, and back again. Enough data is transferred to determine error conditions on the channel. If the test fails on any channel, this status is saved so that it can be returned if all of the other tests pass.

For best results, run all three tests after you first install the storage array and any time that you that have made changes to the storage array or the components that are connected to the storage array (such as hubs, switches, and host adapters).

The test results contain a generic, overall status message and a set of specific test results. Each test result contains these items:

- Test (read, write, or data loopback)
- Port (read or write)
- Level (internal or external)
- Status (pass or fail)

Events are written to the Event Log when the diagnostics are started and when testing is completed. These events help you to evaluate whether diagnostics testing was successful or failed and the reason for the failure.

Running Read Link Status Diagnostics

Read link status (RLS) error counts refer to link errors that have been detected in the traffic flow of a Fibre Channel loop. The errors detected are represented as a count (32-bit field) of error occurrences that are accumulated over time. The counts provide coarse measure of the integrity of the components and devices on the loop. By analyzing the error counts that are retrieved, you can determine the components or devices within the Fibre Channel loop that might be experiencing problems communicating with the other devices on the loop. A high error count for a particular component or device indicates that it might be experiencing problems and should be given immediate attention.

Error counts are calculated from the current baseline. The baseline describes the error count values for each type of device in the Fibre Channel loop, either when the controller goes through its start-of-day sequence or when you reset the baseline. The baseline indicates the difference in error counts from the time the baseline was established to the time you request the read link status data. The script command set provides two commands for running RLS diagnostics:

- `reset storageArray RLSBaseline` – Resets the RLS baseline for all devices by setting all of the counts to 0.
- `save storageArray RLSCounts` – Saves the RLS counters to a file that you can review later. The default file name is `readLinkStatus.csv`.

Run the `reset storageArray RLSBaseline` command before you run the `save storageArray RLSBaseline` command.

[Table 23](#) lists the type of data contained by the file that is generated by the `save storageArray RLSBaseline` command.

Table 23 RLS Baseline Data for the Storage Array

Type of Data	Description
Devices	<p>A list of all devices on the Fibre Channel loop.</p> <p>The devices appear in channel order. Within each channel, the devices are sorted according to the device position in the loop.</p>
Baseline time	<p>The date and time when the baseline was set.</p>
Elapsed time	<p>The time that has elapsed from when the baseline time was set to when the read link status was gathered.</p>
Invalid transmission word (ITW)	<p>The total number of ITW errors that were detected on the Fibre Channel loop from the baseline time to the current date and time. ITW might also be referred to as the Received Bad Character Count.</p> <p>ITW counts indicate that in decoding a read/write transmission, the mapping did not exist and the running disparity of the transmission word is invalid. This data is the key error count to be used when analyzing the error count data.</p>
Link failure (LF)	<p>The total number of LF errors that were detected on the Fibre Channel loop from the baseline time to the current date and time.</p> <p>An LF condition is either a link fault signal, a loss of signal, or a loss of synchronization condition. The LF signal indicates a failure with the media module laser operation.</p>
Loss of synchronization (LOS)	<p>The total number of LOS errors that were detected on the Fibre Channel loop from the baseline time to the current date and time.</p> <p>LOS errors indicate that the receiver cannot acquire symbol lock with the incoming data stream due to a degraded input signal. If this condition persists, the number of LOS errors increases.</p>

Type of Data	Description
Loss of signal (LOSG)	<p>The total number of LOSG errors that were detected on the Fibre Channel loop from the baseline date to the current date and time.</p> <p>LOSG errors typically indicate a loss of signal from the transmitting node or the physical component within the Fibre Channel loop. Physical components where a loss of signal typically occurs include the gigabit interface converters (GBICs), the Small Form-factor Pluggable (SFP) transceivers, and the Fibre Channel fiber-optic cable.</p>
Primitive sequence protocol (PSP)	<p>The total number of PSP errors that were detected on the Fibre Channel loop from the baseline date to the current date and time. PSP refers to the number of NPort protocol errors that were detected and Link Reset Response (LRR) primitive sequences that were received while the link is up. An LRR is issued by another NPort in response to a link reset.</p> <p>An NPort is a Fibre Channel-defined port at the end of a link, such as a server or a workstation. Each port can act as an originator or a responder (or both) and contains a transmitter and receiver. Each port is given a unique name, called an NPort or an NLPort identifier. If an NPort is connected to a loop, it becomes an NLPort. An NLPort is a Fibre Channel controller ID in a hexadecimal number. The hexadecimal number varies depending on the topology:</p> <ul style="list-style-type: none"> • For a private arbitrated loop, the ID is a 1-byte arbitrated loop physical address (ALPA). • For all other arbitrated loops, it appears as a single 24-bit hexadecimal number (a triplet of domain, area, and ALPA where each field is 1 byte). • For fabric and point-to-point, the ID is a 3-byte hexadecimal number used in the DID and SID (destination identifier and source identifier) fields of Fibre Channel frames.

Type of Data	Description
Invalid cyclic redundancy check (ICRC)	<p>The total number of ICRC errors that were detected on the Fibre Channel loop from the baseline date to the current date and time.</p> <p>An ICRC count indicates that a frame has been received with an invalid cyclic redundancy check value. A cyclic redundancy check reads the data, calculates the cyclic redundancy check character, and compares the calculated cyclic redundancy check character with a cyclic check character already present in the data. If they are equal, the new data is presumed to be the same as the old data. If the calculated characters and the old characters do not match, an error is posted, and the data is re-sent.</p>

Interpreting the RLS Results

The way that you interpret the RLS results is based on the concept that the device immediately following the problematic component will have the largest number of invalid transition word (ITW) error counts. The process is to obtain the ITW count for every component and device on the loop, analyze the data in loop order, and identify any large increases in the ITW counts.

IMPORTANT The current error counting standard for when to calculate the ITW error count is not well defined. Different vendor devices calculate at different rates. Analysis of the data must take this discrepancy into consideration.

Collecting Switch-on-a-Chip Error Statistics

Switch-on-a-chip (SOC) error statistics provide information about the loop-switch ports that are connected to the Fibre Channel devices in a storage array. (RLS counts provide information about the Fibre Channel devices.) Reporting SOC error statistics is available only on storage arrays that have SOC loop-switch devices that are incorporated into the controller drive channel or the ESM circuitry. SOC devices are integrated circuits that join together Fibre Channel devices in arbitrated loop topologies. SOC devices automatically collect statistical information for each SOC port that is connected to a controller port, an ESM port, a drive port, or an expansion connector. A Customer and Technical Support representative can use the statistical information with RLS counts to identify problems with Fibre Channel devices that are attached to the loop.

SOC error statistics include this information:

- The port state
- The port insertion count
- The loop state
- The loop up count
- The CRC error count
- The relative frequency drift error average
- The loop cycle count
- The operating system (OS) error count
- The port connections attempted count
- The port connections held off count
- The port utilization

The method for collecting error statistics starts by establishing a baseline for the SOC error statistics. The baseline consists of SOC error statistics that are established at a set time for each SOC device on the loop. The baseline is set by clearing the error counters in each SOC device. You can set a device baseline by performing one of these actions:

- Turning on the power to the device or resetting the device
- Running the `reset storageArray SOCBaseline` command

In addition, each controller also initializes the SOC error counters in all of the drive trays that are attached to the controller following a cold boot (power-on or hot insertion). If you add a drive tray while the power is turned on to the storage array, a new baseline is established for any device on the drive tray.

After you have established the baseline for the SOC devices, you run the storage array for a predetermined amount of time (for example, two hours). At the end of the run time, you collect the SOC error statistics by saving the information to a file. To save the information, run the `save storageArray SOCCounts file filename` command. The default name of the file that contains the SOC error statistics is `socStatistics.csv`. You can use any file name that has the `.csv` extension.

Analyzing the SOC error statistics is beyond the scope of normal storage array management. After you have collected the SOC error statistics in a file, send the file to a Customer and Technical Support representative.

Recovery Operations

Recovery operations include repairing the storage array and returning it to an operational state. This might involve replacing a failed canister, a failed controller, or a drive or restoring data or the storage array to operation. For information about when it is appropriate to replace a canister, see “[Chapter 7, Routine Maintenance](#)” on page 117.

Setting the Controller Operational Mode

A controller has three operational modes:

- Online
- Offline
- Service

Placing a controller online sets it to the Optimal state and makes it active and available for I/O operations. Placing a controller offline makes it unavailable for I/O operations and moves its volume groups to the other controller if failover protection is enabled.

Taking a controller offline can seriously impact data integrity and storage array operation.

- If you do not use write cache mirroring, data in the cache of the controller you place offline is lost.
- If you take a controller offline and you have controller failover protection through a host multi-path driver, the other controller in the pair takes over. Volume groups and their associated volumes that were assigned to the offline controller are automatically reassigned to the remaining controller. If you do not have a multi-path driver installed on the application host and you take a controller offline while the application is using associated volumes, application errors will occur.

ATTENTION Possible loss of data access – Placing a controller offline can cause loss of data.

Use Service mode to replace canisters, such as a controller. Placing a controller in Service mode makes it unavailable for I/O operations and moves its volume groups to the second controller without affecting the preferred path of the volume group. This action might significantly reduce performance. The volume groups are automatically transferred back to the preferred controller when it is placed back online.

If you change a controller to Service mode while an application is using the associated volumes on the controller, the change causes I/O errors unless a multi-path driver is installed on the host. Before you place a controller in Service mode, make sure that the volumes are not in use, or a multi-path driver is installed on all of the hosts that are using these volumes.

In addition, if you do not have a multi-path driver, you must make appropriate operating system-specific modifications to make sure that the volume groups moved are accessed on the new path when you change to Service mode.

IMPORTANT Place a controller in Service mode only under the direction of a Customer and Technical Support representative.

To change the operational mode of a controller, use this command:

```
set controller [(a | b)] availability=(online | offline | serviceMode)
```

Changing the Controller Ownership

You can change which controller is the owner of a volume by using the `set volume` command. The command takes this form:

```
set (allVolumes | volume [volumeName] | volumes [volumeName1 ... volumeNameN]  
| volume <wwID>) owner=(a | b)
```

Initializing a Drive

ATTENTION **Possible loss of data access** – When you initialize a drive, all data on the drive is lost.

You must initialize a drive when you have moved a drive that was previously part of a multidisk volume group from one storage array to another. If you do not move the entire set of drives, the volume group information and the volume information on the drives that you move are incomplete. Each drive that you move contains only part of the information that is defined for the volume and the volume group. To be able to reuse the drives to create a new volume group and volume, you must delete all of the old information from the drives by initializing the drive.

When you initialize a drive, all of the old volume group information and volume information are deleted, and the drive is returned to an unassigned state. Returning a drive to an unassigned state adds unconfigured capacity to a storage array. You can use this capacity to create additional volume groups and volumes.

To initialize a drive, use this command:

```
start drive [trayID,slotID] initialize
```

where *trayID* and *slotID* are the identifiers for the drive.

Reconstructing a Drive

If two or more of the drives in a volume group have failed, the volume shows a status of Failed. All of the volumes in the volume group are no longer operating. To return the volume group to an Optimal status, you must replace the failed drives. Then, you must reconstruct the data on the new drives. The data that you reconstruct is the data as it would appear on the failed drives.

IMPORTANT You can use this command only when the drive is assigned to a RAID 1, RAID 3, RAID 5, or RAID 6 volume group.

To reconstruct a drive, use this command:

```
start drive [trayID,slotID] reconstruct
```

where *trayID* and *slotID* are the identifiers for the drive.

Initializing a Volume

ATTENTION Possible loss of data access – When you initialize a volume, all data on the volume and all of the information about the volume are destroyed.

A volume is automatically initialized when you first create it. If the volume starts showing failures, you might be required to re-initialize the volume to correct the failure condition.

Consider these restrictions when you initialize a volume:

- You cannot cancel the operation after it begins.
- You cannot use this option if any modification operations are in progress on the volume or the volume group.
- You cannot change the cache parameters of the volume while the initialization operation is in progress.

To initialize a volume, use this command:

```
start volume [volumeName] initialize
```

where *volumeName* is the identifier for the volume.

Redistributing Volumes

When you redistribute volumes, you return the volumes to their preferred controller owners. The preferred controller ownership of a volume or a volume group is the controller of an active-active pair that is designated to own the volumes. The preferred owner for a volume is initially designated when the volume is created. If the preferred controller is being replaced or undergoing a firmware download, ownership of the volumes is automatically shifted to the other controller. That controller becomes the current owner of the volumes. This change is considered to be a routine ownership change and is reported in the Event Log.

To redistribute volumes to their preferred controllers, use this command:

```
reset storageArray volumeDistribution
```

IMPORTANT If you run this command without a multi-path driver on the hosts, stop I/O activity to the volumes to prevent application errors.

IMPORTANT You cannot run this command if all of the volumes are currently owned by their preferred controller, or the storage array does not have defined volumes.

Under some host operating systems, you must reconfigure the multi-path host driver. You might also need to make operating system modifications to recognize the new I/O path to the volume.

Replacing Canisters

ATTENTION **Possible loss of data access** – Never remove a component that has a Service Action Required indicator light on unless the Service Action Allowed indicator light is on.

If a component fails and must be replaced, the Service Action Required indicator light on that canister comes on to indicate that service action is required, provided no data availability dependencies or other conditions exist that dictate the canister should not be removed. The Service Action Allowed indicator light automatically comes on or goes off when conditions change. In most cases, the Service Action Allowed indicator light comes on steadily when the Service Action Required indicator light comes on for the canister.

If the interconnect-battery canister must be replaced, the Service Action Allowed indicator light does not come on automatically. Before the Service Action Allowed indicator light on the interconnect-battery canister can come on, you must place controller canister in slot B into Service mode. This action routes all control and I/O activity through one controller to help make sure that data access is maintained while the interconnect-battery canister is removed. The Service Action Allowed indicator light comes on after the new canister has been installed.

The ability to remove a canister depends on the data availability dependencies of the controller tray or the controller-drive tray. The Service Action Allowed indicator light does not come on if removing a canister jeopardizes data on the drive trays or current I/O activity. An example of limiting when you can remove a canister is when one controller canister has a Service Action Required indicator light on. You cannot remove the other controller canister (the Service Action Allowed indicator light does not come on), because doing so would jeopardize the data either on the drive trays or transitioning through the controllers.

A less obvious example is when the power supply for the controller canister in slot A has failed, and the controller canister in slot B has failed. Removing the controller canister in slot B before replacing the failed power-fan canister causes the controller canister in slot A to lose power, which results in a loss of data access. This action occurs because power distribution from each power-fan canister is through the controller canister that is physically connected to that power-fan canister.

So, in the preceding example, these actions occur:

- The power-fan canister has both its Service Action Required indicator light and its Service Action Allowed indicator light on.
- The controller canister in slot B has only its Service Action Required indicator light on, but its Service Action Allowed indicator light is off.
- After the failed power-fan canister has been replaced, the Service Action Allowed indicator light comes on for the controller canister in slot B.

Table 24 shows when the Service Action Allowed indicator light does not come on (the indicator light is suppressed) for each canister. An X in a table cell indicates that service is not allowed, therefore the Service Action Allowed light does not come on. For example, if the power supply in the power-fan canister in slot A has failed, then replacement of the controller canister in slot B, the interconnect-battery canister, and the power-fan canister in slot B is not allowed, which is indicated when the Service Action Allowed indicator light stays off for those canisters.

Table 24 Service Action Not Allowed

Description of Failure or Circumstance	Canister Description				
	Controller in Slot A	Controller in Slot B	Interconnect Battery	Power-Fan in Slot A	Power-Fan in Slot B
The controller canister in slot A has failed or is locked down		X	X		
The controller canister in slot B has failed or is locked down	X				
The controller canister in slot A drive path is unavailable		X	X		
The controller canister in slot B drive path is unavailable	X		X		

Description of Failure or Circumstance	Canister Description				
	Controller in Slot A	Controller in Slot B	Interconnect Battery	Power-Fan in Slot A	Power-Fan in Slot B
The power supply in the power-fan canister in slot A has failed		X	X		X
A fan in the power-fan canister in slot A has failed					
The power supply in the power-fan canister in slot B has failed	X		X	X	
A fan in the power-fan canister in slot B has failed					
The interconnect-battery canister has been removed	X			X	
The controller canister in slot A has been removed		X	X		X
The controller canister in slot B has been removed	X			X	
The power-fan canister in slot A has been removed		X	X		X
The power-fan canister in slot B has been removed	X		X	X	
The battery pack has failed					
The battery pack has been removed					

Script Commands

ATTENTION The script commands are capable of damaging a configuration and causing loss of data access if not used correctly – Command operations are performed as soon as you run the commands. Some commands can immediately delete configurations or data. Before using the script commands, make sure that you have backed up all data, and have saved the current configuration so that you can reinstall it if the changes you make do not work.

This chapter has five sections to help you use script commands:

- “[Naming Conventions](#)” on page 146 lists the general formatting rules for entering the names of storage array entities, such as volumes or drives, with the script commands.
- “[Firmware Compatibility Levels](#)” on page 148 describes how to interpret the firmware level information.
- “[Formatting Rules for Script Commands](#)” on page 149 lists the general formatting rules that apply to the script command syntax.
- “[Commands Listed by Function](#)” on page 151 lists the script commands organized into groups related to the physical features, the logical features, and the operational features of the storage array.
- “[Commands Listed Alphabetically](#)” on page 159 lists the script commands alphabetically and, for each script command, includes script command name, syntax, and parameters.

IMPORTANT Terminology differences – The names of components and features change from time to time; however, the command syntax does not change at the same time. You will notice minor differences between the terminology used to describe components and features and the terminology used in the syntax to describe those same items when used in a command name, a parameter, or a variable.

Naming Conventions

- Names can have a maximum of 30 characters.
- You can use any combination of alphanumeric characters, hyphens, and underscores for the names of the following components:
 - Storage arrays
 - Host groups
 - Hosts
 - Volume groups
 - Volumes
 - HBA host ports
- You must use unique names. If you do not use unique names, the controller firmware returns an error.
- If the name contains more than one word, hyphens, or underscores, enclose the name in double quotation marks (“ ”). In some usages, you must also surround the name with square brackets ([]). The description of each parameter indicates whether you need to enclose a parameter in double quotation marks, square brackets, or both.
- The name character string cannot contain a new line.
- On Windows operating systems, you must enclose the name between two back slashes (\\) in addition to other delimiters. For example, the following name is used in a command that runs under a Windows operating system:

```
[\\“Engineering\\”]
```

- For a UNIX operating system and, when used in a script file, the name appears as in the following example:

```
[“Engineering”]
```

- When you enter a World Wide Identifier (WWID) of an HBA host port, some usages require that you surround the WWID with double quotation marks. In other uses, you must surround the WWID with angle brackets (<>). The description of the WWID parameter indicates whether you need to enclose the WWID in double quotation marks or angle brackets.

Entering Numerical Names

When the storage management software automatically configures a storage array, the storage management software assigns names that consist of numerical characters. Names that consist only of numerical characters are valid names. Numerical character names, however, must be treated differently than names that start with alphabetic characters.

When you enter a script command that requires a name, the script engine looks for a name that starts with an alphabetic character. The script engine might not recognize the following names:

- Names that are only numbers, such as 1 or 2
- Names that start with a number, such as 1Disk or 32Volume

To enter a name that consists only of numerical characters so that the script engine will recognize the name, use a combination of back slashes and double quotation marks. The following are examples of how you can enter names that consist only of numerical characters or start with numerical characters:

- [`\“1\”`]
- [`\“1Disk\”`]

Firmware Compatibility Levels

The script commands and the command parameters do not run under all versions of the controller firmware. The script commands in the following sections list the minimum firmware levels under which the script commands can run. In the script commands, the firmware levels are listed under the heading “Minimum Firmware Level.” This list describes how to interpret the information about the firmware levels.

- If a script command does not list a minimum controller firmware level, the script command and all of the parameters associated with that script command can run under any level of controller firmware.
- A controller firmware number without any explanatory information indicates that the controller firmware level applies to the entire script command and all of the parameters for that script command.
- A controller firmware number that is associated with a parameter indicates the minimum controller firmware level under which the parameter can run.

NOTE The minimum controller firmware level indicates support by the ISSM-EE software that releases the command, as well as support by all storage management software that picks up usage. CLI support capabilities depend on the hardware used. When an unsupported command is entered, an error message appears.

Examples of Firmware Compatibility Levels

The `create hostGroup` command has the following section.

Minimum Firmware Level

5.20

This level indicates that the entire script command runs under a minimum of controller firmware version 5.20.

The `show volume` command has the following section.

Minimum Firmware Level

5.00

5.43 adds the `summary` parameter

These notations indicate that the script command and all of the parameters except `summary` run under a minimum of controller firmware version 5.00. The `summary` parameter runs under a minimum of controller firmware version 5.43.

Formatting Rules for Script Commands

Syntax unique to a specific script command is explained in the Notes section at the end of each script command description.

Case sensitivity – The script commands are not case sensitive. You can type the script commands in lowercase, uppercase, or mixed case. (In the following command descriptions, mixed case is used as an aid to reading the command names and understanding the purpose of the command.)

Spaces – You must enter spaces in the script commands as they are shown in the command descriptions.

Square brackets – Square brackets are used in two ways:

- As part of the command syntax.
- To indicate that the parameters are optional. The description of each parameter tells you if you need to enclose a parameter value in square brackets.

Parentheses – Parentheses shown in the command syntax enclose specific choices for a parameter. That is, if you want to use the parameter, you must enter one of the values enclosed in parentheses. Generally, you do not include parentheses in a script command; however, in some instances, when you enter lists, you must enclose the list in parentheses. Such a list might be a list of tray ID values and slot ID values. The description of each parameter tells you if you need to enclose a parameter value in parentheses.

Vertical bars – Vertical bars in a script command indicate “or” and separate the valid values for the parameter. For example, the syntax for the `raidLevel` parameter in the command description appears as follows:

```
raidLevel=(0 | 1 | 3 | 5 | 6)
```

To use the `raidLevel` parameter to set a RAID level of 5, enter this value:

```
raidLevel=5
```

Drive locations – When you specify drive locations by using tray ID values and slot ID values, separate the ID values with a comma. If you enter more than one set of ID values, separate each set of values with a space. Enclose the set of values in parentheses. For example:

```
(1,1 1,2 1,3 1,4 2,1 2,2 2,3 2,4)
```

Italicized terms – Italicized terms in the command indicate a value or information that you need to provide. For example, when you encounter the italicized term:

```
numberOfDrives
```

replace the italicized term with a value for the number of drives that you want to include with the script command.

Semicolon – Script commands must end with a semicolon (;). You can enter more than one script command on the command line or in a script file. For example, a semicolon is used to separate each script command in the following script file.

```
create volume drives=(0,2 0,3 1,4 1,5 2,6 2,7) raidLevel=5 userLabel="v1"  
capacity=2 gb owner=a;  
create volume volumeGroup=2 userLabel="v2" capacity=1 gb owner=b;  
create volume volumeGroup=2 userLabel="v3" capacity=1 gb owner=a;  
  
create volume drives=(0,4 0,5 1,6 1,7 2,8 2,9) raidLevel=5 userLabel="v4"  
capacity=2 gb owner=b;  
create volume volumeGroup=3 userLabel="v5" capacity=1 gb owner=a;  
create volume volumeGroup=3 userLabel="v6" capacity=1 gb owner=b;
```

Commands Listed by Function

Controller Commands

- “Clear Drive Channel Statistics” on page 169
- “Diagnose Controller” on page 206
- “Enable Controller” on page 215
- “Reset Controller” on page 232
- “Save Controller NVSRAM” on page 240
- “Set Controller” on page 251
- “Set Controller Service Action Allowed Indicator” on page 254
- “Set Drive Channel Status” on page 255
- “Set Host Channel” on page 263
- “Show Controller” on page 292
- “Show Controller NVSRAM” on page 296
- “Show Drive Channel Stats” on page 297
- “Start Drive Channel Locate” on page 322
- “Stop Drive Channel Locate” on page 330

Drive Commands

[“Download Drive Firmware”](#) on page 209

[“Replace Drive”](#) on page 231

[“Revive Drive”](#) on page 238

[“Save Drive Channel Fault Isolation Diagnostic Status”](#) on page 241

[“Save Drive Log”](#) on page 242

[“Set Drive Hot Spare”](#) on page 256

[“Set Drive Service Action Allowed Indicator”](#) on page [Set Drive Service Action Allowed Indicator](#)

[“Set Drive State”](#) on page 258

[“Set Foreign Drive to Native”](#) on page 258

[“Show Drive”](#) on page 354

[“Show Drive Download Progress”](#) on page 298

[“Start Drive Initialize”](#) on page 323

[“Start Drive Locate”](#) on page 324

[“Start Drive Reconstruction”](#) on page 324

[“Stop Drive Locate”](#) on page 330

Host Topology Commands

- “[Create Host](#)” on page 174
- “[Create Host Group](#)” on page 175
- “[Create Host Port](#)” on page 176
- “[Delete Host](#)” on page 202
- “[Delete Host Group](#)” on page 203
- “[Delete Host Port](#)” on page 204
- “[Set Host](#)” on page 259
- “[Set Host Group](#)” on page 264
- “[Set Host Port](#)” on page 265
- “[Show Host Ports](#)” on page 299

Remote Volume Mirroring Commands

- “[Activate Remote Volume Mirroring Feature](#)” on page 159
- “[Check Remote Mirror Status](#)” on page 166
- “[Create Remote Mirror](#)” on page 188
- “[Deactivate Remote Mirror](#)” on page 201
- “[Diagnose Remote Mirror](#)” on page 208
- “[Re-create Remote Volume Mirroring Repository Volume](#)” on page 221
- “[Remove Remote Mirror](#)” on page 226
- “[Resume Remote Mirror](#)” on page 237
- “[Set Remote Mirror](#)” on page 266
- “[Show Remote Volume Mirroring Volume Candidates](#)” on page 300
- “[Show Remote Volume Mirroring Volume Synchronization Progress](#)” on page 301

[“Start Remote Volume Mirroring Synchronization”](#) on page 325

[“Suspend Remote Mirror”](#) on page 333

Session Command

[“Set Session”](#) on page 268

Snapshot Commands

[“Create Snapshot Volume”](#) on page 192

[“Re-create Snapshot”](#) on page 224

[“Re-create Snapshot Collection”](#) on page 225

[“Set Snapshot Volume”](#) on page 271

[“Stop Snapshot”](#) on page 331

Storage Array Commands

[“Activate Storage Array Firmware”](#) on page 162

[“Autoconfigure Storage Array”](#) on page 162

[“Autoconfigure Storage Array Hot Spares”](#) on page 166

[“Clear Storage Array Configuration”](#) on page 169

[“Clear Storage Array Event Log”](#) on page 171

[“Clear Storage Array Firmware Pending Area”](#) on page 171

[“Disable Storage Array Feature”](#) on page 209

[“Download Storage Array Drive Firmware”](#) on page 212

[“Download Storage Array Firmware/NVSRAM”](#) on page 213

[“Download Storage Array NVSRAM”](#) on page 214

[“Enable Storage Array Feature Key”](#) on page 216

- “Reset Storage Array Battery Install Date” on page 233
- “Reset Storage Array Diagnostic Data” on page 234
- “Reset Storage Array RLS Baseline” on page 235
- “Reset Storage Array SOC Baseline” on page 235
- “Reset Storage Array Volume Distribution” on page 236
- “Save Storage Array Configuration” on page 243
- “Save Storage Array Diagnostic Data” on page 245
- “Save Storage Array Events” on page 246
- “Save Storage Array Performance Statistics” on page 247
- “Save Storage Array RLS Counts” on page 248
- “Save Storage Array SOC Counts” on page 249
- “Save Storage Array State Capture” on page 250
- “Save Storage Array Support Data” on page 250
- “Set Storage Array” on page 272
- “Set Storage Array Redundancy Mode” on page 276
- “Set Storage Array Time” on page 276
- “Set Storage Array Tray Positions” on page 276
- “Show Storage Array” on page 302
- “Show Storage Array Auto Configure” on page 306
- “Show Storage Array Host Topology” on page 309
- “Show Storage Array LUN Mappings” on page 311
- “Show Storage Array Unreadable Sectors” on page 312
- “Start Storage Array Locate” on page 325
- “Stop Storage Array Drive Firmware Download” on page 331

[“Stop Storage Array Locate”](#) on page 332

Tray Commands

[“Download Environmental Card Firmware”](#) on page 211

[“Set Tray Alarm”](#) on page 278

[“Set Tray Identification”](#) on page 279

[“Set Tray Service Action Allowed Indicator”](#) on page 280

[“Start Tray Locate”](#) on page 326

[“Stop Tray Locate”](#) on page 332

Uncategorized Commands

[“Show String”](#) on page 313

Volume Commands

[“Check Volume Parity”](#) on page 167

[“Clear Volume Reservations”](#) on page 172

[“Clear Volume Unreadable Sectors”](#) on page 173

[“Create RAID Volume \(Automatic Drive Select\)”](#) on page 177

[“Create RAID Volume \(Free Extent Based Select\)”](#) on page 180

[“Create RAID Volume \(Manual Drive Select\)”](#) on page 184

[“Delete Volume”](#) on page 205

[“Recover RAID Volume”](#) on page 218

[“Remove Volume LUN Mapping”](#) on page 228

[“Repair Volume Parity”](#) on page 230

[“Show Volumes”](#) on page 362

[“Show Volume Action Progress”](#) on page 314

[“Show Volume Performance Statistics”](#) on page 321

[“Show Volume Reservations”](#) on page 322

[“Start Volume Initialization”](#) on page 329

Volume Copy Commands

[“Create Volume Copy”](#) on page 197

[“Recopy Volume Copy”](#) on page 217

[“Remove Volume Copy”](#) on page 227

[“Set Volume Copy”](#) on page 289

[“Show Volume Copy”](#) on page 315

[“Show Volume Copy Source Candidates”](#) on page 316

[“Show Volume Copy Target Candidates”](#) on page 317

[“Stop Volume Copy”](#) on page 332

Volume Group Commands

[“Create Volume Group”](#) on page 198

[“Delete Volume Group”](#) on page 206

[“Revive Volume Group”](#) on page 239

[“Set Volume Group”](#) on page 290

[“Set Volume Group Forced State”](#) on page 291

[“Show Volume Group”](#) on page 317

[“Show Volume Group Export Dependencies”](#) on page 319

[“Show Volume Group Import Dependencies”](#) on page 320

[“Start Volume Group Defragment”](#) on page 326

[“Start Volume Group Export”](#) on page 327

[“Start Volume Group Export”](#) on page 328

[“Start Volume Group Locate”](#) on page 329

[“Stop Volume Group Locate”](#) on page 333

Commands Listed Alphabetically

Activate Remote Volume Mirroring Feature

This command creates the mirror repository volume and activates the Remote Volume Mirroring feature. When you use this command, you can define the mirror repository volume in one of three ways:

- User-defined drives
- User-defined volume group
- User-defined number of drives

If you choose to define a number of drives, the controller firmware chooses which drives to use for the mirror repository volume.

Syntax (User-Defined Drives)

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
[trayLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Volume Group)

```
activate storageArray feature=remoteMirror
repositoryVolumeGroup=volumeGroupName [freeCapacityArea=freeCapacityIndexNumber]
```

Syntax (User-Defined Number of Drives)

```
activate storageArray feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
[driveType=(fibre | SATA | SAS | PATA)]
[trayLossProtect=(TRUE | FALSE)]
```

Parameters

Parameter	Description
repositoryRAIDLevel	The RAID level for the mirror repository volume. Valid values are 1, 3, 5, or 6 .
repositoryDrives	The drives for the mirror repository volume. Specify the tray ID value and the slot ID value for each drive that you assign to the mirror repository volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
repositoryVolumeGroup	The name of the mirror repository volume group where the mirror repository volume is located. (To determine the names of the volume groups in your storage array, run the <code>show storageArray profile</code> command.)
freeCapacityArea	The index number of the free space in an existing volume group that you want to use to create the mirror repository volume. Free capacity is defined as the free capacity between existing volumes in a volume group. For example, a volume group might have the following areas: volume 1, free capacity, volume 2, free capacity, volume 3, free capacity. To use the free capacity following volume 2, you would specify: <code>freeCapacityArea=2</code> Run the <code>show volumeGroup</code> command to determine if a free capacity area exists.

Parameter	Description
repositoryDriveCount	The number of unassigned drives that you want to use for the mirror repository volume.
driveType	The type of drives that you want to use for the mirror repository volume. Valid drive types are fibre , SATA , SAS , or PATA . Use this parameter only when you use the repositoryDriveCount parameter. You must use this parameter only when you have more than one type of drive in your storage array.
trayLossProtect	The setting to enforce tray loss protection when you create the mirror repository volume. To enforce tray loss protection, set this parameter to TRUE . The default value is FALSE .

Notes

If the drives that you select for the repositoryDrives parameter are not compatible with other parameters (such as the repositoryRAIDLevel parameter), the script command returns an error, and Remote Volume Mirroring is not activated. The error returns the amount of space that is needed for the mirror repository volume. You can then re-enter the command, and specify the appropriate amount of space.

If you enter a value for the repository storage space that is too small for the mirror repository volumes, the controller firmware returns an error message that provides the amount of space that is needed for the mirror repository volumes. The command does not try to activate Remote Volume Mirroring. You can re-enter the command by using the value from the error message for the repository storage space value.

When you assign the drives, if you set the trayLossProtect parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the trayLossProtect parameter to **FALSE**, the storage array performs operations, but the volume group that you create might not have tray loss protection.

When the controller firmware assigns the drives, if you set the trayLossProtect parameter to **TRUE**, the storage array returns an error if the controller firmware cannot provide drives that result in the new volume group having tray loss protection. If you set the trayLossProtect parameter to **FALSE**, the storage array performs the operation even if it means that the volume group might not have tray loss protection.

Minimum Firmware Level

6.10

7.10 adds RAID 6 capability.

Activate Storage Array Firmware

This command activates firmware that you have previously downloaded to the pending configuration area on the controllers in the storage array.

Syntax

```
activate storageArray firmware
```

Parameters

None.

Minimum Firmware Level

6.10

Autoconfigure Storage Array

This command automatically configures a storage array. Before you enter the `autoConfigure storageArray` command, run the `show storageArray autoConfiguration` command. The `show storageArray autoConfiguration` command returns configuration information in the form of a list of valid drive types, RAID levels, volume information, and hot spare information. (This list corresponds to the parameters for the `autoConfigure storageArray` command.) The controllers audit the storage array and then determine the highest RAID level that the storage array can support and the most efficient volume definition for the RAID level. If the configuration that is described by the returned list is acceptable, you can enter the `autoConfigure storageArray` command without any parameters. If you want to modify the configuration, you can change the parameters to meet your configuration requirements. You can change a single parameter or all of the parameters. After you enter the `autoConfigure storageArray` command, the controllers set up the storage array by using either the default parameters or those you selected.

Syntax

```

autoConfigure storageArray
[driveType=(fibre | SATA | SAS | PATA)
raidLevel=(0 | 1 | 3 | 5 | 6)
volumeGroupWidth=numberOfDrives
volumeGroupCount=numberOfVolumeGroups
volumesPerGroupCount=numberOfVolumesPerGroup
hotSpareCount=numberOfHotSpares
segmentSize=segmentSizeValue
cacheReadPrefetch=(TRUE | FALSE)]

```

Parameters

Parameter	Description
driveType	The type of drives that you want to use for the storage array. Valid drive types are fibre , SATA , SAS , or PATA . The <code>driveType</code> parameter is not required if only one type of drive is in the storage array.
raidLevel	The RAID level of the volume group that contains the drives in the storage array. Valid RAID levels are 0 , 1 , 3 , 5 , or 6 .
volumeGroupWidth	The number of drives in a volume group in the storage array.
volumeGroupCount	The number of volume groups in the storage array. Use integer values.
volumesPerGroupCount	The number of equal-capacity volumes per volume group. Use integer values.

Parameter	Description
hotSpareCount	The number of hot spares that you want in the storage array. Use integer values.
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .

Notes

Drives

A volume group is a set of drives that are logically grouped together by the controllers in the storage array. The number of drives in a volume group is a limitation of the controller firmware. It is recommended that you adhere to the 30-drive limit and to the 2-TB size maximum for the volume group, which is imposed by some operating systems. This table relates drive capacity to the number of drives in a volume group.

Drive Size	Maximum Number of Drives in the Volume Group
18 GB	30 drives
36 GB	30 drives
73 GB	28 drives
180 GB	11 drives

A 73-GB drive or a 180-GB drive does not report exactly 73 GB or 180 GB as its capacity.

Hot Spares

Hot spare drives can replace any failed drive in the storage array. The hot spare must be the same type of drive as the drive that failed (that is, a SATA hot spare cannot replace a Fibre Channel drive). A hot spare must have capacity greater than or equal to any drive that can fail. If a hot spare is smaller than a failed drive, you cannot use the hot spare to rebuild the data from the failed drive. Hot spares are available only for RAID levels 1, 3, 5, or 6.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks. When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once. For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drive into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

Minimum Firmware Level

6.10

7.10 adds RAID 6 capability and removes hot spare limits.

Autoconfigure Storage Array Hot Spares

This command automatically defines and configures the hot spares in a storage array. You can run this command at any time. This command provides the best hot spare coverage for a storage array.

Syntax

```
autoConfigure storageArray hotSpares
```

Parameters

None.

Notes

When you run the `autoconfigure storageArray hotSpares` command, the controller firmware determines the number of hot spares to create based on the total number and type of drives in the storage array. For Fibre Channel drives, SATA drives, and SAS drives, the controller firmware creates one hot spare for the storage array and one additional hot spare for every 60 drives in the storage array. For PATA drives, the controller firmware creates one hot spare for the storage array and one additional hot spare for every 20 drives in the storage array.

Minimum Firmware Level

6.10

Check Remote Mirror Status

This command returns the status of a remote-mirror volume. Use this command to determine when the status of the remote-mirror volume becomes Optimal.

Syntax

```
check remoteMirror localVolume [volumeName] optimalStatus  
timeout=timeoutValue
```

Parameters

Parameter	Description
localVolume	The name of any remote-mirror volume. The remote-mirror volume can be the primary volume or the secondary volume of a remote-mirror pair. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").
timeout	The time interval within which the software can return the remote-mirror volume status. The timeout value is in minutes.

Notes

This command waits until the status becomes Optimal or the timeout interval expires. Use this command when you run the Asynchronous Remote Volume Mirroring utility.

Minimum Firmware Level

6.10

Check Volume Parity

This command checks a volume for parity and media errors and writes the results of the check to a file.

Syntax

```
check volume [volumeName]
parity [parityErrorFile=filename]
[mediaErrorFile=filename]
[priority=(highest | high | medium | low | lowest)]
[startingLBA=LBAvalue] [endingLBA=LBAvalue]
[verbose=(TRUE | FALSE)]
```

Parameters

Parameter	Description
volume	The name of the specific volume for which you want to check parity. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").
parityErrorFile	The name of the file in which you want to save the parity error information. Enclose the file name in double quotation marks (" ").
mediaErrorFile	The name of the file in which you want to save the media error information. Enclose the file name in double quotation marks (" ").
priority	The priority that the parity check has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .
startingLBA	The starting logical block address. Use integer values.
endingLBA	The ending logical block address. Use integer values.
verbose	The setting to capture progress details, such as percent complete, and to show the information as the volume parity is being repaired. To capture progress details, set this parameter to TRUE . To prevent capturing progress details, set this parameter to FALSE .

Notes

The starting logical block address and the ending logical block address are useful for very large single-volume LUNs. Running a volume parity check on a very large single volume LUN can take a long time. By defining the beginning address and ending address of the data blocks, you can reduce the time that a volume parity check takes to complete.

Minimum Firmware Level

6.10

Clear Drive Channel Statistics

This command resets the statistics for all of the drive channels.

Syntax

```
clear allDriveChannels stats
```

Parameters

None.

Minimum Firmware Level

6.10

Clear Storage Array Configuration

Use this command to perform one of these operations:

- Clear the entire storage array configuration, and return it back to the initial installation state
- Clear the configuration except for security information and identification information
- Clear volume group configuration information and volume configuration information only

ATTENTION Possible damage to the storage array configuration – As soon as you run this command, the existing storage array configuration is deleted.

Syntax

```
clear storageArray configuration [all | volumeGroups]
```

Parameters

Parameter	Description
None	If you do not enter a parameter, this command removes all configuration information for the storage array, except for information related to security and identification.
all	The setting to remove the entire configuration of the storage array, including security information and identification information. Removing all configuration information returns the storage array to its initial state.
volumeGroups	The setting to remove the volume configuration and the volume group configuration. The rest of the configuration stays intact.

Notes

When you run this command, the storage array becomes unresponsive, and all script processing is canceled. You must remove and re-add the storage array to resume communication with the host. To remove an unresponsive storage array, access the Enterprise Management Window, and select **Edit >> Remove**. To re-add the storage array, access the Enterprise Management Window, select **Edit >> Add Storage Array**, and enter the appropriate IP addresses.

Minimum Firmware Level

6.10

7.10 adds these parameters:

- all
- volumeGroups

Clear Storage Array Event Log

This command clears the Event Log in the storage array by deleting the data in the Event Log buffer.

ATTENTION Possible damage to the storage array configuration – As soon as you run this command, the existing Event Log in the storage array is deleted.

Syntax

```
clear storageArray eventLog
```

Parameters

None.

Minimum Firmware Level

6.10

Clear Storage Array Firmware Pending Area

This command deletes a firmware image or NVSRAM values that you have previously downloaded from the pending area buffer.

ATTENTION Possible damage to the storage array configuration – As soon as you run this command, the contents of the existing pending area in the storage array are deleted.

Syntax

```
clear storageArray firmwarePendingArea
```

Parameters

None.

Minimum Firmware Level

6.10

Clear Volume Reservations

This command clears persistent volume reservations.

Syntax

```
clear (allVolumes | volume [volumeName] |  
volumes [volumeName1 ... volumeNameN]) reservations
```

Parameters

Parameter	Description
allVolumes	The setting to clear persistent volume reservations on all of the volumes in the storage array.
volume or volumes	The name of the specific volume for which you want to clear persistent volume reservations. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

5.40

Clear Volume Unreadable Sectors

This command clears unreadable sector information from one or more volumes.

Syntax

```
clear (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN]) unreadableSectors
```

Parameters

Parameter	Description
allVolumes	The setting to clear unreadable sector information from all of the volumes in the storage array.
volume or volumes	The name of the specific volume for which you want to clear unreadable sector information. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

6.10

Create Host

This command creates a new host. If you do not specify a host group in which to create the new host, the new host is created in the Default Group.

Syntax

```
create host userLabel="hostName"  
[hostGroup=("hostGroupName" | defaultGroup)]  
[hostType=(hostTypeIndexLabel | hostTypeIndexNumber)]
```

Parameters

Parameter	Description
userLabel	The name that you want to give the host that you are creating. Enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group in which you want to create a new host. Enclose the host group name in double quotation marks (" "). (If a host group does not exist, you can create a new host group by using the <code>create hostGroup</code> command.) The defaultGroup option is the host group that contains the host to which the volume is mapped.
hostType	The index label or the index number that identifies the host type. Use the <code>show storageArray hostTypeTable</code> command to generate a list of available host type identifiers. If the host type has special characters, enclose the host type in double quotation marks (" ").

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

A host is a computer that is attached to the storage array and accesses the volumes on the storage array through its HBA host ports. You can define specific volume-to-LUN mappings to an individual host. You also can assign the host to a host group that shares access to one or more volumes.

A host group is an optional topological element that you can define if you want to designate a collection of hosts that share access to the same volumes. The host group is a logical entity. Define a host group only if you have two or more hosts that share access to the same volumes.

If you do not specify a host group in which to place the host that you are creating, the newly defined host belongs to the default host group.

Minimum Firmware Level

5.20

7.10 adds the `hostType` parameter.

Create Host Group

This command creates a new host group.

Syntax

```
create hostGroup userLabel="hostGroupName"
```

Parameter

Parameter	Description
userLabel	The name that you want to give the host group that you are creating. Enclose the host name in double quotation marks (" ").

Notes

A host group is an optional topological element that you can define if you want to designate a collection of hosts that share access to the same volumes. The host group is a logical entity. Define a host group only if you have two or more hosts that can share access to the same volumes.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

5.20

Create Host Port

This command creates a new HBA host port identification. The identification is a software value that represents the physical HBA host port to the controller. Without the HBA host port identification, the controller cannot receive instructions or data from the host port.

Syntax

```
create hostPort identifier="wwID"
userLabel="portLabel"
host="hostName"
```

Parameters

Parameter	Description
identifier	The World Wide Identifier (WWID) of the HBA host port. Enclose the WWID in double quotation marks (" ").
userLabel	The name that you want to give to the new HBA host port. Enclose the HBA host port label in double quotation marks (" ").
host	The name of the host for which you are defining an HBA host port. Enclose the host name in double quotation marks (" ").
hostType	<i>This parameter is deprecated.</i> The index label or the index number that identifies the host type. Use the <code>show storageArray hostTypeTable</code> command to generate a list of available host type identifiers. If the host type identifier has special characters, enclose the host type identifier in double quotation marks (" ").

Notes

An HBA host port is a physical connection on a host adapter that resides within a host computer. An HBA host port provides a host access to the volumes in a storage array. If the host bus adapter has only one physical connection (one host port), the terms host port and host bus adapter are synonymous. You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

5.20

7.10 deprecates the `hostType` parameter. The `hostType` parameter has been added to the `create host` command.

Create RAID Volume (Automatic Drive Select)

This command creates a volume group across the drives in the storage array, and a new volume in the volume group. The storage array controllers choose the drives to be included in the volume.

IMPORTANT If you have drives with different capacities, you cannot automatically create volumes by specifying the `driveCount` parameter. If you want to create volumes with drives of different capacities.

Syntax

```
create volume driveCount=numberOfDrives
raidLevel=(0 | 1 | 3 | 5 | 6)
userLabel="volumeName"
[driveType=(fibre | SATA | SAS | PATA)
capacity=volumeCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia)
trayLossProtect=(TRUE | FALSE)
dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
<code>driveCount</code>	The number of unassigned drives that you want to use in the volume group.
<code>raidLevel</code>	The RAID level of the volume group that contains the volume. Valid values are 0 , 1 , 3 , 5 , or 6 .
<code>userLabel</code>	The name that you want to give to the new volume. Enclose the new volume name in double quotation marks (" ").
<code>driveType</code>	The type of drive that you want to use in the volume. You cannot mix drive types. Valid drive types are fibre , SATA , SAS , or PATA .
<code>capacity</code>	The size of the volume that you are adding to the storage array. Size is defined in units of bytes , KB , MB , GB , or TB .

Parameter	Description
owner	The controller that owns the volume. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. If you do not specify an owner, the controller firmware determines the owner.
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
usageHint	The setting for both <code>cacheReadPrefetch</code> parameter and the <code>segmentSize</code> parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the volume. Valid values are fileSystem, dataBase, or multiMedia .
trayLossProtect	The setting to enforce tray loss protection when you create the volume group. To enforce tray loss protection, set this parameter to TRUE . The default setting is FALSE .
dssPreAllocate	The setting to make sure that reserve capacity is allocated for future segment size increases. The default value is TRUE .

Notes

The `driveCount` parameter lets you choose the number of drives that you want to use in the volume group. You do not need to specify the drives by tray ID and slot ID. The controllers choose the specific drives to use for the volume group.

The `owner` parameter defines which controller owns the volume.

If you do not specify a capacity using the `capacity` parameter, all of the drive capacity that is available in the volume group is used. If you do not specify capacity units, **bytes** is used as the default value.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drives into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

You do not need to enter a value for the `cacheReadPrefetch` parameter or the `segmentSize` parameter. If you do not enter a value, the controller firmware uses the `usageHint` parameter with **fileSystem** as the default value. Entering a value for the `usageHint` parameter and a value for the `cacheReadPrefetch` parameter or a value for the `segmentSize` parameter does not cause an error. The value that you enter for the `cacheReadPrefetch` parameter or the `segmentSize` parameter takes priority over the value for the `usageHint` parameter.

Tray Loss Protection

For tray loss protection to work, each drive in a volume group must be on a separate tray. If you set the `trayLossProtect` parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs operations, but the volume group that you create might not have tray loss protection.

Tray loss protection is not valid when you create volumes on existing volume groups.

Minimum Firmware Level

5.20

7.10 adds RAID 6 capability and the `dssPreAllocate` parameter.

Create RAID Volume (Free Extent Based Select)

This command creates a volume in the free space of a volume group.

Syntax

```
create volume volumeGroup=[stringLiteral]
userLabel="volumeName"
[freeCapacityArea=freeCapacityIndexNumber
capacity=volumeCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia)]
[dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) for a specific volume group in your storage array.
userLabel	The name that you want to give the new volume. Enclose the new volume name in double quotation marks (" ").
freeCapacityArea	The index number of the free space in an existing volume group that you want to use to create the new volume. Free capacity is defined as the free capacity between existing volumes in a volume group. For example, a volume group might have the following areas: volume 1, free capacity, volume 2, free capacity, volume 3, free capacity. To use the free capacity following volume 2, you would specify: freeCapacityArea=2 Run the <code>show volumeGroup</code> command to determine if the free capacity area exists.
capacity	The size of the volume that you are adding to the storage array. Size is defined in units of bytes , KB , MB , GB , or TB .
owner	The controller that owns the volume. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. If you do not specify an owner, the controller firmware determines the owner.
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .

Parameter	Description
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
usageHint	The settings for both the <code>cacheReadPrefetch</code> parameter and the <code>segmentSize</code> parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the volume. Valid values are fileSystem, dataBase, or multiMedia .
dssPreAllocate	The setting to make sure that reserve capacity is allocated for future segment size increases. The default value is TRUE .

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

The `owner` parameter defines which controller owns the volume. The preferred controller ownership of a volume is the controller that currently owns the volume group.

If you do not specify a capacity using the `capacity` parameter, all of the available capacity in the free capacity area of the volume group is used. If you do not specify capacity units, **bytes** is used as the default value.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drives into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**. You do not need to enter a value for the `cacheReadPrefetch` parameter or the `segmentSize` parameter. If you do not enter a value, the controller firmware uses the `usageHint` parameter with **fileSystem** as the default value.

Entering a value for the `usageHint` parameter and a value for the `cacheReadPrefetch` parameter or a value for the `segmentSize` parameter does not cause an error. The value that you enter for the `cacheReadPrefetch` parameter or the `segmentSize` parameter takes priority over the value for the `usageHint` parameter.

Minimum Firmware Level

5.20

7.10 adds the `dssPreAllocate` parameter.

Create RAID Volume (Manual Drive Select)

This command creates a new volume group and volume and lets you specify the drives for the volume.

IMPORTANT You cannot use mixed drive types in the same volume group and volume. This command fails if you specify different types of drives for the RAID volume.

Syntax

```
create volume drives=(trayID1,slotID1...trayIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6)
userLabel="volumeName"
[capacity=volumeCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia)
trayLossProtect=(TRUE | FALSE)
dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
drives	The drives that you want to assign to the volume that you want to create. Specify the tray ID and slot ID for each unassigned drive that you want to assign to the volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
raidLevel	The RAID level of the volume group that contains the volume. Valid values are 0 , 1 , 3 , 5 , or 6 .
userLabel	The name that you want to give the new volume. Enclose the new volume name in double quotation marks (" ").
capacity	The size of the volume that you are adding to the storage array. Size is defined in units of bytes , KB , MB , GB , or TB .

Parameter	Description
owner	The controller that owns the volume. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. If you do not specify an owner, the controller firmware determines the owner.
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
usageHint	The settings for both the <code>cacheReadPrefetch</code> parameter and the <code>segmentSize</code> parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the volume. Valid values are fileSystem, dataBase, or multiMedia .
trayLossProtect	The setting to enforce tray loss protection when you create the repository. To enforce tray loss protection, set this parameter to TRUE . The default value is FALSE .
dssPreAllocate	The setting to make sure that reserve capacity is allocated for future segment size increases. This default value is TRUE .

Notes

If you set the `raidLevel` parameter to RAID 1, the controller firmware takes the list of drives and pairs them by using the following algorithm:

$$\text{Data drive} = X$$
$$\text{Parity drive} = N/2 + X$$

where X is 1 to $N/2$, and N is the number of drives in the list. For example, if you have six drives, the mirror pairs are as follows:

Data	Parity
1	$N/2 + 1 = 4$
2	$N/2 + 2 = 5$
3	$N/2 + 3 = 6$

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

The `owner` parameter defines which controller owns the volume. The preferred controller ownership of a volume is the controller that currently owns the volume group.

If you do not specify a capacity using the `capacity` parameter, all of the drive capacity that is available in the volume group is used. If you do not specify capacity units, **bytes** is used as the default value.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drive into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

You do not need to enter a value for the `cacheReadPrefetch` parameter or the `segmentSize` parameter. If you do not enter a value, the controller firmware uses the `usageHint` parameter with **fileSystem** as the default value. Entering a value for the `usageHint` parameter and a value for the `cacheReadPrefetch` parameter or a value for the `segmentSize` parameter does not cause an error. The value that you enter for the `cacheReadPrefetch` parameter or the `segmentSize` parameter takes priority over the value for the `usageHint` parameter.

For tray loss protection to work, each drive in a volume group must be on a separate tray. If you set the `trayLossProtect` parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the `trayLossProtect` parameter to

FALSE, the storage array performs operations, but the volume group that you create might not have tray loss protection. Tray loss protection is not valid when you create volumes on existing volume groups.

Minimum Firmware Level

5.20

7.10 adds RAID 6 capability and the `dssPreAllocate` parameter.

Create Remote Mirror

This command creates both the primary volume and the secondary volume for a remote-mirror pair. This command also sets the write mode (Synchronous or Asynchronous) and the synchronization priority.

Syntax

```
create remoteMirror primary="primaryVolumeName"
secondary="secondaryVolumeName"
(remoteStorageArrayName="storageArrayName" |
remoteStorageArrayWwn="wwID")
[remotePassword="password"
syncPriority=(highest | high | medium | low | lowest)
autoResync=(enabled | disabled)
writeOrder=(preserved | notPreserved)
writeMode=(synchronous | asynchronous)]
```

Parameters

Parameter	Description
primary	The name of an existing volume on the local storage array that you want to use for the primary volume. Enclose the primary volume name in double quotation marks (" ").
secondary	The name of an existing volume on the remote storage array that you want to use for the secondary volume. Enclose the secondary volume name in double quotation marks (" ").
remoteStorageArrayName	The name of the remote storage array. Enclose the remote storage array name in double quotation marks (" ").
remoteStorageArrayWwn	The World Wide Identifier (WWID) of the remote storage array. Enclose the WWID in double quotation marks (" ").
remotePassword	The password for the remote storage array. Use this parameter when the remote storage array is password protected. Enclose the password in double quotation marks (" ").
syncPriority	The priority that full synchronization has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .

Parameter	Description
autoResync	The settings for automatic resynchronization between the primary volumes and the secondary volumes of a remote-mirror pair. This parameter has the following values: <ul style="list-style-type: none">• enabled – Automatic resynchronization is turned on. You do not need to do anything further to resynchronize the primary volume and the secondary volume.• disabled – Automatic resynchronization is turned off. To resynchronize the primary volumes and the secondary volume, you must run the <code>resume remoteMirror</code> command.
writeOrder	The write order for data transmission between the primary volume and the secondary volume. Valid values are preserved or notPreserved .
writeMode	How the primary volume writes to the secondary volume. Valid values are synchronous or asynchronous .

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

When you choose the primary volume and the secondary volume, the secondary volume must be of equal or greater size than the primary volume. The RAID level of the secondary volume does not have to be the same as the primary volume.

The IS4000 controllers can define a maximum of 64 remote mirrors.

Passwords are stored on each storage array in a management domain. If a password was not previously set, you do not need a password. The password can be any combination of alphanumeric characters with a maximum of 30 characters. (You can define a storage array password by using the `set storageArray` command.)

Synchronization priority defines the amount of system resources that are used to synchronize the data between the primary volume and the secondary volume of a mirror relationship. If you select the highest priority level, the data synchronization uses the most system resources to perform the full synchronization, which decreases performance for host data transfers.

The `writeOrder` parameter applies only to asynchronous mirrors and makes them become part of a consistency group. Setting the `writeOrder` parameter to **preserved** causes the remote-mirror pair to transmit data from the primary volume to the secondary volume in the same order as the host writes to the primary volume. In the event of a transmission link failure, the data is buffered until a full synchronization can occur. This action can require additional system overhead to maintain the buffered data, which slows operations. Setting the `writeOrder` parameter to **notPreserved** frees the system from having to maintain data in a buffer, but it requires forcing a full synchronization to make sure that the secondary volume has the same data as the primary volume.

Minimum Firmware Level

6.10

Create Snapshot Volume

This command creates a snapshot of a base volume. This command defines three ways to create a snapshot volume:

- User-defined drives
- User-defined volume group
- User-defined number of drives

If you choose to define a number of drives, the controller firmware chooses which drives to use for the snapshot volume.

Syntax (User-Defined Drives)

```
create snapshotVolume baseVolume=baseVolumeName
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
userLabel=snapshotVolumeName
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel=repositoryName
repositoryFullPolicy=(failBaseWrites | failSnapShot)
trayLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Volume Group)

```
create snapshotVolume baseVolume=baseVolumeName
[repositoryVolumeGroup=volumeGroupName
freeCapacityArea=freeCapacityIndexNumber
userLabel=snapshotVolumeName
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel=repositoryName
repositoryFullPolicy=(failBaseWrites | failSnapShot)
trayLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Number of Drives)

```

create snapshotVolume baseVolume="baseVolumeName"
[RAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
driveType=(fibre | SATA | SAS | PATA)
userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failSnapShot)
trayLossProtect=(TRUE | FALSE)]

```

A snapshot volume is a point-in-time image of a volume in a storage array, which is created by using the `create snapshotVolume` command and specific parameters as described in the next section.

Parameters

Parameter	Description
baseVolume	The name of the base volume from which you want to take a snapshot. Enclose the base volume name in double quotation marks (" ").
repositoryRAIDLevel	The RAID level for the snapshot repository volume. Valid values are 1 , 3 , 5 , or 6 .
repositoryDrives	The drives that you want to assign to the snapshot repository volume. Specify the tray ID and the slot ID for each drive that you assign to the snapshot repository volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the list of drives in parentheses.
repositoryDriveCount	The number of unassigned drives that you want to use for the snapshot repository volume.
driveType	The type of drives that you want to use for the snapshot repository volume. Valid drive types are SCSI , fibre , or SAS .

Parameter	Description
repositoryVolumeGroup	The alphanumeric identifier (including - and _) of the volume group where the snapshot repository volume is located.
freeCapacityArea	The index number of the free space in an existing volume group that you want to use to create the snapshot repository volume. Free capacity is defined as the free capacity between existing volumes in a volume group. For example, a volume group might have the following areas: volume 1, free capacity, volume 2, free capacity, volume 3, free capacity. To use the free capacity following volume 2, you would specify: freeCapacityArea=2 Run the <code>show volumeGroup</code> command to determine if a free capacity area exists.
userLabel	The name that you want to give the snapshot volume. Enclose the snapshot volume name in double quotation marks (" ").
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the snapshot repository volume is nearing full. Use integer values. For example, a value of 70 means 70 percent. The default value is 50 .
repositoryPercentOfBase	The size of the snapshot repository volume as a percentage of the base volume. Use integer values. For example, a value of 40 means 40 percent. The default value is 20 .

Parameter	Description
repositoryUserLabel	The name that you want to give to the snapshot repository volume. Enclose the snapshot repository volume name in double quotation marks (" ").
repositoryFullPolicy	How you want snapshot processing to continue if the snapshot repository volume is full. You can choose to fail writes to the base volume (failBaseWrites) or fail the snapshot volume (failSnapShot). The default value is failSnapShot .
trayLossProtect	The setting to enforce tray loss protection when you create the snapshot repository volume. To enforce tray loss protection, set this parameter to TRUE . The default value is FALSE .

Command Examples

```
smcli 172.22.333.444 172. 22. 333.555 -c "create snapshotVolume
baseVolume="customer_data" ;"
```

```
smcli 172.22.333.444 172.22.333.445 -c "create FlashCopyLogicalDrive
baseLogicalDrive="SQL_Data" repositoryArray="Copy_Serv_Volumes"
userLabel="\SQL_data_Flash_1\" repositoryPercentOfBase=20;"
```

Notes

The volume that you are taking a snapshot of must be a standard volume in the storage array. The maximum number of snapshot volumes that you can create is one-half of the total number of volumes that are supported by a controller.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

One technique for naming the snapshot volume and the snapshot repository volume is to add a hyphenated suffix to the original base volume name. The suffix distinguishes between the snapshot volume and the snapshot repository volume. For example, if you have a base volume with a name of Engineering Data, the snapshot volume can have a name of Engineering Data-S1, and the snapshot repository volume can have a name of EngineeringData-R1.

If you do not choose a name for either the snapshot volume or the snapshot repository volume, the storage management software creates a default name by using the base volume name. An example of the snapshot volume name that the controllers might create is, if the base volume name is `aaa` and does not have a snapshot volume, the default snapshot volume name is `aaa-1`. If the base volume already has $n-1$ number of snapshot volumes, the default name is `aaa-n`. An example of the snapshot repository volume name that the controller might create is, if the base volume name is `aaa` and does not have a snapshot repository volume, the default snapshot repository volume name is `aaa-R1`. If the base volume already has $n-1$ number of snapshot repository volumes, the default name is `aaa-Rn`.

If you do not specify the unconfigured space or free space, the snapshot repository volume is placed in the same volume group as the base volume. If the volume group where the base volume resides does not have enough space, this command fails.

When you assign the drives, if you set the `trayLossProtect` parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs operations, but the volume group that you create might not have tray loss protection.

When the controller firmware assigns the drives, if you set the `trayLossProtect` parameter to **TRUE**, the storage array returns an error if the controller firmware cannot provide drives that result in the new volume group having tray loss protection. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs the operation even if it means the volume group might not have tray loss protection.

Minimum Firmware Level

5.00

7.10 adds RAID 6 capability

Create Volume Copy

This command creates a volume copy and starts the volume copy operation.

IMPORTANT You can have a maximum of eight volume copies in progress at one time. If you try to create more than eight volume copies at one time, the controllers return a status of Pending until one of the volume copies that is in progress finishes and returns a status of Complete.

Syntax

```
create volumeCopy source="sourceName"
target="targetName"
[copyPriority=(highest | high | medium | low | lowest)
targetReadOnlyEnabled=(TRUE | FALSE)]
```

Parameters

Parameter	Description
source	The name of an existing volume that you want to use as the source volume. Enclose the source volume name in double quotation marks (" ").
target	The name of an existing volume that you want to use as the target volume. Enclose the target volume name in double quotation marks (" ").
copyPriority	The priority that volume copy has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .
targetReadOnlyEnabled	The setting so that you can write to the target volume or only read from the target volume. To write to the target volume, set this parameter to FALSE . To prevent writing to the target volume, set this parameter to TRUE .

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Copy priority defines the amount of system resources that are used to copy the data between the source volume and the target volume of a volume copy pair. If you select the highest priority level, the volume copy uses the most system resources to perform volume copy, which decreases performance for host data transfers.

Minimum Firmware Level

5.40

Create Volume Group

This command creates either a free-capacity volume group or a volume group with one volume when you enter a set of unassigned drives.

Syntax

```
create volumeGroup [volumeGroupName]  
drives=(trayID1,slotID1 ... trayIDn,slotIDn)  
raidLevel=(0 | 1 | 3 | 5 | 6)  
[capacity=volumeCapacity  
owner=(a | b)  
cacheReadPrefetch=(TRUE | FALSE)  
segmentSize=segmentSizeValue  
usageHint=(fileSystem | dataBase | multiMedia)  
trayLossProtect=(TRUE | FALSE)]
```

Parameters

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) that you want to give the new volume group. Enclose the volume group identifier in square brackets ([]).
drives	The drives that you want to assign to the volume group that you want to create. Specify the tray ID and slot ID for each drive that you assign to the volume group. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
raidLevel	The RAID level of the volume group that contains the volume. Valid values are 0 , 1 , 3 , 5 , or 6 .
capacity	The size of the volume that you are adding to the storage array. Size is defined in units of bytes , KB , MB , GB , or TB . These examples show the syntax: capacity=500 bytes capacity=2 GB Note: A white-space (' ') needs to be placed between the specified size and defined unit (e.g., 40 GB).
owner	The setting for which controller owns the volume group. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. If you do not specify an owner, the controller firmware determines the owner.
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .

Parameter	Description
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
usageHint	The setting for both the <code>cacheReadPrefetch</code> parameter and the <code>segmentSize</code> parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the volume. Valid values are fileSystem, dataBase, or multiMedia .
trayLossProtect	The setting to enforce tray loss protection when you create the volume group. To enforce tray loss protection, set this parameter to TRUE . The default value is FALSE .

Notes

The `drives` parameter lets you choose the number of drives that you want to use in the volume group. If you choose this option, you do not need to specify the drives by tray ID and slot ID. The controllers choose the specific drives to use for the volume group.

If you do not specify a capacity by using the `capacity` parameter, all of the drive capacity that is available in the volume group is used. If you do not specify capacity units, **bytes** is used as the default value.

Cache Read Prefetch

The `cacheReadPrefetch` command lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drives into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

You do not need to enter a value for the `cacheReadPrefetch` parameter or the `segmentSize` parameter. If you do not enter a value, the controller firmware uses the `usageHint` parameter with **fileSystem** as the default value. Entering a value for the `usageHint` parameter and a value for the `cacheReadPrefetch` parameter or a value for the `segmentSize` parameter does not cause an error. The value that you enter for the `cacheReadPrefetch` parameter or the `segmentSize` parameter takes priority over the value for the `usageHint` parameter.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Tray Loss Protection

For tray loss protection to work, each drive in a volume group must be in a separate tray. If you set the `trayLossProtect` parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs operations, but the volume group that you create might not have tray loss protection. Tray loss protection is not valid when you create volumes on existing volume groups.

Minimum Firmware Level

7.10

Deactivate Remote Mirror

This command deactivates the Remote Volume Mirroring feature, disassembles the mirror repository volume, and releases the controller-owner of the secondary volume. The controller host port that is dedicated to the secondary volume is available for host data transfers.

Syntax

```
deactivate storageArray feature=remoteMirror
```

Parameters

None.

Minimum Firmware Level

6.10

Delete Host

This command deletes a host.

Syntax

```
delete host [hostName]
```

Parameter

Parameter	Description
host	The name of the host that you want to delete. Enclose the host name in square brackets ([]). If the host name has special characters, you also must enclose the host name in double quotation marks (" ").

Notes

A host is a computer that is attached to the storage array and accesses the volumes on the storage array through the HBA host ports on the host.

Minimum Firmware Level

5.20

Delete Host Group

This command deletes a host group.

ATTENTION Possible damage to the storage array configuration – This command deletes all of the host definitions in the host group.

Syntax

```
delete hostGroup [hostGroupName]
```

Parameter

Parameter	Description
hostGroup	The name of the host group that you want to delete. Enclose the host group name in square brackets ([]). If the host group name has special characters, you also must enclose the host group name in double quotation marks (" ").

Notes

A host group is an optional topological element that is a collection of hosts that share access to the same volumes. The host group is a logical entity.

Minimum Firmware Level

5.20

Delete Host Port

This command deletes an HBA host port identification. The identification is a software value that represents the physical HBA host port to the controller. By deleting the identification, the controller no longer recognizes instructions and data from the HBA host port.

Syntax

```
delete hostPort [hostPortName]
```

Parameter

Parameter	Description
hostPort	The name of the HBA host port that you want to delete. Enclose the name of the HBA host port in square brackets ([]).

Notes

An HBA host port is a physical connection on a host bus adapter that resides within a host computer. An HBA host port provides a host access to the volumes in a storage array. If the host bus adapter has only one physical connection (one host port), the terms HBA host port and host bus adapter are synonymous.

Minimum Firmware Level

5.20

Delete Volume

This command deletes one or more standard volumes, snapshot volumes, or snapshot repository volumes.

ATTENTION Possible damage to the storage array configuration – All of the data in the volume is lost as soon as you run this command.

Syntax

```
delete (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN])
removeVolumeGroup=(TRUE | FALSE)
```

Parameters

Parameter	Description
allVolumes	This parameter deletes all of the volumes in a storage array.
volume or volumes	The name of the volume that you want to delete. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").
removeVolumeGroup	Deleting the last volume in a volume group does not delete the volume group. You can have a stand-alone volume group (minus any volumes). To remove the stand-alone volume group, set this parameter to TRUE . To keep stand-alone volume groups intact, set this parameter to FALSE .

Notes

When you use the `allVolumes` parameter, this command deletes volumes until all of the volumes are removed or until an error is encountered. If an error is encountered, this command does not try to delete the remaining volumes. Deleting volumes from different volume groups is possible. All of the volume groups that become empty are deleted if you set the `removeVolumeGroup` parameter to **TRUE**. If you want to delete an entire volume group, you can also use the `delete volumeGroup` command.

Minimum Firmware Level

6.10: Firmware version 7.10 adds the `removeVolumeGroup` parameter.

Delete Volume Group

This command deletes an entire volume group and its associated volumes.

ATTENTION **Possible damage to the storage array configuration** – All of the data in the volume group is lost as soon as you run this command.

Syntax

```
delete volumeGroup [volumeGroupName]
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) of the volume group that you want to delete. Enclose the volume group identifier in square brackets ([]).

Minimum Firmware Level

6.10

Diagnose Controller

This command runs diagnostic tests on the controller. The diagnostic tests consist of loopback tests in which data is written to the drives and read from the drives.

Syntax

```
diagnose controller [(a | b)]  
loopbackDriveChannel=(allchannels | (1 | 2 | 3 | 4))  
testID=(1 | 2 | 3 | discreteLines)  
[patternFile="filename"]
```

Parameters

Parameter	Description
controller	The controller on which you want to run the diagnostic tests. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]). If you do not specify a controller, the storage management software returns a syntax error.
loopbackDriveChannel	The drive channels on which you want to run the diagnostic tests. You can either choose to run the diagnostics on all channels or select a specific channel on which to run diagnostics. If you select a specific channel, valid values for the drive channels are 1 , 2 , 3 , or 4 .
testID	The identifier for the diagnostic test you want to run. The identifier and corresponding tests are as follows: <ul style="list-style-type: none"> • 1 – Read test • 2 – Write test • 3 – Data loop-back test • discreteLines – Discrete lines diagnostic test
patternFile	The name of a file that contains a data pattern that you want to use as test data. Enclose the file name of the data pattern in double quotation marks (" ").

Notes

When you run a data loop-back test, you can optionally specify a file that contains a data pattern. If you do not specify a file, the controller firmware provides a default pattern.

Discrete lines are control lines and status lines that are connected between two controllers in a controller tray. The discrete lines diagnostic test lets each controller check that control signal transitions can be observed at the control inputs of the alternate controller. The discrete lines diagnostic test automatically runs after each power-cycle or each controller-reset. You can run the discrete lines diagnostic test after you have replaced a component that failed the initial discrete lines diagnostic test.

The discrete lines diagnostic test returns one of the following messages:

- When the discrete lines diagnostic test runs successfully, this message appears:
- The controller discrete lines successfully passed the diagnostic test. No failures were detected.
- If the discrete lines diagnostic test fails, this message appears:
- One or more controller discrete lines failed the diagnostic test.
- If the CLI cannot run the discrete lines diagnostic test, the CLI returns Error 270, which means that the discrete lines diagnostic test could not start nor complete.

Minimum Firmware Level

6.10 adds the read test, the write test, and the data loop-back test.

6.14 adds the discrete lines diagnostic test.

Diagnose Remote Mirror

This command tests the connection between the specified primary volumes and the mirror volumes on a storage array with the Remote Volume Mirroring feature enabled.

Syntax

```
diagnose remoteMirror (primary [primaryVolumeName] |  
primaries [primaryVolumeName1 ... primaryVolumeNameN])  
testID=connectivity
```

Parameters

Parameter	Description
primary or primaries	The name of the primary volume of the remote mirror pair that you want to test. You can enter more than one primary volume name. Enclose the primary volume names in square brackets ([]). If the primary volume name has special characters, you also must enclose the primary volume name in double quotation marks (" ").

Minimum Firmware Level

6.10

Disable Storage Array Feature

This command disables a storage array feature. Run the `show storageArray` command to show a list of the feature identifiers for all enabled features in the storage array.

Syntax

```
disable storageArray [featurePack | feature=(storagePartition2 |
storagePartition4 | storagePartition8 | storagePartition16 |
storagePartition64 | storagePartitionMax | snapshot | remoteMirror |
volumeCopy | goldKey | mixedDriveTypes)]
```

Parameters

None.

Notes

If you specify the `remoteMirror` parameter, this command disables the Remote Volume Mirroring feature and takes away the structure of the mirror repository volume.

Minimum Firmware Level

5.00

6.50 adds the `goldKey` parameter and the `mixedDriveTypes` parameter.

Download Drive Firmware

This command downloads a firmware image to a drive.

ATTENTION **Possible damage to the storage array configuration** – Downloading drive firmware incorrectly can result in damage to the drives or a loss of data.

Syntax

```
download drive [trayID,slotID] firmware file="filename"
```

Parameters

Parameter	Description
drive	The drive to which you want to download the firmware image. Specify the tray ID and slot ID for the drive. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 .
filename	The file path and the file name of the file that contains the firmware image. Enclose the file path and the file name of the firmware image in double quotation marks (“ ”).

Notes

Before trying to download drive firmware, take these precautions:

- Stop all I/O activity to the storage array before you download the firmware image.
- Make sure that the firmware image file is compatible with the drive tray. If you download a firmware image file that is not compatible with the drive tray that you have selected, the drive tray might become unusable.
- Do not make any configuration changes to the storage array while you download drive firmware. Trying to make a configuration change can cause the firmware download to fail and make the selected drives unusable.

You can use this command to test the firmware on one drive before you install the firmware on all of the drives in a storage array. (Use the `download storageArray driveFirmware` command to download firmware on all of the drives in the storage array.) This command blocks all I/O activity until the download finishes or fails. The download returns one of these statuses:

- Successful
- Unsuccessful With Reason
- Never Attempted With Reason

Minimum Firmware Level

6.10

Download Environmental Card Firmware

This command downloads ESM firmware.

Syntax

```
download (allTrays | tray [trayID])
firmware file="filename"
```

Parameters

Parameter	Description
tray	The drive tray that contains the ESM card to which you want to load new firmware. Tray ID values are 0 to 99 . Enclose the tray ID value in square brackets ([]).
file	The file path and the file name of the file that contains the firmware image. Enclose the file path and the file name of the firmware image in double quotation marks (" ").

Notes

You can use these parameters:

- The `allTrays` parameter downloads new firmware to all of the trays in the storage array.
- The `tray` parameter downloads new firmware to a specific drive tray. If you need to download new firmware to more than one drive tray, but not all drive trays, you must enter this command for each drive tray.

Minimum Firmware Level

5.20

Download Storage Array Drive Firmware

This command downloads firmware images to all of the drives in the storage array.

Syntax

```
download storageArray driveFirmware file="filename"  
[file="filename2"... file="filenameN"]
```

Parameter

Parameter	Description
file	The file path and the file name of the file that contains the firmware image. Enclose the file path and the file name of the firmware image in double quotation marks (" ").

Notes

When you run this command, you can download more than one firmware image file to the drives in a storage array. The number of firmware image files that you can download depends on the storage array. The storage management software returns an error if you try to download more firmware image files than the storage array can accept.

You can schedule downloads for multiple drives at the same time, including multiple drives in a redundant volume group. Each firmware image file contains information about the drive types on which the firmware image runs. The specified firmware images can be downloaded only to a compatible drive. Use the `download drive firmware` command to download a firmware image to a specific drive. The `download storageArray driveFirmware` command blocks all I/O activity until a download try has been made for each candidate drive or you run the `stop storageArray downloadDriveFirmware` command. When the `download storageArray driveFirmware` command finishes downloading the firmware image, each candidate drive shows the download status for each drive. One of these statuses is returned:

- Successful
- Unsuccessful With Reason
- Never Attempted With Reason

Minimum Firmware Level

5.20

Download Storage Array Firmware/NVSRAM

This command downloads firmware and, optionally, NVSRAM values for the storage array controller. If you want to download only NVSRAM values, use the `download storageArray NVSRAM` command.

Syntax

```
download storageArray firmware [, NVSRAM ]
file= "filename" [, "NVSRAM-filename"]
[downgrade=(TRUE | FALSE)]
[activateNow=(TRUE | FALSE)]
```

Parameters

Parameter	Description
NVSRAM	The setting to download a file with NVSRAM values when you download a firmware file. Do not include square brackets with this parameter. Include a comma after the <code>firmware</code> parameter.
file	The file path and the name of the file that contains the firmware. Valid file names must end with a <code>.dlp</code> extension. Enclose the file path and the file name in double quotation marks (" ").
NVSRAM-filename	The file path and the name of the file that contains the NVSRAM values. Valid file names must end with a <code>.dlp</code> extension. Enclose the NVSRAM file name in double quotation marks (" "). Include a comma after the file name.
downgrade	The setting to load firmware that is a previous version. The default value is FALSE . Set the <code>downgrade</code> parameter to TRUE if you want to download an earlier version of firmware.
activateNow	The setting to activate the firmware image and the NVSRAM image. The default value is TRUE . If you set the <code>activateNow</code> parameter to FALSE , you must run the <code>activate storageArray firmware</code> command to activate the firmware values and the NVSRAM values at a later time.

Minimum Firmware Level

5.00

Download Storage Array NVSRAM

This command downloads NVSRAM values for the storage array controller.

Syntax

```
download storageArray NVSRAM file="filename"
```

Parameter

Parameter	Description
file	The file path and the file name of the file that contains the NVSRAM values. Valid file names must end with a .dlp extension. Enclose the file path and the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Enable Controller

This command revives a controller that has become quiesced while running diagnostics.

Syntax

```
enable controller [(a | b)] dataTransfer
```

Parameter

Parameter	Description
controller	The controller that you want to revive. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]). If you do not specify a controller, the storage management software returns a syntax error.

Minimum Firmware Level

6.10

Enable Storage Array Feature Key

This command enables a feature by using a Feature Key file.

Syntax

```
enable storageArray [featurePack | feature]  
file="filename"
```

Parameter

Parameter	Description
file	The file path and the file name of a valid feature key file. Valid file names for feature key files end with a <code>.key</code> extension. Enclose the file path and file name in double quotation marks (" ").

Notes

A feature pack is a predefined set of premium features, such as SANshare Storage Partitioning and Remote Volume Mirroring. These features are combined for the convenience of the users.

A feature is an additional application to enhance the capabilities of a storage array.

Minimum Firmware Level

6.10

6.50 adds the `featurePack` parameter.

Recopy Volume Copy

This command reinitiates a volume copy operation by using an existing volume copy pair.

Syntax

```
recopy volumeCopy target [targetName]
[source [sourceName]]
[copyPriority=(highest | high | medium | low | lowest)
targetReadOnlyEnabled=(TRUE | FALSE)]
```

Parameters

Parameter	Description
target	The name of the target volume for which you want to reinitiate a volume copy operation. Enclose the target volume name in square brackets ([]). If the target volume name has special characters, you also must enclose the target volume name in double quotation marks (" ").
source	The name of the source volume for which you want to reinitiate a volume copy operation. Enclose the source volume name in square brackets ([]). If the source volume name has special characters, you also must enclose the source volume name in double quotation marks (" ").
copyPriority	The priority that the volume copy has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .
targetReadOnlyEnabled	The setting so that you can write to the target volume or only read from the target volume. To write to the target volume, set this parameter to FALSE . To prevent writing to the target volume, set this parameter to TRUE .

Notes

Copy priority defines the amount of system resources that are used to copy the data between the source volume and the target volume of a volume copy pair. If you select the highest priority level, the volume copy uses the most system resources to perform the volume copy, which decreases performance for host data transfers.

Minimum Firmware Level

6.10

Recover RAID Volume

This command creates a RAID volume with the given properties without initializing any of the user data areas on the drives. Parameter values are derived from the Recovery Profile data file for the storage array.

Syntax

```
recover volume (drive=(trayID,slotID) |
drives=(trayID1,slotID1 ... trayIDn,slotIDn) |
volumeGroup=volumeGroupName)
[newVolumeGroup=volumeGroupName]
userLabel="volumeName"
capacity=volumeCapacity
offset=offsetValue
raidLevel=(0 | 1 | 3 | 5 | 6)
segmentSize=segmentSizeValue
[owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)]
```

Parameters

Parameter	Description
drive or drives	The drives that you want to assign to the volume that you want to create. Specify the tray ID and slot ID for each drive that you assign to the volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
volumeGroup	The name of the volume group in which you want to create the new volume. (To determine the names of the volume groups in your storage array, run the <code>show storageArray profile</code> command.)
newVolumeGroup	This parameter enables the user to specify a name for a new volume group that is automatically created by the controller.
userLabel	The name that you want to give the new volume. Enclose the new volume name in double quotation marks (" ").
capacity	The size of the volume that you are adding to the storage array. Size is defined in units of bytes , KB , MB , GB , or TB .

Parameter	Description
offset	The number of blocks from the start of the volume group to the start of the referenced volume.
raidLevel	The RAID level of the volume group that contains the drives. Valid values are 0, 1, 3, 5, or 6 .
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
owner	The controller that owns the volume. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. If you do not specify an owner, the controller firmware determines the owner.
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

The `owner` parameter defines which controller owns the volume. The preferred controller ownership of a volume is the controller that currently owns the volume group.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from disk into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

Minimum Firmware Level

5.43

7.10 adds RAID 6 capability and the `newVolumeGroup` parameter.

Re-create Remote Volume Mirroring Repository Volume

This command creates a new Remote Volume Mirroring repository volume (also called a mirror repository volume) by using the parameters defined for a previous mirror repository volume. The underlying requirement is that you have previously created a mirror repository volume. When you use this command, you can define the mirror repository volume in one of three ways: user-defined drives, user-defined volume group, or user-defined number of drives for the mirror repository volume. If you choose to define a number of drives, the controller firmware chooses which drives to use for the mirror repository volume.

Syntax (User-Defined Drives)

```
recreate storageArray mirrorRepository
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
[trayLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Volume Group)

```
recreate storageArray mirrorRepository
repositoryVolumeGroup=volumeGroupName [freeCapacityArea=freeCapacityIndexNum
ber]
```

Syntax (User-Defined Number of Drives)

```
recreate storageArray mirrorRepository
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
[driveType=(fibre | SATA | SAS | PATA)]
[trayLossProtect=(TRUE | FALSE)]
```

Parameters

Parameter	Description
repositoryRAIDLevel	The RAID level for the mirror repository volume. Valid values are 1, 3, 5, or 6 .
repositoryDrives	The drives for the mirror repository volume. Specify the tray ID and slot ID for each drive that you assign to the mirror repository volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
repositoryVolumeGroup	The name of the volume group where the mirror repository volume is located.
freeCapacityArea	<p>The index number of the free space in an existing volume group that you want to use to re-create the mirror repository volume. Free capacity is defined as the free capacity between existing volumes in a volume group. For example, a volume group might have the following areas: volume 1, free capacity, volume 2, free capacity, volume 3, free capacity. To use the free capacity following volume 2, you would specify:</p> <pre>freeCapacityArea=2</pre> <p>Run the <code>show volumeGroup</code> command to determine if a free capacity area exists.</p>
repositoryDriveCount	The number of unassigned drives that you want to use for the mirror repository volume.
driveType	The type of drives that you want to use for the mirror repository volume. Valid drive types are fibre, SATA, SAS, or PATA .
trayLossProtect	The setting to enforce tray loss protection when you create the mirror repository volume. To enforce tray loss protection, set this parameter to TRUE . The default value is FALSE .

Notes

If you enter a value for the storage space of the mirror repository volume that is too small, the controller firmware returns an error message, which states the amount of space that is needed for the mirror repository volume. The command does not try to change the mirror repository volume. You can re-enter the command by using the value from the error message for the storage space value of the mirror repository volume.

When you assign the drives, if you set the `trayLossProtect` parameter to **TRUE** and have selected more than one drive from any one tray, the storage array returns an error. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs operations, but the mirror repository volume that you create might not have tray loss protection.

When the controller firmware assigns the drives, if you set the `trayLossProtect` parameter to **TRUE**, the storage array returns an error if the controller firmware cannot provide drives that result in the new mirror repository volume having tray loss protection. If you set the `trayLossProtect` parameter to **FALSE**, the storage array performs the operation even if it means that the mirror repository volume might not have tray loss protection.

Minimum Firmware Level

6.10

7.10 adds RAID 6 capability

Re-create Snapshot

This command starts a fresh copy-on-write operation by using an existing snapshot volume.

Syntax

```
recreate snapshot (volume [volumeName] |
volumes [volumeName1 ... volumeNameN])
[userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryFullPolicy (failBaseWrites | failSnapShot)]
```

Parameters

Parameter	Description
volume or volumes	The name of the specific volume for which you want to start a fresh copy-on-write operation. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you must also enclose the volume name in double quotation marks (" ").
userLabel	The name of the snapshot volume. Enclose the snapshot volume name in double quotation marks (" "). If you enter more than one snapshot volume name, this command fails.
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the snapshot repository volume is nearing full. Use integer values. For example, a value of 70 means 70 percent. The default value is 50 .
repositoryFullPolicy	The type of processing that you want to continue if the snapshot repository volume is full. You can choose to fail writes to the base volume (failBaseWrites) or fail writes to the snapshot volume (failSnapShot). The default value is failSnapShot .

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

If you do not specify a value for the `warningThresholdPercent` parameter or the `repositoryFullPolicy` parameter, the previously set value is used.

Minimum Firmware Level

5.00

Re-create Snapshot Collection

This command restarts multiple snapshot volumes in one batch operation. This command makes sure that all of the snapshot volumes specified in the value are valid, and then restarts each snapshot volume. You can specify a single snapshot volume or a list of snapshot volumes.

Syntax

```
recreate snapshot collection (snapshotVolume [volumeName] | snapshotVolumes
[volumeName1 ... volumeNameN])
```

Parameters

Parameter	Description
<code>snapshotVolume</code> or <code>snapshotVolumes</code>	The name of the specific snapshot volume or snapshot volumes for which you want to initiate a restart. Enclose the snapshot volume name in square brackets ([]). If the snapshot volume name has special characters, you must also enclose the snapshot volume name in double quotation marks (" ").

Notes

If any one of the snapshot volumes does not pass the validation check, the command fails, and the snapshot volumes are not re-created.

Minimum Firmware Level

7.10

Remove Remote Mirror

This command removes the mirror relationship between the primary volume and the secondary volume in a remote-mirror pair.

Syntax

```
remove remoteMirror (localVolume [volumeName] |  
localVolumes [volumeName1 ... volumeNameN])
```

Parameters

Parameter	Description
localVolume or localVolumes	The name of the primary volume (the volume on the local storage array) that you want to remove. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Minimum Firmware Level

6.10

Remove Volume Copy

This command removes a volume copy pair.

Syntax

```
remove volumeCopy target [targetName]  
[source [sourceName]]
```

Parameters

Parameter	Description
target	The name of the target volume that you want to remove. Enclose the target volume name in square brackets ([]). If the target volume name has special characters, you also must enclose the target volume name in double quotation marks (" ").
source	The name of the source volume that you want to remove. Enclose the source volume name in square brackets ([]). If the source volume name has special characters, you also must enclose the source volume name in double quotation marks (" ").

Minimum Firmware Level

5.40

Remove Volume LUN Mapping

This command removes the logical unit number (LUN) mapping.

Syntax

```
remove (allVolumes | volume [volumeName] |  
volumes [volumeName1 ... volumeNameN] | accessVolume)  
lunMapping (host="hostName" |  
hostGroup="hostGroupName" | defaultGroup)
```

Parameters

Parameter	Description
allVolumes	This parameter removes the LUN mapping from all of the volumes.
volume or volumes	The name of the specific volume that you want to remove from the LUN mapping. You can enter more than one volume name. Enclose the volume name in double quotation marks (" ") inside of square brackets ([]).
accessVolume	This parameter removes the access volume.
host	The name of the host to which the volume is mapped. Enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group that contains the host to which the volume is mapped. Enclose the host group name in double quotation marks (" "). The defaultGroup value is the host group that contains the host to which the volume is mapped.

Notes

The access volume is the volume in a SAN environment that is used for communication between the storage management software and the storage array controller. The access volume uses a LUN address and consumes 20 MB of storage space that is not available for application data storage. An access volume is required only for in-band managed storage arrays.

ATTENTION **Removing an access volume can damage your configuration** – The agent uses the access volumes to communicate with a storage array. If you remove an access volume mapping for a storage array from a host that has an agent running on it, the storage management software is no longer able to manage the storage array through the agent.

You must use the `host` parameter and the `hostGroup` parameter when you specify a non-access volume or an access volume. The script engine ignores the `host` parameter or the `hostGroup` parameter when you use the `allVolumes` parameter or the `volumes` parameter.

Minimum Firmware Level

6.10

Repair Volume Parity

This command repairs the parity errors on a volume.

Syntax

```
repair volume [volumeName] parity  
parityErrorFile="filename"  
[verbose=(TRUE | FALSE)]
```

Parameters

Parameter	Description
volume	The name of the specific volume for which you want to repair parity. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").
parityErrorFile	The name of the file that contains the parity error information that you use to repair the errors. Enclose the file name in double quotation marks (" ").
verbose	The setting to capture progress details, such as percent complete, and to show the information as the volume parity is being repaired. To capture progress details, set this parameter to TRUE . To prevent capturing progress details, set this parameter to FALSE .

Minimum Firmware Level

6.10

Replace Drive

This command redefines the composition of a volume group. You can use this command to replace a drive with either an unassigned drive or a fully integrated hot spare.

Syntax

```
replace drive [trayID,slotID] replacementDrive=trayID,slotID
```

Parameters

Parameter	Description
drive	The tray and the slot of the drive that you want to replace. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID value and the slot ID value in square brackets ([]).
replacementDrive	The tray and the slot of the drive that you want to use as a replacement drive. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 .

Minimum Firmware Level

7.10

Reset Controller

This command resets a controller, and it is disruptive to I/O operations.

IMPORTANT When you reset a controller, the controller is removed from the data path and is not available for I/O operations until the reset operation is complete. If a host is using volumes that are owned by the controller being reset, the I/O directed to the controller is rejected. Before resetting the controller, either make sure that the volumes that are owned by the controller are not in use or make sure that a multi-path driver is installed on all of the hosts that use these volumes.

Syntax

```
reset controller [(a | b)]
```

Parameter

Parameter	Description
controller	The controller that you want to reset. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]). If you do not specify a controller owner, the controller firmware returns a syntax error.

Notes

The controller that receives the reset controller command resets the controller specified. For example, if the reset controller command is sent to controller A to request a reset of controller A, then controller A reboots itself by doing a soft reboot. If the reset controller command is sent to controller A to request a reset of controller B, then controller A holds controller B in reset and then releases controller B from reset, which is a hard reboot. A soft reboot in some products only resets the IOC chip. A hard reboot resets both the IOC and the expander chips in the controller.

Minimum Firmware Level

5.20

Reset Storage Array Battery Install Date

This command resets the age of the batteries in a storage array to zero days. You can reset the age of the batteries for an entire storage array or the age of a battery in a specific controller or in a specific battery pack.

Syntax

```
reset storageArray batteryInstallDate
(controller=[(a | b)] | batteryPack [left | right])
```

Parameters

Parameter	Description
controller	The controller that contains the battery for which you want to reset the age. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Use the <code>controller</code> parameter only for controllers with batteries.
batteryPack	The battery pack contains both a left battery and a right battery. Valid identifiers are left or right , where left is the battery that supports the controller in slot A, and right is the battery that supports the controller in slot B. Use the <code>batteryPack</code> parameter only for controller trays with battery packs.

Notes

A controller might have a battery associated with it, so the controller is identified as either **a** or **b**. With the release of the CE7900 controller tray, battery packs inside the interconnect-battery canister are identified as either **left** or **right**. If the command statement uses the wrong parameter, an error appears.

Minimum Firmware Level

6.10

Reset Storage Array Diagnostic Data

This command resets the NVSRAM that contains the diagnostic data for the storage array. This command does not delete the diagnostic data. This command replaces the Needs Attention status with the Diagnostic Data Available status. The old diagnostic data is written over automatically when new data is captured. The memory that contains the diagnostic data is also cleared when the controllers reboot. Before resetting the diagnostic data, use the `save storageArray diagnosticData` command to save the diagnostic data to a file.

IMPORTANT Run this command only with the assistance of a Customer and Technical Support representative.

Syntax

```
reset storageArray diagnosticData
```

Parameters

None.

Minimum Firmware Level

6.16

Reset Storage Array RLS Baseline

This command resets the Read Link Status (RLS) baseline for all devices by setting all of the RLS counts to 0.

Syntax

```
reset storageArray RLSBaseline
```

Parameters

None.

Minimum Firmware Level

5.00

Reset Storage Array SOC Baseline

This command resets the baseline for all switch-on-a-chip (SOC) devices that are accessed through the controllers. This command resets the baseline by setting all of the SOC counts to 0. This command is valid only for Fibre Channel devices in an Arbitrated Loop topology.

Syntax

```
reset storageArray SOCBaseline
```

Parameters

None.

Minimum Firmware Level

6.16

Reset Storage Array Volume Distribution

This command reassigns (moves) all of the volumes to their preferred controller.

Syntax

```
reset storageArray volumeDistribution
```

Parameters

None.

Notes

If you use this command on a host without a multi-path driver, to prevent application errors, you must stop I/O operations to the volumes until this command has completed.

Under certain host operating system environments, you might be required to reconfigure the multi-path host driver. You might also need to make operating system modifications to recognize the new I/O path to the volumes.

Minimum Firmware Level

5.20

Resume Remote Mirror

This command resumes a suspended Remote Volume Mirroring operation.

Syntax

```
resume remoteMirror (primary [volumeName] |
primaries [volumeName1 ... volumeNameN])
[writeConsistency=(TRUE | FALSE) }
```

Parameters

Parameter	Description
primary or primaries	The name of the primary volume for which you want to resume operation. You can enter more than one primary volume name. Enclose the primary volume name in square brackets ([]). If the primary volume name has special characters, you also must enclose the primary volume name in double quotation marks (" ").
writeConsistency	The setting to identify the volumes in this command that are in a write-consistency group or are separate. For the volumes to be in the same write-consistency group, set this parameter to TRUE . For the volumes to be separate, set this parameter to FALSE .

Notes

If you set the `writeConsistency` parameter to **TRUE**, the volumes must be in a write-consistency group (or groups). This command resumes all write-consistency groups that contain the volumes. For example, if volumes A, B, and C are in a write-consistency group and they have remote counterparts A', B', and C', the `resume remoteMirror volume ["A"] writeConsistency=TRUE` command resumes A-A', B-B', and C-C'.

Minimum Firmware Level

6.10

Revive Drive

This command forces the specified drive to the Optimal state.

ATTENTION Possible loss of data access – Correct use of this command depends on the data configuration on all of the drives in the volume group. Never try to revive a drive unless you are supervised by a Customer and Technical Support representative.

Syntax

```
revive drive [trayID,slotID]
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID value and the slot ID value in square brackets ([]).

Minimum Firmware Level

5.43

Revive Volume Group

This command forces the specified volume group and its associated failed drives to the Optimal state.

ATTENTION Possible loss of data access – Correct use of this command depends on the data configuration on all of the drives in the volume group. Never try to revive a drive unless you are supervised by a Customer and Technical Support representative.

Syntax

```
revive volumeGroup [volumeGroupName]
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) of the volume group to be set to the Optimal state. Enclose the volume group identifier in square brackets ([]).

Minimum Firmware Level

6.10

Save Controller NVSRAM

This command saves a copy of the controller NVSRAM values to a file. This command saves all of the regions.

Syntax

```
save controller [(a | b)]  
NVSRAM file="filename"
```

Parameters

Parameter	Description
controller	The controller with the NVSRAM values that you want to save. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]).
file	The name of the file in which you want to save the values. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Save Drive Channel Fault Isolation Diagnostic Status

This command saves the drive channel fault isolation diagnostic data that is returned from the `start driveChannel faultDiagnostics` command. You can save the diagnostic data to a file as standard text or as XML.

Syntax

```
save driveChannel faultDiagnostics file="filename"
```

Parameter

Parameter	Description
file	The name of the file in which you are storing the results of the fault isolation diagnostics test on the drive channel. Enclose the name in double quotation marks (" ").

Notes

A file extension is not automatically appended to the saved file. You must specify the applicable format suffix for the file. If you specify a file extension of `.txt`, then the output will be in a text file format. If you specify a file extension of `.xml`, then the output will be in an XML file format.

Save Drive Log

This command saves the log sense data to a file. Log sense data is maintained by the storage array for each drive.

Syntax

```
save allDrives logFile="filename"
```

Parameter

Parameter	Description
logFile	The name of the file to which you want to write the log sense data. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Save Storage Array Configuration

This command creates a script file that you can use to create the current storage array volume configuration.

Syntax

```
save storageArray configuration file="filename"
[(allconfig | globalSettings=(TRUE | FALSE)
volumeConfigAndSettings=(TRUE | FALSE)
hostTopology=(TRUE | FALSE)
lunMappings=(TRUE | FALSE))]
```

Parameters

Parameter	Description
file	The name of the file that contains the configuration values. Enclose the file name in double quotation marks (" ").
allConfig	The setting to save all of the configuration values to the file. (If you choose this parameter, all of the configuration parameters are set to TRUE .)
globalSettings	The setting to save the global settings to the file. To save the global settings, set this parameter to TRUE . To prevent saving the global settings, set this parameter to FALSE . The default value is TRUE .

Parameter	Description
volumeConfigAndSettings	The setting to save the volume configuration settings and all of the global settings to the file. To save the volume configuration settings and global settings, set this parameter to TRUE . To prevent saving the volume configuration settings and global settings, set this parameter to FALSE . The default value is TRUE .
hostTopology	The setting to save the host topology to the file. To save the host topology, set this parameter to TRUE . To prevent saving the host topology, set this parameter to FALSE . The default value is FALSE .
lunMappings	The setting to save the LUN mapping to the file. To save the LUN mapping, set this parameter to TRUE . To prevent saving the LUN mapping, set this parameter to FALSE . The default value is FALSE .

Notes

When you use this command, you can specify any combination of the parameters for the global setting, the volume configuration setting, the host topology, or the LUN mapping. If you want to enter all settings, use the `allConfig` parameter. The parameters are all optional.

Minimum Firmware Level

6.10

Save Storage Array Diagnostic Data

This command saves the storage array diagnostic data from either the controllers or the environmental services monitors (ESMs) to a file. You can review the file contents at a later time. You can also send the file to a Customer and Technical Support representative for further review.

After you have saved the diagnostic data, you can reset the NVSRAM registers that contain the diagnostic data so that the old data can be overwritten. Use the `reset storageArray diagnosticData` command to reset the diagnostic data registers.

IMPORTANT Run this command only with the assistance of a Customer and Technical Support representative.

Syntax

```
save storageArray diagnosticData [(controller | esm)]
file="filename"
```

Parameters

Parameter	Description
diagnosticData	This parameter allows you to download the diagnostic data from either the controllers or the ESMs.
file	The name of the file to which you want to save the diagnostic data. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.16

Save Storage Array Events

This command saves events from the Major Event Log to a file. You can save all of the events or only the critical events.

Syntax

```
save storageArray (allEvents | criticalEvents)
file="filename"
[count=numberOfEvents]
```

Parameters

Parameter	Description
file	The name of the file to which you want to save the events. Enclose the file name in double quotation marks (" ").
count	The number of events or critical events that you want to save to a file. If you do not enter a value for the count, all events or all critical events are saved to the file. If you enter a value for the count, only that number of events or critical events (starting with the last event entered) are saved to the file. Use integer values.

Notes

You have the option to save all events (**allEvents**) or only the critical events (**criticalEvents**).

Minimum Firmware Level

6.10

Save Storage Array Performance Statistics

This command saves the performance statistics to a file. Before you use this command, run the `set session performanceMonitorInterval` command and the `set session performanceMonitorIterations` command to specify how often statistics are collected.

Syntax

```
save storageArray performanceStats file="filename"
```

Parameter

Parameter	Description
file	The name of the file to which you want to save the performance statistics. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Save Storage Array RLS Counts

This command saves the read link status (RLS) counters to a file.

Syntax

```
save storageArray RLSCounts file="filename"
```

Parameter

Parameter	Description
file	The name of the file to which you want to save the RLS counters. Enclose the file name in double quotation marks (" ").

Notes

To more effectively save RLS counters to a file, perform these steps:

- 1 Run the `reset storageArray RLSBaseline` command to set all of the RLS counters to 0.
- 2 Run the storage array for a predetermined amount of time (for instance, two hours).
- 3 Run the `save storageArray RLSCounts file="filename"` command.

The default name of the file that contains the RLS counts is `readLinkStatus.csv`. You can use any file name if you use the `.csv` extension.

Minimum Firmware Level

6.10

Save Storage Array SOC Counts

This command saves the SOC error statistics to a file. This command is valid only for Fibre Channel devices in an Arbitrated Loop topology.

Syntax

```
save storageArray SOCCounts file="filename"
```

Parameter

Parameter	Description
file	The name of the file to which you want to save the SOC error statistics. Enclose the file name in double quotation marks (" ").

Notes

To more effectively save SOC error statistics to a file, perform these steps:

- 1 Run the `reset storageArray SOCBaseline` command to set all of the SOC counters to 0.
- 2 Run the storage array for a predetermined amount of time (for example, two hours).
- 3 Run the `save storageArray SOCCounts file="filename"` command.

The default name of the file that contains the SOC error statistics is `socStatistics.csv`. You can use any file name if you use the `.csv` extension.

Minimum Firmware Level

6.16

Save Storage Array State Capture

This command saves the state capture of a storage array to a file.

Syntax

```
save storageArray stateCapture file="filename"
```

Parameter

Parameter	Description
file	The name of the file to which you want to save the state capture. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Save Storage Array Support Data

This command saves the support-related information of the storage array to a file. Support-related information includes these items:

- The storage array profile
- The Major Event Log information
- The read link status (RLS) data
- The NVSRAM data
- Current problems and associated recovery information
- The performance statistics for the entire storage array
- The persistent registration information and the persistent reservation information
- Detailed information about the current status of the storage array
- The diagnostic data for the drive
- A recovery profile for the storage array

- The unreadable sectors that are detected on the storage array
- The state capture data

Syntax

```
save storageArray supportData file="filename"
```

Parameter

Parameter	Description
file	The name of the file to which you want to save the support-related data for the storage array. Enclose the file name in double quotation marks (" ").

Minimum Firmware Level

6.10

Set Controller

This command defines the attributes for the controllers.

Syntax

```
set controller [(a | b)]
availability=(online | offline | serviceMode) |
ethernetPort [(1 | 2)] ethernetPortOptions
| globalNVSramByte [nvsramOffset]=(nvsramByteSetting | nvsramBitSetting) |
hostNVSramByte [hostType, nvsramOffset]=(nvsramByteSetting |
nvsramBitSetting) |
rloginEnabled=(TRUE | FALSE) |
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
controller	The controller for which you want to define properties. Valid identifiers for the controller are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the identifier for the controller in square brackets ([]). If you do not specify a controller, the firmware for the controller returns a syntax error.
availability	The mode for the controller, which you can set to online , offline , or serviceMode (service).
ethernetPort	The attributes (options) for the management Ethernet ports. The entries to support this parameter are listed in the “Syntax Element Statement Data” table that follows. Many settings are possible, including setting the IP address, the gateway address, and the subnet mask address.
globalNVSARAMByte	A portion of the controller NVSRAM. Specify the region to be modified using the starting byte offset within the region, and the size and value of the new data to be stored into NVSRAM.
hostNVSARAMByte	The NVSRAM for the host-specific region. The setting specifies the host index for the specific host, the starting offset within the region, the number of bytes, and the values to be written.
rloginEnabled	The setting for whether the remote login feature is turned on or turned off. To turn on the remote login feature, set this parameter to TRUE . To turn off the remote login feature, set this parameter to FALSE .
serviceAllowedIndicator	The setting for whether the Service Action Allowed indicator light is turned on or turned off. To turn on the Service Action Allowed indicator light, set this parameter to on . To turn off the Service Action Allowed indicator light, set this parameter to off .

Syntax Element Statement Data

Options for the ethernetPort Parameter
IPv4Address=(0-255).(0-255).(0-255).(0-255)
IPv4ConfigurationMethod=[(static dhcp)]
IPv4GatewayIP=(0-255).(0-255).(0-255).(0-255)
IPv4SubnetMask=(0-255).(0-255).(0-255).(0-255)

Notes

When you use this command, you can specify one or more of the parameters. You do not need to use all of the parameters.

Setting the availability parameter to **serviceMode** causes the alternate controller to take ownership of all of the volumes. The specified controller no longer has any volumes and refuses to take ownership of any more volumes. Service mode is persistent across reset cycles and power cycles until the availability parameter is set to **online**.

Use the `show controller NVSRAM` command to show the NVSRAM information.

The default port value for the `tcpListeningPort` parameter is **3260**.

Set Controller Service Action Allowed Indicator

This command turns on or turns off the Service Action Allowed indicator light on a controller in a controller tray or a controller-drive tray. If the storage array does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage array supports the command but is unable to turn on or turn off the indicator light, this command returns an error. (To turn on or turn off the Service Action Allowed indicator light on the power-fan canister or the interconnect-battery canister, use the `set tray serviceAllowedIndicator` command.)

Syntax

```
set controller=[(a | b)]
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
controller	The controller that has the Service Action Allowed indicator light that you want to turn on or turn off. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]). If you do not specify a controller, the controller firmware returns a syntax error.
serviceAllowedIndicator	The setting to turn on or turn off the Service Action Allowed indicator light. To turn on the Service Action Allowed indicator light, set this parameter to on . To turn off the Service Action Allowed indicator light, set this parameter to off .

Notes

The IS4000 controllers support this command.

Minimum Firmware Level

6.14

Set Drive Channel Status

This command defines how the drive channel performs.

Syntax

```
set driveChannel [(1 | 2 | 3 | 4)]
status=(optimal | degraded)
```

Parameters

Parameter	Description
driveChannel	The identifier number of the drive channel for which you want to set the status. Valid drive channel values are 1 , 2 , 3 , or 4 . Enclose the drive channel number in square brackets ([]).
status	The condition of the drive channel. You can set the drive channel status to optimal or degraded .

Notes

Use the **optimal** value to move a degraded drive channel back to the Optimal state. Use the **degraded** value when the drive channel is experiencing problems, and the storage array requires additional time for data transfers.

Minimum Firmware Level

6.10

Set Drive Hot Spare

This command assigns or unassigns one or more drives as a hot spare.

Syntax

```
set (drive [trayID,slotID] |  
drives [trayID1,slotID1 ... trayIDn,slotIDn])  
hotSpare=(TRUE | FALSE)
```

Parameters

Parameter	Description
drive or drives	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in square brackets ([]).
hotSpare	The setting to assign the drive as the hot spare. To assign the drive as the hot spare, set this parameter to TRUE . To remove a hot spare assignment from a drive, set this parameter to FALSE .

Minimum Firmware Level

6.10

Set Drive Service Action Allowed Indicator

This command turns on or turns off the Service Action Allowed indicator light on a drive in drive trays that support the Service Action Allowed indicator light feature. If the storage array does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage array supports the command but is unable to turn on or turn off the indicator light, this command returns an error.

Syntax

```
set (drive [trayID,slotID] |
drives [trayID1,slotID1 ... trayIDn,slotIDn])
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
drive or drives	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in square brackets ([]).
serviceAllowedIndicator	The setting to turn on or turn off the Service Action Allowed indicator light. To turn on the Service Action Allowed indicator light, set this parameter to on . To turn off the Service Action Allowed indicator light, set this parameter to off .

Notes

None.

Minimum Firmware Level

6.16

Set Drive State

This command sets a drive to the Failed state. (To return a drive to the Optimal state, use the `revive drive` command.)

Syntax

```
set drive [trayID,slotID] operationalState=failed
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in square brackets ([]).

Minimum Firmware Level

5.20

Set Foreign Drive to Native

A drive is considered to be native when it is a part of the storage array's volume group configuration. A drive is considered to be foreign when it does not belong to the storage array's volume group configuration or when it fails to be imported with the drives of a volume group that are transferred to a new storage array. The latter failure creates an incomplete volume group on the new storage array.

Run this command to add the missing (foreign) drives back into their original volume group and to make them part of the new storage array volume group configuration of the new storage array.

Use this operation for emergency recovery only: when one or more drives need to be changed from a foreign drive status and returned to a native status within their original volume group.

ATTENTION **Possible data corruption or data loss** – Using this command for reasons other than what is stated previously might result in data loss without notification.

Syntax

```
set (drive [trayID, slotID] | allDrives) toNative
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in square brackets ([]).
allDrives	The setting to select all of the drives.

Minimum Firmware Level

7.10

Set Host

This command assigns a host to a host group or moves a host to a different host group. You can also create a new host group and assign the host to the new host group with this command. The actions performed by this command depend on whether the host has individual volume-to-LUN mappings or does not have individual volume-to-LUN mappings.

Syntax

```
set host [hostName]
hostGroup= ("hostGroupName" | none | defaultGroup)
userLabel= "newHostName"
hostType= (hostTypeIndexLabel | hostTypeIndexNumber)
```

Parameters

Parameter	Description
host	The name of the host that you want to assign to a host group. Enclose the host name in square brackets ([]). If the host name has special characters, you also must enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group to which you want to assign the host. (The following table defines how the command runs if the host does or does not have individual volume-to-LUN mappings.) Enclose the host group name in double quotation marks (" "). The defaultGroup option is the host group that contains the host to which the volume is mapped.
userLabel	The new host name. Enclose the host name in double quotation marks (" ").
hostType	The index label or number of the host type for the HBA host port. Use the <code>show storageArray hostTypeTable</code> command to generate a list of available host type identifiers. If the host type has special characters, enclose the host type in double quotation marks (" ").

Host Group Parameter	Host Has Individual Volume-to-LUN Mappings	Host Does Not Have Individual Volume-to-LUN Mappings
hostGroupName	The host is removed from the present host group and is placed under the new host group defined by <i>hostGroupName</i> .	The host is removed from the present host group and is placed under the new host group defined by <i>hostGroupName</i> .
none	The host is removed from the host group as an independent partition and is placed under the root node.	The host is removed from the present host group and is placed under the default group.
defaultGroup	The command fails.	The host is removed from the present host group and is placed under the default group.

Notes

When you use this command, you can specify one or more of the optional parameters.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

The CHAP secret must be between 12 characters and 57 characters. [Table 25](#) lists the valid characters.

Table 25 Valid Characters for a CHAP Secret

Space	!	“	#	\$	%	&	'	()	*	+
,	-	.	/	0	1	2	3	4	5	6	7
8	9	:	;	<	=	>	?	@	A	B	C
D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	[
\]	^	_	'	a	b	c	d	e	f	g
h	i	j	k	l	m	n	o	p	q	r	s
t	u	v	w	x	y	z	{		}	~	

Minimum Firmware Level
6.10

Set Host Channel

This command defines the loop ID for the host channel.

Syntax

```
set hostChannel [hostChannelNumber]  
preferredID=portID
```

Parameters

Parameter	Description
hostChannel	The identifier number of the host channel for which you want to set the loop ID. Enclose the host channel identifier number in square brackets ([]). Valid host channel values are a1 , a2 , b1 , or b2 . Valid host channel values are a1 , a2 , a3 , a4 , b1 , b2 , b3 , or b4 .
preferredID	The new port ID for the specified host channel. Port ID values are 0 to 127 .

Minimum Firmware Level

6.10

Set Host Group

This command renames a host group.

Syntax

```
set hostGroup [hostGroupName]  
userLabel="newHostGroupName"
```

Parameters

Parameter	Description
hostGroup	The name of the host group that you want to rename. Enclose the host group name in square brackets ([]). If the host group name has special characters, you also must enclose the host group name in double quotation marks (" ").
userLabel	The new name for the host group. Enclose the new host group name in double quotation marks (" ").

Notes

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

6.10

Set Host Port

This command changes the host type for an HBA host port. You can also change an HBA host port label with this command.

Syntax

```
set hostPort [portLabel] host="hostName" userLabel="newPortLabel"
```

Parameters

Parameter	Description
hostPort	The name of the HBA host port for which you want to change the host type, or for which you want to create a new name. Enclose the HBA host port name in square brackets ([]). If the HBA host port label has special characters, enclose the HBA host port label in double quotation marks (" ").
host	The name of the host to which the HBA host port is connected. Enclose the host name in double quotation marks (" ").
userLabel	The new name that you want to give to the HBA host port. Enclose the new name of the HBA host port in double quotation marks (" ").

Notes

When you use this command, you can specify one or more of the optional parameters.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

Minimum Firmware Level

6.10

Set Remote Mirror

This command defines the properties for a remote-mirror pair.

Syntax

```
set remoteMirror (localVolume [volumeName] |
localVolumes [volumeName1 ... volumeNameN])
role=(primary | secondary)
[force=(TRUE | FALSE)]
syncPriority=(highest | high | medium | low | lowest)
autoResync=(enabled | disabled)
writeOrder=(preserved | notPreserved)
writeMode=(synchronous | asynchronous)
```

Parameters

Parameter	Description
localVolume or localVolumes	The name of the primary volume for which you want to define properties. You can enter more than one primary volume name. Enclose the primary volume name in square brackets ([]). If the primary volume name has special characters, you also must enclose the primary volume name in double quotation marks (" ").
role	The setting for the volume to act as the primary volume or the secondary volume. To define the volume as the primary volume, set this parameter to primary . To define the volume as the secondary volume, set this parameter to secondary . This parameter applies only when the volume is part of a mirror relationship.
force	The role reversal is forced if the communications link between the storage arrays is down and promotion or demotion on the local side results in a dual-primary condition or a dual-secondary condition. To force a role reversal, set this parameter to TRUE . The default value is FALSE .
syncPriority	The priority that full synchronization has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .

Parameter	Description
autoResync	The settings for automatic resynchronization between the primary volumes and the secondary volumes of a remote-mirror pair. This parameter has these values: <ul style="list-style-type: none"> • enabled – Automatic resynchronization is turned on. You do not need to do anything further to resynchronize the primary volume and the secondary volume. • disabled – Automatic resynchronization is turned off. To resynchronize the primary volumes and the secondary volume, you must run the <code>resume remoteMirror</code> command.
writeOrder	This parameter defines write order for data transmission between the primary volume and the secondary volume. Valid values are preserved or notPreserved .
writeMode	This parameter defines how the primary volume writes to the secondary volume. Valid values are synchronous or asynchronous .

Notes

When you use this command, you can specify one or more of the optional parameters.

Synchronization priority defines the amount of system resources that are used to synchronize the data between the primary volumes and the secondary volumes of a mirror relationship. If you select the highest priority level, the data synchronization uses the most system resources to perform the full synchronization, which decreases the performance for host data transfers.

The `writeOrder` parameter applies only to asynchronous mirrors and makes them become part of a consistency group. Setting the `writeOrder` parameter to **preserved** causes the remote-mirror pair to transmit data from the primary volume to the secondary volume in the same order as the host writes to the primary volume. In the event of a transmission link failure, the data is buffered until a full synchronization can occur. This action can require additional system overhead to maintain the buffered data, which slows operations. Setting the `writeOrder` parameter to **notPreserved** frees the system from having to maintain data in a buffer, but it requires forcing a full synchronization to make sure that the secondary volume has the same data as the primary volume.

Minimum Firmware Level

6.10

Set Session

This command defines how you want the current script engine session to run.

Syntax

```
set session errorAction=(stop | continue)
password=" storageArrayPassword"
performanceMonitorInterval=intervalValueperformanceMonitorIterations=iterationValue
```

Parameters

Parameter	Description
errorAction	How the session responds if an error is encountered during processing. You can choose to stop the session if an error is encountered, or you can continue the session after encountering an error. The default value is stop . (This parameter defines the action for execution errors, not syntax errors. Some error conditions might override the continue value.)

Parameter	Description
password	The password for the storage array. Enclose the password in double quotation marks (" ").
performanceMonitorInterval	The frequency of gathering performance data. Enter an integer value for the polling interval, in seconds, for which you want to capture data. The range of values is 3 to 3600 seconds. The default value is 5 seconds.
performanceMonitorIterations	The number of samples to capture. Enter an integer value. The range of values for samples captured is 1 to 3600 . The default value is 5 .

Notes

When you use this command, you can specify one or more of the optional parameters.

Passwords are stored on each storage array in a management domain. If a password was not previously set, you do not need a password. The password can be any combination of alphanumeric characters with a maximum of 30 characters. (You can define a storage array password by using the `set storageArray` command.)

The polling interval and the number of iterations that you specify remain in effect until you end the session. After you end the session, the polling interval and the number of iterations return to their default values.

Minimum Firmware Level

5.20

Set Snapshot Volume

This command defines the properties for a snapshot volume and lets you rename a snapshot volume.

Syntax

```
set (volume [volumeName] |
volumes [volumeName1 ... volumeNameN])
userLabel="snapshotVolumeName"
warningThresholdPercent=percentValue
repositoryFullPolicy=(failBaseWrites | failSnapShot)
```

Parameters

Parameter	Description
volume or volumes	The name of the specific snapshot volume for which you want to define properties. (You can enter more than one volume name if you use the <code>volumes</code> parameter). Enclose the snapshot volume name in double quotation marks (“ ”) inside of square brackets ([]).
userLabel	A new name that you want to give to a snapshot volume. Enclose the new snapshot volume name in double quotation marks (“ ”).
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the snapshot repository is nearing full. Use integer values. For example, a value of 70 means 70 percent. The default value is 50 .
repositoryFullPolicy	How you want snapshot processing to continue if the snapshot repository volume is full. You can choose to fail writes to the base volume (failBaseWrites) or fail writes to the snapshot volume (failSnapShot). The default value is failSnapShot .

Notes

When you use this command, you can specify one or more of the optional parameters.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

You can set the `warningThresholdPercent` parameter and the `repositoryFullPolicy` parameter for both the snapshot repository volume or the snapshot volume.

Minimum Firmware Level

6.10

Set Storage Array

This command defines the properties of the storage array.

Syntax

```
set storageArray {alarm=(enable | disable | mute) |
cacheBlockSize=cacheBlockSizeValue |
cacheFlushStart=cacheFlushStartSize |
cacheFlushStop=cacheFlushStopSize |
defaultHostType=("hostTypeName" | hostTypeIdentifier)
failoverAlertDelay=delayValue |
mediaScanRate=(disabled | 1-30) |
password="password" |
userLabel="storageArrayName" }
```

Parameters

Parameter	Description
alarm	<p>The setting for the audible alarm. This parameter has these values:</p> <ul style="list-style-type: none"> • enable – The audible alarm is turned on and sounds if a fault occurs. • disable – The audible alarm is turned off and does not sound if a fault occurs. • mute – The audible alarm is turned off if it is sounding. If another fault occurs after you set the audible alarm to mute, the audible alarm sounds again.
cacheBlockSize	<p>The cache block size that is used by the controller for managing the cache. Valid values are 4 (4 KB), 8 (8 KB), 16 (16 KB), or 32 (32 KB).</p>
cacheFlushStart	<p>The percentage of unwritten data in the cache that causes a cache flush. Use integer values from 0 to 100 to define the percentage. The default value is 80.</p>
cacheFlushStop	<p>The percentage of unwritten data in the cache that stops a cache flush in progress. Use integer values from 0 to 100 to define the percentage. This value must be less than the value of the <code>cacheFlushStart</code> parameter.</p>
defaultHostType	<p>The default host type of any unconfigured HBA host port to which the controllers are connected. To generate a list of valid host types for the storage array, run the <code>show storageArray hostTypeTable</code> command. Host types are identified by a name or a numerical index. Enclose the host type name in double quotation marks (“ ”). Do not enclose the host type numerical identifier in double quotation marks.</p>
failoverAlertDelay	<p>The failover alert delay time in minutes. The valid values for the delay time are 0 to 60 minutes. The default value is 5.</p>

Parameter	Description
mediaScanRate	The number of days over which the media scan runs. Valid values are disabled , which turns off the media scan, or 1 day to 30 days, where 1 day is the fastest scan rate, and 30 days is the slowest scan rate. A value other than disabled or 1 to 30 does not allow the media scan to function.
password	The password for the storage array. Enclose the password in double quotation marks (" ").
userLabel	The name for the storage array. Enclose the storage array name in double quotation marks (" ").

Notes

When you use this command, you can specify one or more of the optional parameters.

Cache Block Size

When you define cache block sizes, use the 4-KB cache block size for storage arrays that require I/O streams that are typically small and random. Use the 8-KB cache block size when the majority of your I/O streams are larger than 4 KB but smaller than 8 KB. Use the 16-KB cache block size or the 32-KB cache block size for storage arrays that require large data transfer, sequential, or high-bandwidth applications.

This parameter defines the supported cache block size for all of the volumes in the storage array. Not all controller types support all cache block sizes. For redundant configurations, this parameter includes all of the volumes that are owned by both controllers within the storage array.

Cache Flush Start and Cache Flush Stop

When you define values to start a cache flush, a value that is too low increases the chance that data needed for a host read is not in the cache. A low value also increases the number of drive writes that are necessary to maintain the cache level, which increases system overhead and decreases performance.

When setting storage array cache settings, the value of the `cacheFlushStart` parameter must always be greater than or equal to the value of the `cacheFlushStop` parameter. For example, if the value of the `cacheFlushStart` parameter is set to **80**, you may set the value of the `cacheFlushStop` parameter within the range of **0** to **80**.

When you define values to stop a cache flush, the lower the value, the higher the chance that the data for a host read requires a drive read rather than reading from the cache.

Default Host Type

When you define host types, if SANshare Storage Partitioning is enabled, the default host type affects only those volumes that are mapped in the default group. If SANshare Storage Partitioning is not enabled, all of the hosts that are attached to the storage array must run the same operating system and be compatible with the default host type.

Media Scan Rate

Media scan runs on all of the volumes in the storage array that have Optimal status, do not have modification operations in progress, and have the `mediaScanRate` parameter enabled. Use the `set volume` command to enable or disable the `mediaScanRate` parameter.

Password

Passwords are stored on each storage array. The password can be any combination of alphanumeric characters with a maximum of 30 characters.

Minimum Firmware Level

5.00 adds the `defaultHostType` parameter.

5.40 adds the `failoverAlertDelay` parameter.

6.14 adds the `alarm` parameter.

Set Storage Array Redundancy Mode

This command sets the redundancy mode of the storage array to either simplex or duplex.

Syntax

```
set storageArray redundancyMode=(simplex | duplex)
```

Parameter

Parameter	Description
redundancyMode	Use simplex mode when you have a single controller. Use duplex mode when you have two controllers.

Minimum Firmware Level

6.10

Set Storage Array Time

This command sets the clocks on both controllers in a storage array by synchronizing the controller clocks with the clock of the host from which you run this command.

Syntax

```
set storageArray time
```

Parameters

None.

Minimum Firmware Level

6.10

Set Storage Array Tray Positions

This command defines the position of the trays in a storage array. You must include all of the trays in the storage array when you enter this command.

Syntax

```
set storageArray trayPositions=(controller | trayID-list)
```

Parameter

Parameter	Description
trayPositions	A list of all of the tray IDs. The sequence of the tray IDs in the list defines the positions for the controller tray and the drive trays in a storage array. Valid values are 0 to 99 . Enter the tray ID values separated by a space. Enclose the list of tray ID values in parentheses. For storage arrays where the controller tray has a predefined identifier that is not in the range of valid tray position values, use the controller value.

Notes

This command defines the position of a tray in a storage array by the position of the tray ID in the `trayPositions` list. For example, if you have a controller tray with an ID set to **84** and drive trays with IDs set to **1**, **12**, and **50**, the `trayPositions` sequence (**84 1 12 50**) places the controller tray in the first position, drive tray **1** in the second position, drive tray **12** in the third position, and drive tray **50** in the fourth position. The `trayPositions` sequence (**1 84 50 12**) places the controller tray in the second position, drive tray **1** in the first position, drive tray **50** in the third position, and drive tray **12** in the fourth position.

Minimum Firmware Level

6.10

For 6.14 and 6.16, **controller** is not a valid value.

Set Tray Alarm

This command turns on, turns off, or mutes the audible alarm for a specific tray or all of the trays in a storage array.

Syntax

```
set (allTrays | tray [trayID]
alarm=(enable | disable | mute))
```

Parameters

Parameter	Description
allTrays	The setting to select all of the trays in a storage array that have audible alarms that you want to turn on, turn off, or mute.
tray	The specific tray that has the audible alarm that you want to turn on, turn off, or mute. Tray ID values are 0 to 99 . Enclose the tray ID value in square brackets ([]).
alarm	The setting for the audible alarm. This alarm has these values: <ul style="list-style-type: none"> • enable – The audible alarm is turned on and sounds if a fault occurs. • disable – The audible alarm is turned off and does not sound if a fault occurs. • mute – The audible alarm is turned off if it is sounding. (If another fault occurs after you set the audible alarm to mute , the audible alarm sounds again.)

Minimum Firmware Level

6.16

Set Tray Identification

This command sets the tray ID of a controller tray, controller-drive tray, or a drive tray in a storage array. This command is valid only for controller trays, controller-drive tray, or drive trays that have tray IDs that you can set through the controller firmware. You cannot use this command for controller trays, controller-drive tray, or drive trays that have a tray ID that you set with a switch.

Syntax

```
set tray ["serial-number"] id=trayID
```

Parameters

Parameter	Description
tray	The serial number of the controller tray, controller-drive tray, or the drive tray for which you are setting the tray ID. Serial numbers can be any combination of alphanumeric characters and any length. Enclose the serial number in double quotation marks (" ").
id	The value for the controller tray tray ID, controller-drive tray tray ID, or the drive tray tray ID. Tray ID values are 00 through 99 . You do not need to enclose the tray ID value in parentheses.

Minimum Firmware Level

6.16 adds support for controller trays, controller-drive trays, and drive trays that have tray IDs set through the controller firmware.

Set Tray Service Action Allowed Indicator

This command turns on or turns off the Service Action Allowed indicator light on a power-fan canister, an interconnect-battery canister, or an environmental services monitor (ESM) canister. If the storage array does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage array supports the command but is unable to turn on or turn off the indicator light, this command returns an error.

To turn on or turn off the Service Action Allowed indicator light on the controller canister, use the `set controller serviceAllowedIndicator` command.

Syntax

```
set tray [trayID]
(powerFan [(left | right)] |
interconnect |
battery [(left | right)] |
esm [(left | right)])
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
tray	The tray where the power-fan canister, the interconnect-battery canister, or the ESM canister resides. Tray ID values are 0 to 99 . Enclose the tray ID value in square brackets ([]). If you do not enter a tray ID value, the tray ID of the controller tray is the default value.
powerFan	The Service Action Allowed indicator light on the power-fan canister that you want to turn on or turn off. Valid power-fan canister identifiers are left or right . Enclose the power-fan canister identifier in square brackets ([]).
interconnect	The Service Action Allowed indicator light for the interconnect-battery canister.

Parameter	Description
battery	The Service Action Allowed indicator light for a battery. Valid battery identifiers are left or right .
esm	The Service Action Allowed indicator light for an ESM canister. Valid ESM canister identifiers are left or right . (The values left and right are with respect to the rear of the drive tray.)
serviceAllowedIndicator	The setting to turn on or turn off the Service Action Allowed indicator light. To turn on the Service Action Allowed indicator light, set this parameter to on . To turn off the Service Action Allowed indicator light, set this parameter to off .

Example

This command turns on the Service Action Allowed indicator light for the left ESM on tray 5 with the IP address of 155.155.155.155.

```
smcli 155.155.155.155 -c "set tray [5] ESM [left]
serviceAllowedIndicator=on";
```

Minimum Firmware Level

6.14 adds these parameters:

- powerFan
- interconnect

6.16 adds these parameters:

- tray
- esm

Set Volume

This command defines the properties for a volume. You can use most parameters to define properties for one or more volumes. You also can use some parameters to define properties for only one volume. The syntax definitions are separated to show which parameters apply to several volumes and which apply to only one volume. Also, the syntax for volume mapping is listed separately.

Syntax Applicable to One or More Volumes

```
set (allVolumes | volume ["volumeName"] |
volumes ["volumeName1" ... "volumeNameN"] | volume <wwID>)
cacheFlushModifier=cacheFlushModifierValue
cacheWithoutBatteryEnabled=(TRUE | FALSE)
mediaScanEnabled=(TRUE | FALSE)
mirrorCacheEnabled=(TRUE | FALSE)
modificationPriority=(highest | high | medium | low | lowest)
owner=(a | b)
readCacheEnabled=(TRUE | FALSE)
writeCacheEnabled=(TRUE | FALSE)
cacheReadPrefetch=(TRUE | FALSE)
```

Syntax Applicable to Only One Volume

```
set (volume ["volumeName"] | volume <wwID>)
addCapacity=volumeCapacity
[addDrives=(trayID1,slotID1 ... trayIDn,slotIDn)]
redundancyCheckEnabled=(TRUE | FALSE)
segmentSize=segmentSizeValue
userLabel=volumeName
preReadRedundancyCheck=(TRUE | FALSE)
```

Syntax Applicable to Volume Mapping

```
set (volume=["volumeName"] | volume <wwID> | accessVolume)
logicalUnitNumber=LUN
(host="hostName" |
hostGroup=("hostGroupName" | defaultGroup)
```

Parameters

Parameter	Description
allVolumes	The properties for all volumes in the storage array.
volume or volumes	The name of the specific volume for which you want to define properties. (You can enter more than one volume name if you use the <code>volumes</code> parameter). Enclose the volume name in double quotation marks (" ") inside of square brackets ([]).
volume	The World Wide Identifier (WWID) of the volume for which you are setting properties. You can use the WWID instead of the volume name to identify the volume. Enclose the WWID in angle brackets (< >).
cacheFlushModifier	The maximum amount of time that data for the volume stays in cache before the data is flushed to physical storage. Valid values are listed in the Notes section.
cacheWithoutBatteryEnabled	The setting to turn on or turn off caching without batteries. To turn on caching without batteries, set this parameter to TRUE . To turn off caching without batteries, set this parameter to FALSE .
mediaScanEnabled	The setting to turn on or turn off media scan for the volume. To turn on media scan, set this parameter to TRUE . To turn off media scan, set this parameter to FALSE . (If media scan is disabled at the storage array level, this parameter has no effect.)
mirrorCacheEnabled	The setting to turn on or turn off the mirror cache. To turn on the mirror cache, set this parameter to TRUE . To turn off the mirror cache, set this parameter to FALSE .

Parameter	Description
modificationPriority	The priority for volume modifications while the storage array is operational. Valid values are highest , high , medium , low , or lowest .
owner	The controller that owns the volume. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Use this parameter only if you want to change the volume owner.
readCacheEnabled	The setting to turn on or turn off the read cache. To turn on the read cache, set this parameter to TRUE . To turn off the read cache, set this parameter to FALSE .
writeCacheEnabled	The setting to turn on or turn off the write cache. To turn on the write cache, set this parameter to TRUE . To turn off the write cache, set this parameter to FALSE .
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .
addCapacity	The setting to increase the storage size (capacity) of the volume for which you are defining properties. Size is defined in units of bytes , KB , MB , GB , or TB . The default value is bytes .
addDrives	The setting to add new drives to the volume. Specify the tray ID value and the slot ID value for each drive that you assign to the volume. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in square brackets ([]). Use this parameter with the <code>addCapacity</code> parameter if you need to specify additional drives to accommodate the new size.

Parameter	Description
redundancyCheckEnabled	The setting to turn on or turn off redundancy checking during a media scan. To turn on redundancy checking, set this parameter to TRUE . To turn off redundancy checking, set this parameter to FALSE .
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
userLabel	The new name that you want to give an existing volume. Enclose the new volume name in double quotation marks (" ").
preReadRedundancyCheck	The setting to check the consistency of RAID redundancy data on the stripes during read operations. Do not use this operation for non-redundant volumes, for example RAID 0. To check redundancy consistency, set this parameter to TRUE . For no stripe checking, set this parameter to FALSE .
accessVolume	The logical unit number for the access volume. The logical unit number is the only property that you can set for the access volume.
logicalUnitNumber	Specific volume-to-LUN mappings to an individual host. This parameter also assigns the host to a host group.
host	The name of the host to which the volume is mapped. Enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group to which the volume is mapped. Enclose the host group name in double quotation marks (" "). defaultGroup is the host group that contains the host to which the volume is mapped.

Notes

Host I/O errors might result in volume groups with more than 32 volumes. This operation might also result in internal controller reboots due to the expiration of the timeout period before the operation completes. If you experience this issue, quiesce host I/O and try the operation again.

When you use this command, you can specify one or more of the optional parameters.

You can apply these parameters to only one volume at a time:

- `addCapacity`
- `segmentSize`
- `userLabel`
- `logicalUnitNumber`

Add Capacity, Add Drives, and Segment Size

Setting the `addCapacity` parameter, the `addDrives` parameter, or the `segmentSize` parameter starts a long-running operation that you cannot stop. These long-running operations are performed in the background and do not prevent you from running other commands. To show the progress of long-running operations, use the `show volume actionProgress` command.

Access Volume

The access volume is the volume in a SAN environment that is used for in-band communication between the storage management software and the storage array controller. This volume uses a LUN address and consumes 20 MB of storage space that is not available for application data storage. An access volume is required only for in-band managed storage arrays. If you specify the `accessVolume` parameter, the only property you can set is the `logicalUnitNumber` parameter.

Cache Flush Modifier

Valid values for the cache flush modifier are listed in this table.

Value	Description
Immediate	Data is flushed as soon as it is placed into the cache.
250	Data is flushed after 250 ms.
500	Data is flushed after 500 ms.
750	Data is flushed after 750 ms.
1	Data is flushed after 1 s.
1500	Data is flushed after 1500 ms.
2	Data is flushed after 2 s.
5	Data is flushed after 5 s.
10	Data is flushed after 10 s.
20	Data is flushed after 20 s.
60	Data is flushed after 60 s (1 min.).
120	Data is flushed after 120 s (2 min.).
300	Data is flushed after 300 s (5 min.).
1200	Data is flushed after 1200 s (20 min.).
3600	Data is flushed after 3600 s (1 hr).
Infinite	Data in cache is not subject to any age or time constraints. The data is flushed based on other criteria that are managed by the controller.

Cache Without Battery Enabled

Write caching without batteries enables write caching to continue if the controller batteries are completely discharged, not fully charged, or not present. If you set this parameter to **TRUE** without an uninterruptible power supply (UPS) or other backup power source, you can lose data if the power to the storage array fails. This parameter has no effect if write caching is disabled.

Modification Priority

Modification priority defines the amount of system resources that are used when modifying volume properties. If you select the highest priority level, the volume modification uses the most system resources, which decreases the performance for host data transfers.

Cache Read Prefetch

The `cacheReadPrefetch` parameter enables the controller to copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drive into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests.

If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Minimum Firmware Level

5.00 adds the `addCapacity` parameter.

7.10 adds the `preReadRedundancyCheck` parameter.

Set Volume Copy

This command defines the properties for a volume copy pair.

Syntax

```
set volumeCopy target [targetName]  
[source [sourceName]]  
copyPriority=(highest | high | medium | low | lowest)  
targetReadOnlyEnabled=(TRUE | FALSE)
```

Parameters

Parameter	Description
target	The name of the target volume for which you want to define properties. Enclose the target volume name in square brackets ([]). If the target volume name has special characters, you also must enclose the target volume name in double quotation marks (" ").
source	The name of the source volume for which you want to define properties. Enclose the source volume name in square brackets ([]). If the source volume name has special characters, you also must enclose the source volume name in double quotation marks (" ").
copyPriority	The priority that the volume copy has relative to host I/O activity. Valid values are highest , high , medium , low , or lowest .
targetReadOnlyEnabled	The setting so that you can write to the target volume or only read from the target volume. To write to the target volume, set this parameter to FALSE . To prevent writing to the target volume, set this parameter to TRUE .

Notes

When you use this command, you can specify one or more of the optional parameters.

Minimum Firmware Level

5.40

Set Volume Group

This command defines the properties for a volume group.

Syntax

```
set volumeGroup [volumeGroupName]
addDrives=(trayID1,slotID1 ... trayIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6)
owner=(a | b)
availability=(online | offline)
```

Parameters

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) of the volume group for which you want to set properties. Enclose the volume group identifier in square brackets ([]).
addDrives	The drive, by tray and slot location, that you want to include in the volume group. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID values and the slot ID values in parentheses.
raidLevel	The RAID level for the volume group. Valid values are 0 , 1 , 3 , 5 , or 6 .
availability	The setting for the volume group, which is either online or offline .
owner	The controller that owns the volume group. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Use this parameter only if you want to change the volume group owner.

Notes

Host I/O errors might result in volume groups with more than 32 volumes. This operation also might result in internal controller reboots because the timeout period ends before the volume group definition is set. If you experience this issue, quiesce the host I/O operations, and try the command again. \ When you use this command, you can specify one or more of the parameters.

IMPORTANT Specifying the `addDrives` parameter or the `raidLevel` parameter starts a long-running operation that you cannot stop.

Minimum Firmware Level

6.10

7.10 adds RAID 6 capability.

Set Volume Group Forced State

This command moves a volume group into a Forced state. Use this command if the `start volumeGroup import` command does not move the volume group to an Imported state or if the import operation does not work because of hardware errors. In a Forced state, the volume group can be imported, and you can then identify the hardware errors.

Syntax

```
set volumeGroup [volumeGroupName] forcedState
```

Parameter

Parameter	Description
<code>volumeGroup</code>	The alphanumeric identifier (including - and _) of the volume group that you want to place in a Forced state. Enclose the volume group identifier in square brackets ([]).

Notes

You can move the drives that comprise a volume group from one storage array to another storage array. The CLI provides three commands that let you move the drives. The commands are `start volumeGroup export`, `start volumeGroup import`, and `set volumeGroup forcedState`.

In the Forced state, you can perform an import operation on the volume group.

Minimum Firmware Level

7.10

Show Controller

For each controller in a storage array, this command returns the following information:

- The status (Online or Offline)
- The current firmware and NVSRAM configuration
- The pending firmware configuration and NVSRAM configuration (if any)
- The board ID
- The product ID
- The product revision
- The serial number
- The date of manufacture
- The cache size or the processor size
- The date and the time to which the controller is set
- The associated volumes (including the preferred owner)
- The Ethernet port
- The physical disk interface
- The host interface, which applies only to Fibre Channel host interfaces

Syntax

```
show (allControllers | controller [(a | b)]) [summary]
```

Parameters

Parameter	Description
allControllers	The setting to return information about both controllers in the storage array.
controller	The setting to return information about a specific controller in the storage array. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]).
summary	The setting to return a concise list of information about both controllers in the storage array.

Notes

The following list is an example of the information that is returned by the `show controller` command. This example only shows how the information is presented and should not be considered to represent best practice for a storage array configuration.

```
Controller in Slot A
Status: Online
Current configuration
  Firmware version: 96.10.21.00
  Appware version: 96.10.21.00
  Bootware version: 96.10.21.00
  NVSRAM version: N4884-610800-001
Pending configuration
  Firmware version: Not applicable
  Appware version: Not applicable
  Bootware version: Not applicable
  NVSRAM version: Not applicable
  Transferred on: Not applicable
Board ID: 4884
Product ID: INF-01-00
Product revision: 9610
Serial number: 1T14148766
Date of manufacture: October 14, 2001
Cache/processor size (MB): 1024/128
Date/Time: Wed Feb 18 13:55:53 MST 2004
Associated Volumes (* = Preferred Owner):
  1*, 2*, CTL 0 Mirror Repository*, Mirror Repository 1*,
  JCG_Remote_MirrorMenuTests*
Ethernet port: 1
  MAC address: 00:a0:b8:0c:c3:f5
  Host name: ausctlr9
  Network configuration: Static
  IP address: 172.22.4.249
  Subnet mask: 255.255.255.0
  Gateway: 172.22.4.1
  Remote login: Enabled
Drive interface: Fibre
  Channel: 1
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
Drive interface: Fibre
  Channel: 2
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
```

```
Drive interface: Fibre
Channel: 3
Current ID: 125/0x1
Maximum data rate: 2 Gbps
Current data rate: 1 Gbps
Data rate control: Switch
Link status: Up
Drive interface: Fibre
Channel: 4
Current ID: 125/0x1
Maximum data rate: 2 Gbps
Current data rate: 1 Gbps
Data rate control: Switch
Link status: Up
Host interface: Fibre
Port: 1
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 126/0x0
NL-Port ID: 0x011100
Maximum data rate: 2 Gbps
Current data rate: 1 Gbps
Data rate control: Switch
Link status: Up
Topology: Fabric Attach
World-wide port name: 20:2c:00:a0:b8:0c:c3:f6
World-wide node name: 20:2c:00:a0:b8:0c:c3:f5
Part type: HPFC-5200    revision 10
Host interface: Fibre
Port: 2
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 126/0x0
NL-Port ID: 0x011000
Maximum data rate: 2 Gbps
Current data rate: 1 Gbps
Data rate control: Switch
Link status: Up
Topology: Fabric Attach
World-wide port name: 20:2c:00:a0:b8:0c:c3:f7
World-wide node name: 20:2c:00:a0:b8:0c:c3:f5
Part type: HPFC-5200    revision 10
```

When you use the `summary` parameter, the command returns the list of information without the drive channel information and the host channel information.

The `show storageArray` command also returns detailed information about the controller.

Minimum Firmware Level

5.43 adds the `summary` parameter.

Show Controller NVSRAM

This command returns a list of the NVSRAM byte values for the specified host type. If you do not enter the optional parameters, this command returns a list of all of the NVSRAM byte values. To view an example of a table of NVSRAM values that are returned by this command.

Syntax

```
show (allControllers | controller [(a | b)]) NVSRAM
[hostType=hostTypeIndexLabel | host="hostName"]
```

Parameters

Parameter	Description
allControllers	The setting to return information about both controllers in the storage array.
controller	The setting to return information about a specific controller in the storage array. Valid controller identifiers are a or b , where a is the controller in slot A, and b is the controller in slot B. Enclose the controller identifier in square brackets ([]).
hostType	The index label or number of the host type. Use the <code>show storageArray hostTypeTable</code> command to generate a list of available host type identifiers.
host	The name of the host that is connected to the controllers. Enclose the host name in double quotation marks (" ").

Notes

Use the `show controller NVSRAM` command to show parts of or all of the NVSRAM before using the `set controller` command to change NVSRAM values.

Minimum Firmware Level

6.10

Show Drive Channel Stats

This command shows the cumulative data transfer for the drive channel and error information. If the controller has automatically degraded a drive channel, this command also shows interval statistics. When you use this command, you can show information about one specific drive channel, several drive channels, or all drive channels. To view an example that shows a table that is returned by this command.

Syntax

```
show (driveChannel [(1 | 2 | 3 | 4 | 5 | 6 | 7 | 8)] |  
driveChannels [1 2 3 4 5 6 7 8] | allDriveChannels) stats
```

Parameters

Parameter	Description
driveChannel	The identifier number of the drive channel for which you want to show information. Valid drive channel values are 1, 2, 3, 4, 5, 6, 7, or 8 . Enclose the drive channel in square brackets ([]).
driveChannels	The identifier number of the drive channels for which you want to show information. Valid drive channel values are 1, 2, 3, 4, 5, 6, 7, or 8 . Enclose the drive channels in square brackets ([]) with the drive channel value separated by a space.
allDriveChannels	The identifier that selects all of the drive channels.

Minimum Firmware Level

6.10

7.30 adds up to eight drive channels.

Show Drive Download Progress

This command returns the status of firmware downloads for the drives that are targeted by the `download drive firmware` command or the `download storageArray driveFirmware` command.

Syntax

```
show allDrives downloadProgress
```

Parameters

None.

Notes

When all of the firmware downloads have successfully completed, this command returns good status. If any firmware downloads fail, this command shows the firmware download status of each drive that was targeted. This command returns the statuses shown in this table.

Status	Definition
Successful	The downloads completed without errors.
Not Attempted	The downloads did not start.
Partial Download	The download is in progress.
Failed	The downloads completed with errors.

Minimum Firmware Level

6.10

Show Host Ports

For all of the HBA host ports that are connected to a storage array, this command returns this information:

- The HBA host port identifier
- The HBA host port name
- The HBA host type

Syntax

```
show allHostPorts
```

Parameters

None.

Notes

This command returns HBA host port information similar to this example:

HOST PORT IDENTIFIER	HOST PORT NAME	HOST TYPE
12:34:56:54:33:22:22:22	Jupiter1	Solaris
12:34:56:78:98:98:88:88	Pluto1	Windows 2000/Server 2003 Clustered
54:32:12:34:34:55:65:66	Undefined	Undefined

Minimum Firmware Level

5.40

Show Remote Volume Mirroring Volume Candidates

This command returns information about the candidate volumes on a remote storage array that you can use as secondary volumes in a Remote Volume Mirroring configuration.

Syntax

```
show remoteMirror candidates primary="volumeName"  
remoteStorageArrayName="storageArrayName"
```

Parameters

Parameter	Description
primary	The name of the local volume that you want for the primary volume in the remote-mirror pair. Enclose the primary volume name in double quotation marks (" ").
remoteStorageArrayName	The remote storage array that contains possible volumes for a secondary volume. If the remote storage array name has special characters, you must also enclose the remote storage array name in double quotation marks (" ").

Minimum Firmware Level

5.40

Show Remote Volume Mirroring Volume Synchronization Progress

This command returns the progress of data synchronization between the primary volume and the secondary volume in a Remote Volume Mirroring configuration. This command shows the progress as a percentage of data synchronization that has been completed.

Syntax

```
show remoteMirror (localVolume ["volumeName"] |  
localVolumes ["volumeName1" ... "volumeNameN"])  
synchronizationProgress
```

Parameter

Parameter	Description
localVolume or localVolumes	The name of the primary volume of the remote mirror pair for which you want to check synchronization progress. Enclose the primary volume name in double quotation marks (" ") inside of square brackets ([]).

Minimum Firmware Level

5.40

Show Storage Array

This command returns configuration information about the storage array. The parameters return lists of values for the components and features in the storage array. You can enter the command with a single parameter or more than one parameter. If you enter the command without any parameters, the entire storage array profile is shown (which is the same information as if you entered the `profile` parameter).

Syntax

```
show storageArray profile batteryAge connections defaultHostType
healthStatus hostTypeTable hotSpareCoverage features time volumeDistribution
summary
```

Parameters

Parameter	Description
profile	The parameter to show all of the properties of the logical components and the physical components that comprise the storage array. The information appears in several screens.
batteryAge	The parameter to show the status, the age of the battery in days, and the number of days until the battery needs to be replaced.
connections	The parameter to show a list of where the drive channel ports are located and where the drive channels are connected.
defaultHostType	The parameter to show the default host type and the host type index.
healthStatus	The parameter to show the health, logical properties, and physical component properties of the storage array.
hostTypeTable	The parameter to show a table of all of the host types that are known to the controller. Each row in the table shows a host type index and the platform that the host type index represents.

Parameter	Description
hotSpareCoverage	The parameter to show information about which volumes of the storage array have hot spare coverage and which volumes do not.
features	The parameter to show a list of the feature identifiers for all enabled features in the storage array.
time	The parameter to show the current time to which both controllers in the storage array are set.
volumeDistribution	The parameter to show the current controller owner for each volume in the storage array.
summary	The parameter to show a concise list of information about the storage array configuration.

Notes

The `profile` parameter shows detailed information about the storage array. The information appears on several screens on a display monitor. You might need to increase the size of your display buffer to see all of the information. Because this information is so detailed, you might want to save the output of this parameter to a file. To save the output to a file, run the `show storageArray` command that looks like this example:

```
c:\...\smee\client>smcli 123.45.67.89 -c "show storagearray profile;" -o
"c:\\folder\\storagearray profile.txt"
```

The previous command syntax is for a host that is running a Windows operating system. The actual syntax varies depending on your operating system.

When you save the information to a file, you can use the information as a record of your configuration and as an aid during recovery.

The `batteryAge` parameter returns information in the form:

```
Battery status: Optimal
Age: 1 day(s)
Days until replacement: 718 day(s)
```

The `defaultHostType` parameter returns information in the form:

Default host type: Linux (Host type index 6)

The healthStatus parameter returns information in the form:

```
Storage array health status = optimal.
```

The hostTypeTable parameter returns information in the form:

```
NVSRAM Host Type Internal Definitions
Index 0
  Name: Windows NT Non-Clustered (SP5 or higher)
  AVT status: Disabled
Index 1
  Name: Windows 2000/Server 2003 Non-Clustered
  AVT status: Disabled
Index 2
  Name: Solaris
  AVT status: Disabled
Index 3
  Name: HP-UX
  AVT status: Enabled
Index 4
  Name: AIX
  AVT status: Disabled
Index 5
  Name: Irix
  AVT status: Disabled
Index 6 (DEFAULT)
  Name: Linux
  AVT status: Disabled
Index 7
  Name: Windows NT Clustered (SP5 or higher)
  AVT status: Disabled
Index 8
  Name: Windows 2000/Server 2003 Clustered
  AVT status: Disabled
Index 9
  Name: Netware Non-Failover
  AVT status: Enabled
Index 10
  Name: PTX
  AVT status: Enabled
Index 11
  Name: Netware Failover
  AVT status: Enabled
Index 12
  Name: Solaris (with Veritas DMP)
  AVT status: Enabled
```

The hotSpareCoverage parameter returns information in the form:

```
The following volume groups are not protected: 2, 1
Total hot spare drives: 0
  Standby: 0
  In use: 0
```

The `features` parameter returns information in the form:

```
storagePartitionMax
snapshot
remoteMirror
volumeCopy
```

The `time` parameter returns information in the form:

```
Controller in Slot A
Date/Time: Thu Jun 03 14:54:55 MDT 2004
Controller in Slot B
Date/Time: Thu Jun 03 14:54:59 MDT 2004
```

The `volumeDistribution` parameter returns information in the form:

```
Volume name: 10 Current owner is controller in slot: A
Volume name: CTL 0 Mirror Repository0 Current owner is controller in slot: A
Volume name: Mirror Repository 10 Current owner is controller in slot: A
Volume name: 20 Current owner is controller in slot: A
Volume name: JCG_Remote_MirrorMenuTests0 Current owner is controller in slot: A
```

Minimum Firmware Level

5.00 adds the `defaultHostType` parameter.

5.43 adds the `summary` parameter.

6.14 adds the `connections` parameter.

Show Storage Array Auto Configure

This command shows the default auto-configuration that the storage array creates if you run the `autoConfigure storageArray` command. If you want to determine whether the storage array can support specific properties, enter the parameter for the properties when you run this command. You do not need to enter any parameters for this command to return configuration information. If you do not specify any properties, this command returns the RAID 5 candidates for each drive type. If RAID 5 candidates are not available, this command returns candidates for RAID 3, RAID 1, or RAID 0. When you specify auto configuration properties, the controllers validate that the firmware can support the properties.

Syntax

```
show storageArray autoConfiguration
[driveType=(fibre | SATA | SAS | PATA)
raidLevel=(0 | 1 | 3 | 5 | 6)
volumeGroupWidth=numberOfDrives
volumeGroupCount=numberOfVolumeGroups
volumesPerGroupCount=numberOfVolumesPerGroup
hotSpareCount=numberOfHotspares
segmentSize=segmentSizeValue
cacheReadPrefetch=(TRUE | FALSE)]
```

Parameters

Parameter	Description
driveType	The type of drives that you want to use for the storage array. Valid drive types are fibre , SATA , SAS , or PATA . The <code>driveType</code> parameter is not required if only one type of drive is in the storage array.
raidLevel	The RAID level of the volume group that contains the drives in the storage array. Valid RAID levels are 0 , 1 , 3 , 5 , or 6 .
volumeGroupWidth	The number of drives in a volume group in the storage array, which depends on the capacity of the drives. Use integer values.
volumeGroupCount	The number of volume groups in the storage array. Use integer values.

Parameter	Description
volumesPerGroupCount	The number of equal-capacity volumes per volume group. Use integer values.
hotSpareCount	The number of hot spares that you want in the storage array. Use integer values.
segmentSize	The amount of data (in KB) that the controller writes on a single drive in a volume before writing data on the next drive. Valid values are 8, 16, 32, 64, 128, 256, or 512 .
cacheReadPrefetch	The setting to turn on or turn off cache read prefetch. To turn off cache read prefetch, set this parameter to FALSE . To turn on cache read prefetch, set this parameter to TRUE .

Notes

Drives

A volume group is a set of drives that are logically grouped together by the controllers in the storage array. The number of drives in a volume group is a limitation of the controller firmware. You should adhere to the 30-drive limit and to the 2-TB size maximum for the volume group, which some operating system require. The table relates drive capacity to the number of drives in a volume group.

Drive Size	Maximum Number of Drives in the Volume Group
18 GB	30 drives
36 GB	30 drives
73 GB	28 drives
180 GB	11 drives

A 73-GB drive or a 180-GB drive does not report exactly 73 GB or 180 GB as its capacity.

Hot Spares

Hot spare drives can replace any failed drive in the storage array. A hot spare drive must have capacity greater than or equal to any drive that can fail. If a hot spare drive is smaller than a failed drive, you cannot use the hot spare drive to rebuild the data from the failed drive. Hot spare drives are available only for RAID levels 1, 3, 5, or 6.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single drive in a volume before writing data on the next drive. Each data block stores 512 bytes of data. A data block is the smallest unit of storage. The size of a segment determines how many data blocks that it contains. For example, an 8-KB segment holds 16 data blocks. A 64-KB segment holds 128 data blocks.

When you enter a value for the segment size, the value is checked against the supported values that are provided by the controller at run time. If the value that you entered is not valid, the controller returns a list of valid values. Using a single drive for a single request leaves other drives available to simultaneously service other requests. If the volume is in an environment where a single user is transferring large units of data (such as multimedia), performance is maximized when a single data transfer request is serviced with a single data stripe. (A data stripe is the segment size that is multiplied by the number of drives in the volume group that are used for data transfers.) In this case, multiple drives are used for the same request, but each drive is accessed only once.

For optimal performance in a multiuser database or file system storage environment, set your segment size to minimize the number of drives that are required to satisfy a data transfer request.

Cache Read Prefetch

Cache read prefetch lets the controller copy additional data blocks into cache while the controller reads and copies data blocks that are requested by the host from the drive into cache. This action increases the chance that a future request for data can be fulfilled from cache. Cache read prefetch is important for multimedia applications that use sequential data transfers. The configuration settings for the storage array that you use determine the number of additional data blocks that the controller reads into cache. Valid values for the `cacheReadPrefetch` parameter are **TRUE** or **FALSE**.

Minimum Firmware Level

6.10

7.10 adds RAID 6 capability and removes hot spare limits.

Show Storage Array Host Topology

This command returns the storage partition topology, the host type labels, and the host type index for the host storage array.

Syntax

```
show storageArray hostTopology
```

Parameters

None.

Notes

This command returns the host topology information similar to this example:

TOPOLOGY DEFINITIONS

DEFAULT GROUP

Default type: Windows 2000/Server 2003 Non-Clustered

Host Group: scott

Host: scott1

Host Port: 28:37:48:55:55:55:55

Alias: scott11

Type: Windows 2000/Server 2003 Clustered

Host: scott2

Host Port: 98:77:66:55:44:33:21:23

Alias: scott21

Type: Windows 2000/Server 2003 Clustered

Host: Bill

Host Port: 12:34:55:67:89:88:88:88

Alias: Bill1

Type: Windows 2000/Server 2003 Non-Clustered

NVS RAM HOST TYPE INDEX DEFINITIONS

INDEX	AVT STATUS	TYPE
0	Disabled	Windows NT Non-Clustered (SP5 or higher)
1 (Default)	Disabled	Windows 2000/Server 2003 Non-Clustered
2	Disabled	Solaris
3	Enabled	HP-UX
4	Disabled	AIX
5	Disabled	Irix
6	Enabled	Linux
7	Disabled	Windows NT Clustered (SP5 or higher)
8	Disabled	Windows 2000/Server 2003 Clustered
9	Enabled	Netware Non-Failover
10	Enabled	PTX
11	Enabled	Netware Failover
12	Enabled	Solaris (with Veritas DMP)

Minimum Firmware Level

5.20

Show Storage Array LUN Mappings

This command returns information from the storage array profile about the LUN mappings in the storage array. Default group LUN mappings are always shown. If you run this command without any parameters, this command returns all of the LUN mappings.

Syntax

```
show storageArray lunMappings [host ["hostName"] |  
hostgroup ["hostGroupName"]]
```

Parameters

Parameter	Description
host	The name of a specific host for which you want to see the LUN mappings. Enclose the host name in double quotation marks (" ") inside of square brackets ([]).
hostGroup	The name of a specific host group for which you want to see the LUN mappings. Enclose the host group name in double quotation marks (" ") inside of square brackets ([]).

Notes

This command returns host topology information similar to this example:

MAPPINGS (Storage Partitioning - Enabled (0 of 16 used))-----

VOLUME NAME	LUN	CONTROLLER	ACCESSIBLE BY	VOLUME STATUS
Access Volume 7	7	A,B	Default Group	Optimal
21	21	B	Default Group	Optimal
22	22	B	Default Group	Optimal
23	23	B	Default Group	Optimal
24	24	B	Default Group	Optimal
25	25	B	Default Group	Optimal
26	26	B	Default Group	Optimal
27	27	B	Default Group	Optimal
28	28	B	Default Group	Optimal
29	29	B	Default Group	Optimal
30	30	B	Default Group	Optimal
31	31	B	Default Group	Optimal
32	32	B	Default Group	Optimal
33	33	B	Default Group	Optimal
34	34	B	Default Group	Optimal
35	35	B	Default Group	Optimal
36	36	B	Default Group	Optimal
37	37	B	Default Group	Optimal
38	38	B	Default Group	Optimal
39	39	B	Default Group	Optimal
40	40	B	Default Group	Optimal

Minimum Firmware Level

6.10

Show Storage Array Unreadable Sectors

This command returns a table of the addresses of all of the sectors in the storage array that cannot be read. The table is organized with column headings for the following information:

- 1 Volume user label
- 2 LUN
- 3 Accessible by (host or host group)
- 4 Date/time

- 5 Volume-relative logical block address (hexadecimal format – 0xxxxxxxxx)
- 6 Drive location (tray t, slot s)
- 7 Drive-relative logical block address (hexadecimal format – 0xxxxxxxxx)
- 8 Failure type

The data is sorted first by the volume user label and second by the logical block address (LBA). Each entry in the table corresponds to a single sector.

Syntax

```
show storageArray unreadableSectors
```

Parameters

None.

Minimum Firmware Level

6.10

Show String

This command shows a string of text from a script file. This command is similar to the `echo` command in MS-DOS and UNIX.

Syntax

```
show "string"
```

Parameters

None.

Notes

Enclose the string in double quotation marks (“ ”).

Minimum Firmware Level

6.10

Show Volume Action Progress

For a long-running operation that is currently running on a volume, this command returns information about the volume action and amount of the long-running operation that is completed. The amount of the long-running operation that is completed is shown as a percentage (for example, 25 means that 25 percent of the long-running operation is completed).

Syntax

```
show volume ["volumeName"] actionProgress
```

Parameters

Parameter	Description
volume	The name of the volume that is running the long-running operation. Enclose the volume name in double quotation marks (" ") inside of square brackets ([]).

Minimum Firmware Level

5.43

Show Volume Copy

This command returns this information about volume copy operations:

- The copy status
- The start time stamp
- The completion time stamp
- The copy priority
- The source volume World Wide Identifier (WWID) or the target volume WWID
- The target volume Read-Only attribute setting

You can retrieve information about a specific volume copy pair or all of the volume copy pairs in the storage array.

Syntax

```
show volumeCopy (allVolumes | source ["sourceName"] |  
target ["targetName"])
```

Parameters

Parameter	Description
allVolumes	The setting to return information about volume copy operations for all of the volume copy pairs.
source	The name of the source volume about which you want to retrieve information. Enclose the source volume name in double quotation marks (" ") inside of square brackets ([]).
target	The name of the target volume about which you want to retrieve information. Enclose the target volume name in double quotation marks (" ") inside of square brackets ([]).

Minimum Firmware Level

5.40

Show Volume Copy Source Candidates

This command returns information about the candidate volumes that you can use as the source for a volume copy operation.

Syntax

```
show volumeCopy sourceCandidates
```

Parameters

None.

Notes

This command returns volume copy source information as shown in this example:

```
Volume Name: 1
  Capacity(GB): 1.000
  Volume Group: 1
Volume Name: RAID1testusingGUI
  Capacity(GB): 1.000
  Volume Group: 2
Volume Name: Test
  Capacity(GB): 1.000
  Volume Group: 3
Volume Name: 2
  Capacity(GB): 1.000
  Volume Group: 3
Volume Name: 1S1
  Capacity(GB): 1.000
  Volume Group: none
```

Minimum Firmware Level

6.10

Show Volume Copy Target Candidates

This command returns information about the candidate volumes that you can use as the target for a volume copy operation.

Syntax

```
show volumeCopy source ["sourceName"] targetCandidates
```

Parameter

Parameter	Description
source	The name of the source volume for which you are trying to find a candidate target volume. Enclose the source volume name in double quotation marks (" ") inside of square brackets ([]).

Minimum Firmware Level

6.10

Show Volume Group

This command returns this information about a volume group:

- The status (online or offline)
- The drive type (Fibre Channel, SATA, SAS, or PATA)
- Tray loss protection (yes or no)
- The current owner (the controller in slot A or the controller in slot B)
- The associated volumes and free capacity
- The associated drives

Syntax

```
show volumeGroup [volumeGroupName]
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier of the volume group (including - and _) for which you want to show information. Enclose the volume group identifier in square brackets ([]).

Notes

This command returns volume group information as shown in this example:

```
Volume Group 1 (RAID 5)
  Status: Online
  Drive type: Fibre Channel
  Tray loss protection: No
  Current owner: Controller in slot A
  Associated volumes and free capacities:
    1 (1 GB), 1R1 (0.2 GB), Free Capacity (134.533 GB)
  Associated drives (in piece order):
    Drive at Tray 1, Slot 14
    Drive at Tray 1, Slot 13
    Drive at Tray 1, Slot 12
```

Minimum Firmware Level

6.10

Show Volume Group Export Dependencies

This command shows a list of dependencies for the drives in a volume group that you want to move from one storage array to a second storage array.

Syntax

```
show volumeGroup [volumeGroupName] exportDependencies
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) of the volume group for which you want to show export dependencies. Enclose the volume group identifier in square brackets ([]).

Notes

This command spins up the drives in a volume group, reads the DACstore, and shows a list of import dependencies for the volume group. The volume group must be in an Exported state or a Forced state.

Minimum Firmware Level

7.10

Show Volume Group Import Dependencies

This command shows a list of dependencies for the drives in a volume group that you want to move from one storage array to a second storage array.

Syntax

```
show volumeGroup [volumeGroupName] importDependencies [cancelImport=(TRUE | FALSE)]
```

Parameters

Parameter	Description
volumeGroup	The alphanumeric identifier (including - and _) of the volume group for which you want to show import dependencies. Enclose the volume group identifier in square brackets ([]).
cancelImport	The setting to spin the drives back down after the volume group dependencies have been read. To spin down the drives, set this parameter to TRUE . To let the drives stay spinning, set this parameter to FALSE .

Notes

This command returns the dependencies of a specific volume group, which must be in an Exported state or a Forced state. If a decision is made to retain the listed dependencies, then the `cancelImport` parameter can be enforced to spin the drives back down.

The `show volumeGroup importDependencies` command must be run before the `start volumeGroup import` command.

Minimum Firmware Level

7.10

Show Volume Performance Statistics

This command returns information about the performance of the volumes in a storage array.

Syntax

```
show (allVolumes | volume [volumeName]
volumes [volumeName1 ... volumeNameN]) performanceStats
```

Parameters

Parameter	Description
allVolumes	The setting to return performance statistics about all of the volumes in the storage array.
volume or volumes	The name of the specific volume for which you are retrieving performance statistics. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Notes

Before you run the `show volume performanceStat` command, run the `set session performanceMonitorInterval` command and the `set session performanceMonitorIterations` command to define how often you collect the statistics. The `show volume` command returns volume performance statistics as shown in this example:

```
Performance Monitor Statistics for Storage Array: ausctrl9
Date/Time: 2/19/04 4:03:09 PM
Polling interval in seconds: 10

Devices,Total,Read,Cache Hit,Current,Maximum,Current,Maximum
,IOs,Percentage,Percentage,KB/second,KB/second,IO/second,IO/second
Capture Iteration: 1
Date/Time: 2/19/04 4:03:09 PM
CONTROLLER IN SLOT A,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
Volume 1,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
Volume 2,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
Volume 3,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
Volume JCG_Remote_MirrorMenuTests,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
STORAGE ARRAY TOTALS,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
```

Minimum Firmware Level

6.10

Show Volume Reservations

This command returns information about the volumes that have persistent reservations.

Syntax

```
show (allVolumes | volume [volumeName] |
volumes [volumeName1 ... volumeNameN]) reservations
```

Parameters

Parameter	Description
allVolumes	The setting to return persistent reservation information about all of the volumes in the storage array.
volume or volumes	The name of the specific volume for which you are retrieving persistent reservation information. You can enter more than one volume name. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Minimum Firmware Level

5.40

Start Drive Channel Locate

This command identifies the drive trays that are connected to a specific drive channel by turning on the indicator lights for the drive tray that is connected to the drive channel. (Use the `stop driveChannel locate` command to turn off the indicator lights on the drive tray.)

Syntax

```
start driveChannel [(1 | 2 | 3 | 4 | 5 | 6 | 7 | 8)] locate
```

Parameter

Parameter	Description
driveChannel	The identifier number of the drive channel that you want to locate. Valid values for the identifier number for the drive channel are 1, 2, 3, 4, 5, 6, 7, or 8 . Enclose the drive channel identifier number in square brackets ([]).

Minimum Firmware Level

6.10

7.30 adds up to eight drive channels.

Start Drive Initialize

This command starts drive initialization.

ATTENTION **Possible damage to the storage array configuration** – As soon as you enter this command, all user data is destroyed.

Syntax

```
start drive [trayID,slotID] initialize
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID value and the slot ID value in square brackets ([]).

Minimum Firmware Level

6.10

Start Drive Locate

This command locates a drive by turning on an indicator light on the drive. (Run the `stop drive locate` command to turn off the indicator light on the drive.)

Syntax

```
start drive [trayID,slotID] locate
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID value and the slot ID value in square brackets ([]).

Minimum Firmware Level

6.10

Start Drive Reconstruction

This command starts reconstructing a drive.

Syntax

```
start drive [trayID,slotID] reconstruct
```

Parameter

Parameter	Description
drive	The tray and the slot where the drive resides. Tray ID values are 0 to 99 . Slot ID values are 1 to 32 . Enclose the tray ID value and the slot ID value in square brackets ([]).

Minimum Firmware Level

5.43

Start Remote Volume Mirroring Synchronization

This command starts Remote Volume Mirroring synchronization.

Syntax

```
start remoteMirror primary ["volumeName"] synchronize
```

Parameter

Parameter	Description
primary	The name of the primary volume for which you want to start synchronization. Enclose the primary volume name in double quotation marks (" ") inside of square brackets ([]).

Minimum Firmware Level

6.10

Start Storage Array Locate

This command locates a storage array by turning on the indicator lights for the storage array. (Use the `stop storageArray locate` command to turn off the indicator lights for the storage array.)

Syntax

```
start storageArray locate
```

Parameters

None.

Minimum Firmware Level

6.10

Start Tray Locate

This command locates a tray by turning on the indicator light. (Use the `stop tray locate` command to turn off the indicator light for the tray.)

Syntax

```
start tray [trayID] locate
```

Parameter

Parameter	Description
tray	The tray that you want to locate. Tray ID values are 0 to 99 . Enclose the tray ID value in square brackets ([]).

Minimum Firmware Level

6.10

Start Volume Group Defragment

This command starts a defragment operation on the specified volume group.

IMPORTANT Defragmenting a volume group starts a long-running operation that you cannot stop.

Syntax

```
start volumeGroup [volumeGroupName] defragment
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier of the volume group (including - and _) that you want to defragment. Enclose the volume group identifier in square brackets ([]).

Notes

Host I/O errors might result in the volume groups with more than 32 volumes. This operation also might result in internal controller reboots because the timeout period ends before the volume group definition is set. If you experience this issue, quiesce the host I/O operations, and try the command again.

Minimum Firmware Level

6.10

Start Volume Group Export

This command moves a volume group into an Exported state. Then you can remove the drives that comprise the volume group and reinstall the drives in a different storage array.

NOTE Within the volume group, you cannot move volumes that are associated with the premium features from one storage array to another storage array.

Syntax

```
start volumeGroup [volumeGroupName] export
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier of the volume group (including - and _) that you want to export. Enclose the volume group identifier in square brackets ([]).

Notes

When this command is successful, you can run the `start volumeGroup import` command to finish moving the volume group to a Complete state, which makes the volume group available to the new storage array. If this command is unsuccessful because hardware problems prevented the completion of the export, use the `set volumeGroup forceState` command. The `set volumeGroup forceState` command lets you use the `start volumeGroup import` command to import a volume group. After the volume group is in an Exported state or a Forced state, you can remove the drives that comprise the volume group from the storage array. You can reinstall the drives in a different storage array.

Minimum Firmware Level

7.10

Start Volume Group Import

This command moves a volume group into a Complete state to make a newly introduced volume group available to its new storage array. The volume group must be in an Exported state or a Forced state before you run this command. Upon successfully running the command, the volume group is operational.

NOTE Within the volume group, you cannot move volumes that are associated with the premium features from one storage array to another storage array.

Syntax

```
start volumeGroup [volumeGroupName] import
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier of the volume group (including - and _) that you want to import. Enclose the volume group identifier in square brackets ([]).

Notes

Higher-level volumes that are specifically related to premium features (Snapshot, Remote Volume Mirroring, Volume Copy, host-to-volume mapping, and persistent reservations) are removed as part of the import operation.

The `show volumeGroup importDependencies` command must be run before the `start volumeGroup import` command.

Minimum Firmware Level

7.10

Start Volume Group Locate

This command identifies the drives that are logically grouped together to form the specified volume group by blinking the indicator lights on the drives. (Use the `stop volumeGroup locate` command to turn off the indicator lights on the drives.)

Syntax

```
start volumeGroup [volumeGroupName] locate
```

Parameter

Parameter	Description
volumeGroup	The alphanumeric identifier of the volume group (including - and _) for which you want to locate the drives that belong to that volume group. Enclose the volume group identifier in square brackets ([]).

Minimum Firmware Level

6.16

Start Volume Initialization

This command starts the formatting of a volume in a storage array.

NOTE Formatting a volume starts a long-running operation that you cannot stop.

Syntax

```
start volume [volumeName] initialize
```

Parameter

Parameter	Description
volume	The name of the volume for which you are starting the formatting. Enclose the volume name in square brackets ([]). If the volume name has special characters, you also must enclose the volume name in double quotation marks (" ").

Minimum Firmware Level

6.10

Stop Drive Channel Locate

This command turns off the indicator lights on the drive trays that were turned on by the `start driveChannel locate` command.

Syntax

```
stop driveChannel locate
```

Parameters

None.

Minimum Firmware Level

6.10

Stop Drive Locate

This command turns off the indicator light on the drive that was turned on by the `start drive locate` command.

Syntax

```
stop drive locate
```

Parameters

None.

Minimum Firmware Level

Stop Snapshot

This command stops a copy-on-write operation.

Syntax

```
stop snapshot (volume [volumeName] |  
volumes [volumeName1 ... volumeNameN])
```

Parameters

Parameter	Description
volume or volumes	The name of the specific volume for which you want to stop a copy-on-write operation. You can enter more than one volume name. Enclose the volume names in square brackets ([]). If the volume names have special characters, you also must enclose the volume names in double quotation marks (" ").

Minimum Firmware Level

6.10

Stop Storage Array Drive Firmware Download

This command stops a firmware download to the drives in a storage array that was started with the `download storageArray driveFirmware` command. This command does not stop a firmware download that is already in progress to a drive. This command stops all firmware downloads to drives that are waiting for the download.

Syntax

```
stop storageArray driveFirmwareDownload
```

Parameters

None.

Minimum Firmware Level

6.10

Stop Storage Array Locate

This command turns off the indicator lights on the storage array that were turned on by the `start storageArray locate` command.

Syntax

```
stop storageArray locate
```

Parameters

None.

Minimum Firmware Level

6.10

Stop Tray Locate

This command turns off the indicator light on the tray that was turned on by the `start tray locate` command.

Syntax

```
stop tray locate
```

Parameters

None.

Minimum Firmware Level

6.10

Stop Volume Copy

This command stops a volume copy operation.

Syntax

```
stop volumeCopy target [targetName] source [sourceName]
```

Parameters

Parameter	Description
target	The name of the target volume for which you want to stop a volume copy operation. Enclose the target volume name in square brackets ([]). If the target volume name has special characters, you also must enclose the target volume name in double quotation marks (" ").
source	The name of the source volume for which you want to stop a volume copy operation. Enclose the source volume name in square brackets ([]). If the source volume name has special characters, you also must enclose the source volume name in double quotation marks (" ").

Minimum Firmware Level

5.40

Stop Volume Group Locate

This command turns off the indicator lights on the drives that were turned on by the `start volumeGroup locate` command.

Syntax

```
stop volumeGroup locate
```

Parameters

None.

Minimum Firmware Level

6.16

Suspend Remote Mirror

This command suspends a Remote Volume Mirroring operation.

Syntax

```
suspend remoteMirror (primary [primaryVolumeName]
primaries [primaryVolumeName1 ... primaryVolumeNameN])
writeConsistency=(TRUE | FALSE)
```

Parameters

Parameter	Description
primary or primaries	The name of the volume for which you want to suspend operation. Enclose the volume name in square brackets ([]). If the volume name has special characters, you must also enclose the volume name in double quotation marks (" ").
writeConsistency	This parameter defines whether the volumes identified in this command are in a write-consistency group or are separate. For the volumes in the same write-consistency group, set this parameter to TRUE . For the volumes that are separate, set this parameter to FALSE .

Notes

If you set the `writeConsistency` parameter to **TRUE**, the volumes must be in a write-consistency group (or groups). This command suspends all write-consistency groups that contain the volumes. For example, if volumes A, B, and C are in a write-consistency group and they have remote counterparts A', B', and C', the command:

```
suspend remoteMirror volume ["A"] writeConsistency=TRUE
```

suspends A-A', B-B', and C-C'. If you have a write-consistency group 1={A, B, C} and write-consistency group 2={D, E, F}, the command:

```
suspend remoteMirror volumes=["A", "D"] writeConsistency=TRUE
```

suspends both write-consistency groups.

Minimum Firmware Level

6.10

Examples of Information Returned by the Show Commands

This appendix provides examples of information that is returned by the `show` commands. These examples show the type of information and the information detail. This information is useful in determining the components, features, and identifiers that you might need when you configure or maintain a storage array.

Show Storage Array Command

The `show storageArray` command returns information about the components and the features in a storage array. If you run the command with the `profile` parameter, the command returns information in the form shown by this example. This information is the most detailed report that you can receive about the storage array. After you have configured a storage array, save the configuration description to a file as a reference.

```
Storage array profile
PROFILE FOR STORAGE ARRAY: example9 (x/xx/xx 1:06:05 PM)

SUMMARY-----
Number of controllers: 1
Number of volume groups: 1
Total number of volumes (includes an access volume): 5 of 2048 used
  Number of standard volumes: 2
  Number of access volumes: 1
  Number of mirror repositories: 2
  Number of snapshot repositories: 0
Number of snapshot volumes: 0
Number of mirrored pairs: 0 of 64 used
Number of copies: 0
Number of drives: 9
Supported drive types: Fibre (9)
Total hot spare drives: 0
  Standby: 0
  In use: 0
Access volume: LUN 7 (see Mappings section for details)
Default host type: Linux (Host type index 6)
Current configuration
  Firmware version: PkgInfo 96.10.21.00
  NVSRAM version: N4884-610800-001
Pending configuration
  Staged firmware download supported?: No
  Firmware version: Not applicable
  NVSRAM version: Not applicable
  Transferred on: Not applicable
NVSRAM configured for batteries?: Yes
Start cache flushing at (in percentage): 80
Stop cache flushing at (in percentage): 80
Cache block size (in KB): 4
Media scan frequency (in days): Disabled
Failover alert delay (in minutes): 5
Feature enable identifier: 1234567891011121314151617181ABCD
```

```
CONTROLLERS-----
Number of controllers: 1

Controller in Slot A
Status: Online
Current configuration
  Firmware version: 96.10.21.00
  Appware version: 96.10.21.00
  Bootware version: 96.10.21.00
  NVSRAM version: N4884-610800-001
Pending configuration
  Firmware version: Not applicable
  Appware version: Not applicable
  Bootware version: Not applicable
  NVSRAM version: Not applicable
  Transferred on: Not applicable
Board ID: 4884
Product ID: INF-01-00
Product revision: 9610
Serial number: 1T14148766
Date of manufacture: October 14, 2001
Cache/processor size (MB): 1024/128
Date/Time: Fri Feb 13 13:06:05 MST 2004
Associated Volumes (* = Preferred Owner):
  1*, CTL 0 Mirror Repository*, Mirror Repository 1*, Primary_volume*
Ethernet port: 1
  MAC address: 00:a0:b8:0c:c3:f5
  Host name: ausctlr9
  Network configuration: Static
  IP address: 172.22.4.249
  Subnet mask: 255.255.255.0
  Gateway: 172.22.4.1
  Remote login: Enabled

Drive interface: Fibre
  Channel: 1
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
Drive interface: Fibre
  Channel: 2
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
Drive interface: Fibre
  Channel: 3
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
Drive interface: Fibre
  Channel: 4
  Current ID: 125/0x1
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
```

```

Host interface: Fibre
  Port: 1
  Current ID: Not applicable/0xFFFFFFFF
  Preferred ID: 126/0x0
  NL-Port ID: 0x011100
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
  Topology: Fabric Attach
  World-wide port name: 20:2c:00:a0:b8:0c:c3:f6
  World-wide node name: 20:2c:00:a0:b8:0c:c3:f5
  Part type: HPFC-5200    revision 10
Host interface: Fibre
  Port: 2
  Current ID: Not applicable/0xFFFFFFFF
  Preferred ID: 126/0x0
  NL-Port ID: 0x011000
  Maximum data rate: 2 Gbps
  Current data rate: 1 Gbps
  Data rate control: Switch
  Link status: Up
  Topology: Fabric Attach
  World-wide port name: 20:2c:00:a0:b8:0c:c3:f7
  World-wide node name: 20:2c:00:a0:b8:0c:c3:f5
  Part type: HPFC-5200    revision 10

VOLUME GROUPS-----
Number of volume groups: 1

Volume Group 1 (RAID 5)
  Status: Online
  Drive type: Fibre Channel
  Tray loss protection: No
  Current owner: Controller in slot A
  Associated volumes and free capacities:
    CTL 0 Mirror Repository (0.125 GB), Mirror Repository 1 (0.125 GB),
    Primary_Volume (5 GB), 1 (1 GB), Free Capacity (129.113 GB)
  Associated drives (in piece order):
    Drive at Tray 0, Slot 1
    Drive at Tray 0, Slot 2
    Drive at Tray 0, Slot 3
    Drive at Tray 0, Slot 5
    Drive at Tray 0, Slot 6
    Drive at Tray 0, Slot 7
    Drive at Tray 0, Slot 10
    Drive at Tray 0, Slot 12
    Drive at Tray 0, Slot 14

```

STANDARD VOLUMES-----

SUMMARY

Number of standard volumes: 2
 See other Volumes sub-tabs for premium feature information.

NAME	STATUS	CAPACITY	RAID LEVEL	VOLUME GROUP
1	Optimal	1 GB	5	1
Primary_Volume	Optimal	5 GB	5	1

DETAILS

Volume name: 1
 Volume ID: 60:0a:0b:80:00:0c:c3:f5:00:00:00:23:40:22:86:ea
 Subsystem ID (SSID): 3
 Status: Optimal
 Drive type: Fibre Channel
 Tray loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 1 GB
 RAID level: 5
 Segment size: 64 KB
 Modification priority: High
 Associated volume group: 1
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Enabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 1
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Volume name: Primary_Volume
 Volume ID: 60:0a:0b:80:00:0c:c3:f5:00:00:00:1c:40:22:45:ca
 Subsystem ID (SSID): 2
 Status: Optimal
 Drive type: Fibre Channel
 Tray loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 5 GB
 RAID level: 5
 Segment size: 64 KB
 Modification priority: High
 Associated volume group: 1
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Enabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 1
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

MIRROR REPOSITORY VOLUMES-----

Number of mirror repositories: 0

SNAPSHOT VOLUMES-----

Number of snapshot volumes: 0

MIRRORED PAIRS-----

SUMMARY

Number of mirrored pairs: 0 of 64 used
See also Standard sub-tab for standard volume properties

COPIES-----

SUMMARY

Number of copies: 0
See also Standard sub-tab for standard volume properties

MISSING VOLUMES-----

Number of missing volumes: 0
See other Volumes sub-tabs for premium feature information

DRIVES-----

SUMMARY

Number of drives: 9
Supported drive types: Fibre (9)

BASIC:

TRAY,	SLOT	STATUS	CAPACITY	CURRENT DATA RATE	PRODUCT ID	FIRMWARE VERSION
0,	1	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	2	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	3	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	5	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	6	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	7	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	10	Optimal	16.959 GB	1 Gbps	ST318451FC	F28D
0,	12	Optimal	16.959 GB	1 Gbps	ST318451FC	F28B
0,	14	Optimal	16.959 GB	1 Gbps	ST318451FC	F28B

DRIVE CHANNELS:

TRAY,	SLOT	PREFERRED CHANNEL	REDUNDANT CHANNEL
0,	1	4	2
0,	2	2	4
0,	3	4	2
0,	5	4	2
0,	6	2	4
0,	7	4	2
0,	10	2	4
0,	12	2	4
0,	14	2	4

```
HOT SPARE COVERAGE:
  The following volume groups are not protected: 1

  Total hot spare drives: 0
    Standby: 0
    In use: 0

DETAILS
  Drive at Tray 0, Slot 1
    Drive port: 1, Channel: 4, ID: 0/0xEF
    Drive port: 2, Channel: 2, ID: 0/0xEF
    Drive path redundancy: OK
    Status: Optimal
    Raw capacity: 16.959 GB
    Usable capacity: 16.92 GB
    Current data rate: 1 Gbps
    Product ID: ST318451FC
    Firmware version: F28D
    Serial number: 3CC052RK00007116BMW7
    Vendor: SEAGATE
    Date of manufacture: July 19, 2001
    World-wide name: 20:00:00:20:37:f5:b3:b6
    Drive type: Fibre Channel
    Speed: 15015 RPM
    Mode: Assigned
    Associated volume group: Volume group 1

  Drive at Tray 0, Slot 2
    Drive port: 1, Channel: 2, ID: 1/0xE8
    Drive port: 2, Channel: 4, ID: 1/0xE8
    Drive path redundancy: OK
    Status: Optimal
    Raw capacity: 16.959 GB
    Usable capacity: 16.92 GB
    Current data rate: 1 Gbps
    Product ID: ST318451FC
    Firmware version: F28D
    Serial number: 3CC0Q06K0000720291J9
    Vendor: SEAGATE
    Date of manufacture: July 18, 2001
    World-wide name: 20:00:00:20:37:f5:ac:cd
    Drive type: Fibre Channel
    Speed: 15015 RPM
    Mode: Assigned
    Associated volume group: Volume group 1

  Drive at Tray 0, Slot 3
    Drive port: 1, Channel: 4, ID: 2/0xE4
    Drive port: 2, Channel: 2, ID: 2/0xE4
    Drive path redundancy: OK
    Status: Optimal
    Raw capacity: 16.959 GB
    Usable capacity: 16.92 GB
    Current data rate: 1 Gbps
    Product ID: ST318451FC
    Firmware version: F28D
    Serial number: 3CC052MA00007116DB89
    Vendor: SEAGATE
    Date of manufacture: July 19, 2001
    World-wide name: 20:00:00:20:37:f5:b3:fc
    Drive type: Fibre Channel
    Speed: 15015 RPM
    Mode: Assigned
    Associated volume group: Volume group 1
```

Drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28D
Serial number: 3CC0526N00007143A10K
Vendor: SEAGATE
Date of manufacture: July 19, 2001
World-wide name: 20:00:00:20:37:f5:b4:5b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 0, slot 6
Drive port: 1, Channel: 2, ID: 5/0xE0
Drive port: 2, Channel: 4, ID: 5/0xE0
Drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28D
Serial number: 3CC03R5800007112JU2D
Vendor: SEAGATE
Date of manufacture: July 19, 2001
World-wide name: 20:00:00:20:37:f5:b3:06
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 0, slot 7
Drive port: 1, Channel: 4, ID: 6/0xDC
Drive port: 2, Channel: 2, ID: 6/0xDC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28D
Serial number: 3CC052J400007116BSJV
Vendor: SEAGATE
Date of manufacture: July 19, 2001
World-wide name: 20:00:00:20:37:f5:b3:0b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

```
drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28D
Serial number: 3CC0P5Q80000711552CZ
Vendor: SEAGATE
Date of manufacture: June 30, 2001
World-wide name: 20:00:00:20:37:f5:69:50
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 0, Slot 12
Drive port: 1, Channel: 2, ID: 72/0x67
Drive port: 2, Channel: 4, ID: 72/0x67
Drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28B
Serial number: 3CC0P7TS00007116DL90
Vendor: LSILOGIC
Date of manufacture: Not available
World-wide name: 20:00:00:20:37:f5:68:76
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 0, Slot 14
Drive port: 1, Channel: 2, ID: 104/0x2E
Drive port: 2, Channel: 4, ID: 104/0x2E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 16.959 GB
Usable capacity: 16.92 GB
Current data rate: 1 Gbps
Product ID: ST318451FC
Firmware version: F28B
Serial number: 3CC03STH00007112JNM4
Vendor: LSILOGIC
Date of manufacture: Not available
World-wide name: 20:00:00:20:37:f5:b3:69
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1
```

```
DRIVE CHANNEL INFORMATION - Channel 1
SUMMARY-----
Channel 1 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:37 AM
Sample period (days, hh:mm:ss): 4 days, 02:06:52
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 3
Total I/O count: 11

DRIVE CHANNEL INFORMATION - Channel 2
SUMMARY-----
Channel 2 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 4 days, 02:06:47
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 239075

DRIVE CHANNEL INFORMATION - Channel 3
SUMMARY-----
Channel 3 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 4 days, 02:06:47
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 3
Total I/O count: 11

DRIVE CHANNEL INFORMATION - Channel 4
SUMMARY-----
Channel 4 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 4 days, 02:06:47
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 45807
```

```
TRAYS-----
Controller Tray Overall Component Information
Drive technology: Unknown
Minihub data rate mismatch: No
Fan canister: Optimal
Fan canister: Optimal
Battery status: Optimal
  Age: 0 day(s)
  Days until replacement: 719 day(s)
Power supply canister
  Status: Optimal
Power supply canister
  Status: Optimal
Temperature: Optimal
Host mini-hub canister
  Status: Optimal
  Location: Controller A - Port 1
  Serial number: SN 1T13742151
  Part number: PN 348-0043290
  Vendor: VN LSILOGIC
  Date of manufacture: Not available
SFP
  Status: Optimal
  Location: In connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o DFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M6R
  Vendor: IBM
  Date of manufacture: July 1, 2001
SFP
  Status: Optimal
  Location: Out connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o DFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M54
  Vendor: IBM
  Date of manufacture: July 1, 2001
```

```
Host mini-hub canister
  Status: Optimal
  Location: Controller A - Port 2
  Serial number: SN 1T13742219
  Part number: PN 348-0043290
  Vendor: VN LSILOGIC
  Date of manufacture: Not available
  SFP
    Status: Optimal
    Location: In connection
    Supported data rate(s): 1 Gbps, 2 Gbps
    Link length: Intermediate
    Connector: LC
    Transmitter type: Shortwave Laser w/o OFC
    Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
    IEEE company ID: 08 00 5a
    Revision: AA10
    Part number: IBM42P21SNY
    Serial number: 21P7053000M3Z
    Vendor: IBM
    Date of manufacture: July 1, 2001
  SFP
    Status: Optimal
    Location: Out connection
    Supported data rate(s): 1 Gbps, 2 Gbps
    Link length: Intermediate
    Connector: LC
    Transmitter type: Shortwave Laser w/o OFC
    Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
    IEEE company ID: 08 00 5a
    Revision: AA10
    Part number: IBM42P21SNY
    Serial number: 21P7053000M3J
    Vendor: IBM
    Date of manufacture: July 1, 2001
Drive mini-hub canister
  Status: Optimal
  Location: Channel 1
  Serial number: SN 1T13438343
  Part number: PN 348-0043290
  Vendor: VN LSILOGIC
  Date of manufacture: Not available
  SFP
    Status: Optimal
    Location: Out connection
    Supported data rate(s): 1 Gbps, 2 Gbps
    Link length: Intermediate
    Connector: LC
    Transmitter type: Shortwave Laser w/o OFC
    Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
    IEEE company ID: 08 00 5a
    Revision: AA10
    Part number: IBM42P21SNY
    Serial number: 21P7053000M6Z
    Vendor: IBM
    Date of manufacture: July 1, 2001
```

```
Drive mini-hub canister
Status: Optimal
Location: Channel 2
Serial number: SN 1T13844441
Part number: PN 19K1270
Vendor: VN IBM
Date of manufacture: Not available
SFP
  Status: Optimal
  Location: In connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M4E
  Vendor: IBM
  Date of manufacture: July 1, 2001
SFP
  Status: Optimal
  Location: Out connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M40
  Vendor: IBM
  Date of manufacture: July 1, 2001
Drive mini-hub canister
Status: Optimal
Location: Channel 3
Serial number: SN 1T13844373
Part number: PN 348-0043290
Vendor: VN LSILOGIC
Date of manufacture: Not available
SFP
  Status: Optimal
  Location: In connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M4R
  Vendor: IBM
  Date of manufacture: July 1, 2001
SFP
  Status: Optimal
  Location: Out connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000LXS
  Vendor: IBM
  Date of manufacture: July 1, 2001
```

```
Drive mini-hub canister
Status: Optimal
Location: Channel 4
Serial number: SN 1T13844386
Part number: PN 348-0043290
Vendor: VN LSILOGIC
Date of manufacture: Not available
SFP
  Status: Optimal
  Location: In connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M6E
  Vendor: IBM
  Date of manufacture: July 1, 2001
SFP
  Status: Optimal
  Location: Out connection
  Supported data rate(s): 1 Gbps, 2 Gbps
  Link length: Intermediate
  Connector: LC
  Transmitter type: Shortwave Laser w/o OFC
  Transmission media: TM Multi-mode 50m(M5) TM Multi-mode 62.5m(M6)
  IEEE company ID: 08 00 5a
  Revision: AA10
  Part number: IBM42P21SNY
  Serial number: 21P7053000M57
  Vendor: IBM
  Date of manufacture: July 1, 2001
```

Drive Tray 0 Overall Component Information

Drive technology: Fibre channel
Part number: PN 348-0042807
Serial number: SN 0146001584
Vendor: VN LSILOGIC
Date of manufacture: November 1, 2001
Tray path redundancy: OK
Fan canister: Optimal
Fan canister: Optimal
Power supply canister
Status: Optimal
Power supply canister
Status: Optimal
Temperature: Optimal
Temperature: Optimal
ESM card
Status: Optimal
Firmware version: 9163
Maximum data rate: 1 Gbps
Current data rate: 1 Gbps
Location: A (left canister)
Card communication: OK
Product ID: SYM2500-FC
Part number: PN 348-0043832
Serial number: SN 1T22941398
Vendor: LSILOGIC
Date of manufacture: July 1, 2002
GBIC
Status: Optimal
Location: In connection
Type: Optical Shortwave
ESM card
Status: Optimal
Firmware version: 9163
Maximum data rate: 1 Gbps
Current data rate: 1 Gbps
Location: B (right canister)
Card communication: OK
Product ID: SYM2500-FC
Part number: PN 348-0043832
Serial number: SN 1T13945863
Vendor: LSILOGIC
Date of manufacture: October 1, 2001
GBIC
Status: Optimal
Location: In connection
Type: Optical Shortwave

```
MAPPINGS (SANshare STORAGE PARTITIONING - ENABLED (0 of 64 used))-----
Volume-To-LUN Mappings
  Access Volume, LUN 7, Default Group

Topology
  [Undefined Host Ports]
    10:00:00:a0:b8:04:29:3f
    21:00:00:e0:8b:01:34:c1
    21:01:00:e0:8b:22:80:38

  [Default Group]

NVSRAM Host Type Internal Definitions
Index 0
  Name: Windows NT Non-Clustered (SP5 or higher)
  AVT status: Disabled
Index 1
  Name: Windows 2000/Server 2003 Non-Clustered
  AVT status: Disabled
Index 2
  Name: solaris
  AVT status: Disabled
Index 3
  Name: HP-UX
  AVT status: Enabled
Index 4
  Name: AIX
  AVT status: Disabled
Index 5
  Name: Irix
  AVT status: Disabled
Index 6 (DEFAULT)
  Name: Linux
  AVT status: Disabled
Index 7
  Name: Windows NT Clustered (SP5 or higher)
  AVT status: Disabled
Index 8
  Name: Windows 2000/Server 2003 Clustered
  AVT status: Disabled
Index 9
  Name: Netware Non-Failover
  AVT status: Enabled
Index 10
  Name: PTX
  AVT status: Enabled
Index 11
  Name: Netware Failover
  AVT status: Enabled
Index 12
  Name: solaris (with Veritas DMP)
  AVT status: Enabled
```

Show Controller NVSRAM

The `show controller NVSRAM` command returns a table of values in the controller NVSRAM that is similar to that shown in this example. With the information from the table, you can modify the contents of the NVSRAM by using the `set controller` command. This example shows information for a controller in slot A in a controller tray. You can produce a similar table for a controller in slot B, or you can produce a table for both controllers.

```

Controller "a":
Region 232:
  0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 233:
  0000: 0000 7f8e 0000 1384 ff01 0000 0000 0000 ...?...?.....
  0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0020: 0000 0308 8e08 8108 0000 0000 0000 0408 ...?.?.....
  0030: 0220 0220 8920 8820 8d00 0900 0308 0000 ...?.?.?.....
  0040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0050: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 234:
  0000: 1020 3040 5060 0000 0000 0000 0000 0000 ..0.P.....
  0010: 1121 3141 5161 0000 0000 0000 0000 0000 ..1AQa.....
  0020: 1222 3242 5262 0000 0000 0000 0000 0000 ..2BRb.....
  0030: 1323 3343 5363 0000 0000 0000 0000 0000 ..3CSc.....
  0040: 1424 3444 5464 0000 0000 0000 0000 0000 ..4DTd.....
  0050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0070: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 236:
  0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0040: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0070: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 237:
  0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
  0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 238:
  0000: 0000 ca32 0000 0000 0050 0600 0000 0000 ..?2.....P.....
  0010: 0000 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
  0020: 0000 0000 0000 0000 0000 0c04 008a 0000 0000 .....?.....
  0030: 0020 8741 1300 0000 0f00 1400 0000 0000 ..?A.....

```

```

0030: 732e 6b73 2e6c 7369 6c2e 636f 6da1 076a s.ks.lsil.com?.j
0040: 6761 7469 6361 a20b 6171 7561 3a6d 6172 gatica?.aqua.mar
0050: 696e 6511 012f b101 0ab0 0218 47ff 0000 ine...?..?.G...
0060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0090: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00a0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00b0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00c0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00d0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00e0: 0000 0000 0000 0000 0000 0000 0000 0000 .....
00f0: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 241:
0000: 0000 ca32 0000 0000 0050 0600 0000 0000 ..?2.....P.....
0010: 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
0020: 0000 0000 0000 0000 0000 0c04 008a 0000 0000 .....?....
0030: 0020 8741 1300 0000 0f00 1400 0000 0000 ..?A.....
0040: 4465 6661 756c 7400 0000 0000 0000 0000 Default.....
0050: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0060: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0070: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0080: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0090: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 574e 544e 434c 5350 3500 0000 0000 0000 WNTNCLSP5.....
0010: 0000 0000 0100 0000 0000 0000 0000 0000 .....
0020: 0100 0000 0000 0002 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 5732 4b4e 4554 4e43 4c00 0000 0000 0000 W2KNETNCL.....
0010: 0020 0000 0100 0000 0000 0000 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 534f 4c00 0000 0000 0000 0000 0000 0000 SOL.....
0010: 0120 0000 0100 0000 0001 0000 0000 0000 .....
0020: 0000 0000 0000 0002 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4850 5800 0000 0000 0000 0000 0000 0000 HPX.....
0010: 0120 0000 0100 0000 0101 0001 0101 0001 .....
0020: 0000 0000 0100 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4149 5800 0000 0000 0000 0000 0000 0000 AIX.....
0010: 0120 0000 0100 2000 0101 0000 0001 0000 .....
0020: 0000 0100 0000 0001 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4952 5800 0000 0000 0000 0000 0000 0000 IRX.....
0010: 007f 0000 0100 2000 0000 0000 0000 0000 .....
0020: 0000 0100 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

```

Region 242:
0000: 4c4e 5800 0000 0000 0000 0000 0000 0000 LNX.....
0010: 0120 0000 0100 0000 0001 0000 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 574e 5443 4c53 5035 0000 0000 0000 0000 WNTCLSP5.....
0010: 0000 0000 0100 0000 0001 0001 0000 0000 .....
0020: 0100 0000 0000 0002 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 5732 4b4e 4554 434c 0000 0000 0000 0000 W2KNETCL.....
0010: 0020 0000 0100 0000 0001 0001 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4e57 5200 0000 0000 0000 0000 0000 0000 NWR.....
0010: 007f 0000 0100 2000 0000 0000 0000 0000 .....
0020: 0100 0100 0100 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 5054 5800 0000 0000 0000 0000 0000 0000 PTX.....
0010: 0120 0000 0100 0000 0001 0000 0000 0000 .....
0020: 0100 0000 0100 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4e57 5246 4f00 0000 0000 0000 0000 0000 NWRFD.....
0010: 007f 0000 0100 0000 0000 0000 0000 0000 .....
0020: 0100 0100 0100 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 534f 4c41 5654 0000 0000 0000 0000 0000 SOLAVT.....
0010: 0120 0000 0100 0000 0001 0100 0000 0000 .....
0020: 0000 0001 0180 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

Show Drive

The show drive command returns information about the drives in a storage array.

```
DRIVES-----
SUMMARY
  Number of drives: 14
  Supported drive types: Fibre (14)

BASIC:
TRAY,  SLOT  STATUS  CAPACITY  CURRENT DATA RATE  PRODUCT ID  FIRMWARE VERSION
1, 1      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 2      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 3      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 4      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 5      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 6      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 7      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 8      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 9      Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 10     Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 11     Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 12     Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 13     Optimal  68.366 GB  2 Gbps      ST373453FC  MS03
1, 14     Optimal  68.366 GB  2 Gbps      ST373453FC  MS03

DRIVE CHANNELS:
TRAY,  SLOT  PREFERRED CHANNEL  REDUNDANT CHANNEL
1, 1      3  Drive path redundancy: Lost
1, 2      3  Drive path redundancy: Lost
1, 3      3  Drive path redundancy: Lost
1, 4      3  Drive path redundancy: Lost
1, 5      3  Drive path redundancy: Lost
1, 6      3  Drive path redundancy: Lost
1, 7      3  Drive path redundancy: Lost
1, 8      3  Drive path redundancy: Lost
1, 9      3  Drive path redundancy: Lost
1, 10     3  Drive path redundancy: Lost
1, 11     3  Drive path redundancy: Lost
1, 12     3  Drive path redundancy: Lost
1, 13     3  Drive path redundancy: Lost
1, 14     3  Drive path redundancy: Lost

HOT SPARE COVERAGE:
  The following volume groups are not protected: 3

  Total hot spare drives: 2
    Standby: 2
    In use: 0

  Standby drive at tray 1, slot 8 (Fibre, 68.366 GB)
    Protects the following volume groups: 2, 1
```

```
status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW0535L00007329LFD8
Vendor: SEAGATE
Date of manufacture: January 20, 2003
World-wide name: 20:00:00:04:cf:0b:c6:0d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

Drive at Tray 1, Slot 2
Drive port: 1, Channel: 3, ID: 9/0xD6
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW056D500007329VWAP
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:dc:29
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

Drive at Tray 1, Slot 3
Drive port: 1, Channel: 3, ID: 10/0xD5
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04LMK00007329DYZK
Vendor: SEAGATE
Date of manufacture: January 17, 2003
World-wide name: 20:00:00:04:cf:0b:b1:f6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Unassigned
Associated volume group: Volume group None
```

Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04N2T00007329DZGZ
Vendor: SEAGATE
Date of manufacture: January 17, 2003
World-wide name: 20:00:00:04:cf:0b:b1:81
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 3

Drive at Tray 1, Slot 5
Drive port: 1, Channel: 3, ID: 12/0x03
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04YN500007329JQYG
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:db:2c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Unassigned
Associated volume group: Volume group None

Drive at Tray 1, Slot 6
Drive port: 1, Channel: 3, ID: 13/0x02
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW056N300007328ZC57
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:db:66
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

```
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04XKQ00007329VUR8
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:db:a1
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

Drive at Tray 1, Slot 8
Drive port: 1, Channel: 3, ID: 15/0xCE
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04QL300007328H6BU
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:db:6f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Hot spare standby
Associated volume group: Volume group None

Drive at Tray 1, Slot 9
Drive port: 1, Channel: 3, ID: 81/0x54
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04W5E00007329EQX2
Vendor: SEAGATE
Date of manufacture: January 18, 2003
World-wide name: 20:00:00:04:cf:0b:b2:5b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Unassigned
Associated volume group: Volume group None
```

Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04V5P00007329DZPH
Vendor: SEAGATE
Date of manufacture: January 17, 2003
World-wide name: 20:00:00:04:cf:0b:b1:97
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

Drive at Tray 1, Slot 11
Drive port: 1, Channel: 3, ID: 65/0x71
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW056N700007329T7EG
Vendor: SEAGATE
Date of manufacture: January 23, 2003
World-wide name: 20:00:00:04:cf:0b:db:69
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 2

Drive at Tray 1, Slot 12
Drive port: 1, Channel: 3, ID: 73/0x66
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04DL400007329ERF1
Vendor: SEAGATE
Date of manufacture: January 17, 2003
World-wide name: 20:00:00:04:cf:0b:b1:84
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 1, Slot 13
Drive port: 1, Channel: 3, ID: 89/0x4A
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04DJT00007329DZJF
Vendor: SEAGATE
Date of manufacture: January 18, 2003
World-wide name: 20:00:00:04:cf:0b:b0:ff
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Drive at Tray 1, Slot 14
Drive port: 1, Channel: 3, ID: 105/0x2D
Drive path redundancy: Lost
Status: Optimal
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
Current data rate: 2 Gbps
Product ID: ST373453FC
Firmware version: MS03
Serial number: 3HW04VAK00007329ER3P
Vendor: SEAGATE
Date of manufacture: January 17, 2003
World-wide name: 20:00:00:04:cf:0b:b1:6e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated volume group: Volume group 1

Show Drive Channel Status

The `show drive channel stat` command returns information about the drive channels in a storage array. Use this information to determine how well the channels are running and errors that might be occurring on the channels.

```
Channel 1 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:37 AM
Sample period (days, hh:mm:ss): 8 days, 22:44:25
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 3
Total I/O count: 22

DRIVE CHANNEL INFORMATION - Channel 2

SUMMARY-----
Channel 2 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 8 days, 22:44:20
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 412340

DRIVE CHANNEL INFORMATION - Channel 3

SUMMARY-----
Channel 3 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 8 days, 22:44:20
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 3
Total I/O count: 22

DRIVE CHANNEL INFORMATION - Channel 4

SUMMARY-----
Channel 4 status: Optimal
Controller A link status: Up

CUMULATIVE ERROR COUNTS-----
Controller A Cumulative Error Counts
Baseline time set: 2/9/04 10:46:42 AM
Sample period (days, hh:mm:ss): 8 days, 22:44:20
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 162911
```

Show Volumes

The `show volume` command returns information about the volumes in a storage array.

```
STANDARD VOLUMES-----
SUMMARY
Number of standard volumes: 4
See other Volumes sub-tabs for premium feature information.

NAME                STATUS   CAPACITY  RAID LEVEL  VOLUME GROUP
1                   Optimal 1 GB      5           1
2                   Optimal 1 GB      0           3
RAID1testusingGUI  Optimal 1 GB      1           2
Test                Optimal 1 GB      0           3

DETAILS
Volume name: 1
Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:1f:40:04:0b:1a
Subsystem ID (SSID): 0
Status: Optimal
Drive type: Fibre Channel
Tray loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 1 GB
RAID level: 5
Segment size: 64 KB
Modification priority: High
Associated volume group: 1
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 1
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Volume name: 2
Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:36:40:2c:9b:70
Subsystem ID (SSID): 4
Status: Optimal
Drive type: Fibre Channel
Tray loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 1 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated volume group: 3
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 1
Enable background media scan: Disabled
Media scan with redundancy check: Disabled
```

Volume name: RAID1testusingGUI
Volume ID: 60:0a:0b:80:00:0f:21:8a:00:00:00:1b:40:12:8f:45
Subsystem ID (SSID): 1
Status: Optimal
Drive type: Fibre Channel
Tray loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 1 GB
RAID level: 1
Segment size: 64 KB
Modification priority: High
Associated volume group: 2
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 1
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Volume name: Test
Volume ID: 60:0a:0b:80:00:0f:21:8a:00:00:00:21:40:22:44:47
Subsystem ID (SSID): 3
Status: Optimal
Drive type: Fibre Channel
Tray loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 1 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated volume group: 3
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 1
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

```

SNAPSHOT REPOSITORY VOLUMES-----
SUMMARY
  Number of snapshot repositories: 1

  Number of snapshot repositories: 1

  NAME  CAPACITY USAGE(%) THRESHOLD WARNING  FULL POLICY
  1R1   0              50% full          Fail snapshot volume

DETAILS
  Snapshot repository name: 1R1
  Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:27:40:14:ed:4a
  Subsystem ID (SSID): 2
  Status: Optimal
  Drive type: Fibre Channel
  Tray loss protection: No
  Preferred owner: Controller in slot A
  Current owner: Controller in slot A
  Capacity: 0.2 GB
  RAID level: 5
  Segment size: 64 KB
  Modification priority: High
  Associated volume group: 1
  Read cache: Enabled
  Write cache: Enabled
  Write cache without batteries: Disabled
  Write cache with mirroring: Enabled
  Flush write cache after (in seconds): 10.00
  Cache read ahead multiplier: 0
  Enable background media scan: Disabled
  Media scan with redundancy check: Disabled
  Capacity usage (%): 0
  Notify when capacity reaches: 50% full
  Snapshot repository full policy: Fail snapshot volume
  Associated base volume (standard): 1
  Associated snapshot volume: 1S1

```

MIRROR REPOSITORY VOLUMES-----

Number of mirror repositories: 0

SNAPSHOT VOLUMES-----

SUMMARY

Number of snapshot volumes: 1

NAME	STATUS	CREATION	TIMESTAMP
1S1	Optimal	1/26/04	9:34 AM

DETAILS

Snapshot volume name: 1S1
 Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:2a:40:14:ed:4c
 Creation timestamp: 1/26/04 9:34 AM
 Capacity: 1 GB
 Status: Optimal
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Associated base volume (standard): 1
 Associated snapshot repository volume: 1R1

COPIES-----

SUMMARY

Number of copies: 2
 See also Standard sub-tab for standard volume properties
 S = Source volume
 T = Target volume

COPY PAIR	STATUS	COMPLETION	TIMESTAMP
1 (S), RAID1testusingGUI (T)	Completed	2/5/04	9:19:34 AM
Test (S), 2 (T)	Completed	2/13/04	8:46:57 AM

DETAILS

Copy pair: 1 and RAID1testusingGUI
 Copy status: Completed
 Start timestamp: 2/5/04 9:17:09 AM
 Completion timestamp: 2/5/04 9:19:34 AM
 Copy priority: Medium
 Source volume: 1
 Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:1f:40:04:0b:1a
 Target volume: RAID1testusingGUI
 Volume ID: 60:0a:0b:80:00:0f:21:8a:00:00:00:1b:40:12:8f:45
 Read-only: Enabled

Copy pair: Test and 2
 Copy status: Completed
 Start timestamp: 2/13/04 8:46:14 AM
 Completion timestamp: 2/13/04 8:46:57 AM
 Copy priority: Highest
 Source volume: Test
 Volume ID: 60:0a:0b:80:00:0f:21:8a:00:00:00:21:40:22:44:47
 Target volume: 2
 Volume ID: 60:0a:0b:80:00:0f:21:a8:00:00:00:36:40:2c:9b:70
 Read-only: Enabled

Example Script Files

This appendix provides example scripts for configuring a storage array. These examples show how the script commands appear in a complete script file. Also, you can copy these scripts and modify them to create a configuration unique to your storage array.

You can create a script file in two ways:

- Using the `save storageArray configuration` command
- Writing a script

By using the `save storageArray configuration` command, you can create a file that you can use to copy an existing configuration from one storage array to other storage arrays. You can also use this file to restore an existing configuration that has become corrupted. You also can copy an existing file to serve as a pattern from which you create a new script file by modifying portions of the original file. The default file extension is `.scr`.

You can create a new script file by using a text editor, such as Microsoft Notepad. The maximum line length is 256 characters. The command syntax must conform to the usage guidelines on page [28](#) and the command formatting rules on page [151](#). When you create a new script file, you can use any file name and extension that will run on the host operating system.

This example shows how to run a script file from the command line.

```
c:\...\smee\client>smcli 123.45.67.89 -f scriptfile.scr;
```

Configuration Script Example 1

This example creates a new volume by using the `create volume` command in the free space of a volume group.

```
Show "Create RAID 5 Volume 7 on existing Volume Group 1";

//Create volume on volume group created by the create
volume drives command

//Note: For volume groups that use all available
capacity, the last volume on the group is created using
all remaining capacity by omitting the capacity=volume
creation parameter

create volume volumeGroup=1 RAIDLevel=5 userLabel="7"
owner=A segmentSize=16 cacheReadPrefetch=TRUE capacity=2 GB;

show "Setting additional attributes for volume 7";
//Configuration settings that cannot be set during volume
creation
set volume ["7"] cacheFlushModifier=10;
set volume ["7"] cacheWithoutBatteryEnabled=false;
set volume ["7"] mirrorEnabled=true;
set volume ["7"] readCacheEnabled=true;
set volume ["7"] writeCacheEnabled=true;
set volume ["7"] mediaScanEnabled=false;
set volume ["7"] redundancyCheckEnabled=false;
set volume ["7"] modificationPriority=high;
```

This example shows blank lines between the lines beginning with `Show`, `Create`, `//Note`, and `create`. The blank lines are included in this example only for clarity. Each command is actually written on one line in the script file; however, the size of this page has caused the command text to wrap. You might want to include blank lines in your script files to separate blocks of commands or make a comment that stands out. To include a comment, enter two forward slashes (`//`), which causes the script engine to treat the line as a comment.

The first line of text is the `show string` command. This command shows text that is bounded by double quotation marks (" ") on a display monitor when the script file runs. In this example, the text `Create RAID 5 Volume 7 on existing Volume Group 1` serves as a title that describes the expected results of running this script file.

The line beginning with `//Create` is a comment that explains that the purpose of this script file is to create a new volume by using the `create volume` command on an existing volume group.

The line beginning `//Note:` is a comment in the script file that explains that the size of the last volume created that uses all of the available capacity because the `capacity` parameter is not used.

The command in this example creates a new volume in volume group 1. The volume has a RAID level of 5. The volume name (user label) is 7. (Note the double quotation marks around the 7. The double quotation marks define that the information in the double quotation marks is a label.) The new volume is assigned to the controller in slot A in the controller tray. The segment size is set to 16. The volume has a read ahead multiplier value of 256. The capacity of the volume is 2 GB.

The command takes this form:

```
create volume volumeGroup=volumeGroupName
userLabel=volumeName
[freeCapacityArea=freeCapacityIndexNumber]
[capacity=volumeCapacity | owner=(a | b) |
cacheReadPrefetch=(TRUE | FALSE) |
segmentSize=segmentSizeValue]
[trayLossProtect=(TRUE | FALSE)]
```

The general form of the command shows the optional parameters in a different sequence than the optional parameters in the example command. You can enter optional parameters in any sequence. You must enter the required parameters in the sequence shown in the command descriptions.

The line showing “Setting additional attributes for volume 7” is another example of using the `show “string”` command. The reason for placing this command here is to tell the user that the `create volume` command ran successfully and that properties that could not be set by the `create volume` command are now set.

The `set volume` parameters are shown on separate lines. You do not need to use separate lines for each parameter. You can enter more than one parameter with the `set volume` command by leaving a space between the parameters, as in this example:

```
set volume["7"] cacheFlushModifier=10
cacheWithoutBatteryEnabled=false
modificationPriority=high;
```

By using separate lines, you can see more clearly the parameters that you are setting and the values to which you are setting the parameters. Blocking the parameters in this manner makes it easier to either edit the file or copy specific parameter settings for use in another script file.

Configuration Script Example 2

This example creates a new volume by using the `create volume` command with user-defined drives in the storage array.

```
Show "Create RAID3 Volume 2 on existing Volume Group 2";

//This command creates the volume group and the initial volume on that group.
//Note: For volume groups that use all available capacity, the last volume
on the volume group is created using all remaining capacity by omitting the
capacity=volume creation parameter

create volume RAIDLevel=3 userLabel="2" drives=[0,1 0,6 1,7 1,3 2,3 2,6]
owner=B segmentSize=16 capacity=2 GB;

show "Setting additional attributes for volumem 7"
//Configuration settings that cannot be set during volume creation
set volume ["7"] cacheFlushModifier=10;
set volume ["7"] cacheWithoutBatteryEnabled=false;
set volume ["7"] mirrorEnabled=true;
set volume ["7"] readCacheEnabled=true;
set volume ["7"] writeCacheEnabled=true;
set volume ["7"] mediaScanEnabled=false;
set volume ["7"] redundantCheckEnabled=false;
set volume ["7"] modificationPriority=high;
```

The command in this example, like the `create volume` command in the previous example, creates a new volume. The significant difference between these two examples is that this example shows how you can define specific drives to include in the volume. Use the `show storageArray profile` command to find out what drives are available in a storage array.

The `create volume` command takes this form:

```
create volume raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=volumeName  
drives=(trayID1,slotID1...trayIDn,slotIDn)  
[capacity=volumeCapacity | owner=(a | b) |  
cacheReadPrefetch=(TRUE | FALSE) |  
segmentSize=segmentSizeValue]  
[trayLossProtect=(TRUE | FALSE)]
```


Deprecated Commands and Parameters

This appendix lists the commands, the command formats, and the parameters that are no longer supported by this level of software. The information is presented in two tables. “[Deprecated Commands](#)” lists commands that are no longer supported in this level of software and the new commands that replaced them. “[Deprecated Parameters](#)” on page 378 lists the parameters that are no longer supported in this level of software and the new parameters that replaced them.

Deprecated Commands

Table 26 Deprecated Commands

Deprecated Command	New Command
<code>accept storageArray pendingTopology (allHosts host [user-label] hosts [user-label])</code>	Removed.
<code>create hostPort</code>	The requirement to set the host type has been removed. The <code>hostType</code> parameter is used with the <code>create host</code> command.
<code>create mapping volume=userLabel logicalGroupNumber=logicalGroupNum ber [host hostGroup]=hostName hostGroupName</code>	Use the <code>set volume</code> command to define the volume-to-LUN mapping.
<code>create volume (drive drives) [trayID1,slotID1 ... trayIDn,slotIDn]</code>	<code>create volume drives=(trayID1,slotID1... trayIDn,slotIDn)</code> The new syntax for specifying drives requires an equal sign (=) after the <code>drives</code> parameter.

Deprecated Command	New Command
<pre>create volume driveCount [numberOfDrives]</pre>	<pre>create volume driveCount=<i>numberOfDrives</i></pre> <p>The new syntax for specifying the number of drives requires an equal sign (=) after the <code>driveCount</code> parameter.</p>
<pre>create volume volumeGroup [numberOfDrives]</pre>	<pre>create volume volumeGroup=<i>volumeGroupName</i></pre> <p>The new syntax for specifying the volume group name requires an equal sign (=) after the <code>volumeGroup</code> parameter.</p>
<pre>delete mapping volume=<i>userLabel</i> [host hostGroup]=<i>hostName</i> <i>hostGroupName</i></pre>	<p>Use the <code>remove volume LUNMapping</code> command to remove a volume-to-LUN mapping.</p>
<pre>disableSnapshot volume</pre>	<p>Use the <code>stop snapshot</code> command to stop a copy-on-write operation.</p>
<pre>download drive [trayID,slotID] file=<i>filename</i>content=(<i>firmware</i> modePage)</pre>	<p>Use the <code>download storageArray driveFirmware</code> command to download the firmware images to all of the drives in the storage array.</p>
<pre>download storageArray (<i>firmwareFile</i> <i>NVSRAMFile</i>)=<i>filename</i></pre>	<pre>download drive [trayID,slotID] <i>firmware</i> file="<i>filename</i>" download storageArray <i>firmware</i> [, <i>NVSRAM</i>] file="<i>filename</i>" [, "<i>NVSRAM-filename</i>"] [downgrade=(TRUE FALSE)] [activateNow=(TRUE FALSE)]</pre> <p>The new version of the storage management software provides unique commands to perform these functions.</p>
<pre>download storageArray file=<i>filename</i> content=<i>firmware</i> [downgrade=(TRUE FALSE)]</pre>	<p>Use the <code>download storageArray firmware</code> command to download the firmware.</p>
<pre>download storageArray file=<i>filename</i> content=<i>NVSRAM</i></pre>	<p>Use the <code>download storageArray NVSRAM</code> command to download the NVSRAM values. See</p>
<pre>download storageArray file=<i>filename</i> content=<i>featureKey</i></pre>	<p>Use the <code>enable storageArray feature</code> command to enable a premium feature.</p>

Deprecated Command	New Command
download (allTrays tray [trayID]) file=filename content=firmware	Use the download (environmental card) firmware command to download the tray firmware.
download tray [0]	download allTrays firmware file="filename" When you download ESM firmware to all of the drive trays, in the previous command "all trays" was defined by entering [0]. The new command uses the allTrays parameter.
recreate storageArray mirrorRepository	The functionality is no longer supported.
recreateSnapshot volume	Use the recreate snapshot command to start a fresh copy-on-write operation by using an existing snapshot volume.
remove copyEntry target [targetName] [source [sourceName]]	Use the remove volumeCopy command to remove volume copy entries.
remove volumeReservations (allVolumes volume [volumeName])	Use the clear volume command to clear persistent volume reservations.
set controller [(a b)] batteryInstallDate=(TRUE FALSE)	Use the reset storageArray batteryInstallDate command to reset the battery date.
set controller [(a b)] NVSRAMByte [nvsram-offset]=(nvsramByteSetting nvsramBitSetting)	set controller [(a b)] globalNVSRAMByte [nvsramOffset=(nvsramByteSetting nvsramBitSetting) This new command provides additional parameters for setting the NVSRAM values.
set controller [(a b)] serviceMode=(TRUE FALSE)	Use the set controller availability=serviceMode command to place the storage array in Service mode.
set drive [trayID,slotID] operationalState=(optimal failed)	Use the set drive operationalState=failed command to place a drive in the storage array in Failed mode. To return a drive to the Optimal state, use the revive drive command.

Deprecated Command	New Command
set hostPort	The requirement to set the host type has been removed. The hostType parameter is used with the Create Host statement.
set performanceMonitor interval= <i>intervalValue</i> iterations= <i>iterationValue</i>	Use the set sessions command to define values for the performance monitor interval and iterations.
set storageArray batteryInstallDate=(TRUE FALSE)	Use the reset storageArray batteryInstallDate command to reset the battery date.
set storageArray clearEventLog=(TRUE FALSE)	Use the clear storageArray eventLog command to clear the Event Log for the storage array.
set storageArray resetConfiguration=(TRUE FALSE)	Use the clear storageArray configuration command to clear the entire configuration from the controllers in a storage array.
set storageArray RLSBaseline=currentTime	Use the reset storageArray RLSBaseline command to reset the Read Link Status (RLS) baseline for all of the devices.
set storageArray dayOfTime=(TRUE FALSE)	Use the set storageArray time command to set the clocks on both of the controllers in a storage array to the clock of the host.
set volume [<i>volumeName</i>] mirrorEnabled=(TRUE FALSE)	Use the set volume command with mirror cache enabled.
set volumeCopy target [<i>targetName</i>] [source [<i>sourceName</i>]] priority=(lower low medium high highest)	Use the set volumeCopy command to define the volume copy pair.
set volumeLabel ID [<i>hexValue</i>] userLabel= <i>volumeName</i>	Use the set volume command to define a user name for a volume.
show hostTopology	Use the show storageArray hostTopology command to show all of the mappings, the storage partition topology, the host type labels, and the host type index for the host storage array.
show storageArray pendingTopology	Removed.

Deprecated Command	New Command
show storageArray preferredVolumeOwners	show storageArray profile This command, with the <code>profile</code> parameter, returns information about the preferred volume owner.
show volumes volume [<i>userLabel</i>]	show storageArray profile This command, with the <code>profile</code> parameter, returns information about the volume.
start increaseVolCapacity volume= <i>volumeName</i> incrementalCapacity= <i>capacityValue</i> drives=(<i>trayID1</i> , <i>slotID1</i> ... <i>trayIDn</i> , <i>slotIDn</i>)	Use the <code>set volume</code> command to define values for increasing the capacity of a volume.
start volumeCopy source=" <i>sourceName</i> " target=" <i>targetName</i> " copyPriority=(lowest low medium high highest)	Use the <code>create volumeCopy</code> command to create a volume copy pair and to start the volume copy.
upload storageArray file= <i>filename</i> content= <i>configuration</i>	Use the <code>save configuration</code> command to save a storage array configuration.
upload storageArray file= <i>filename</i> content=(allEvents criticalEvents)	Use the <code>save storageArray (allEvents criticalEvents)</code> command to save events to a file.
upload storageArray file= <i>filename</i> content= <i>performanceStats</i>	Use the <code>save storageArray performanceStats</code> command to save the performance statistics to a file.
upload storageArray file= <i>filename</i> content= <i>RLSCounts</i>	Use the <code>save storageArray RLSCounts</code> command to save the RLS counters to a file.
upload storageArray file= <i>filename</i> content= <i>stateDump</i>	Use the <code>save storageArray stateCapture</code> command to save state dumps to a file.

For information on how to handle errors and on how to define a password, use the `set session` command.

Deprecated Parameters

Table 27 Deprecated Parameters

Old Syntax	New Syntax
bootp	Removed
clearEventLog	clear storageArray eventLog
copyEntry	volumeCopy
database	Removed
disableSnapshot	stop snapshot
enforceSoftLimit	Removed
featureKey	feature
filesystem	Removed
gatewayIPAddress	IPv4GatewayIP
hostType	Removed from the create host port command and the set host port command.
id[]	volume<>
increaseVolCapacity	set volume addCapacity
incrementalCapacity	addCapacity
ipAddress	IPv4Address
mapping	lunMapping
modePage	Removed
multimedia	Removed
on error	set session errorAction
performanceMonitor interval	performanceMonitorInterval
performanceMonitor iterations	performanceMonitorIterations
priority	copyPriority
readAheadMultiplier	cacheReadPrefetch
recreateSnapshot	recreate snapshot

Old Syntax	New Syntax
resetConfiguration	reset storageArray configuration
stateDump	stateCapture
subnetMask	IPv4SubnetMask
timeOfDay	time
upload	save
use password	set session password
volumeLabel	Removed
volumeReservations	show volume reservations or reservations

Asynchronous Remote Volume Mirroring

This appendix describes the host utility to achieve periodic consistency with Asynchronous Remote Volume Mirroring configurations. This appendix also describes how to run the Asynchronous Remote Volume Mirroring utility.

Description of the Asynchronous Remote Volume Mirroring Utility

The Asynchronous Remote Volume Mirroring utility lets you periodically synchronize the Remote Volume Mirroring pairs in your storage array. When defining a Remote Volume Mirroring configuration, you have the option to set the write modes to either Synchronous or Asynchronous. Synchronous write mode provides the highest level security for full data recovery from the secondary storage array in the event of a disaster. Synchronous write mode does, however, reduce host I/O performance. Asynchronous write mode offers faster host I/O performance, but it does not guarantee that a copy operation has successfully completed before processing the next write request. With Asynchronous write mode, you cannot make sure that a volume, or collection of volumes, at a secondary site ever reach a consistent, recoverable state.

The Asynchronous Remote Volume Mirroring utility enables you to bring a collection of asynchronous remote volumes into a mutually consistent and recoverable state. You can choose to run the utility based on application demands, link state and speed, and other factors that are relevant to your environment.

The Asynchronous Remote Volume Mirroring utility has these characteristics:

- The utility is implemented as a command line-invoked Java-based application.
- The utility is bundled as part of the ISSM-EE installation package.
- The utility accepts a command line argument that lets you specify the name of a configuration file that contains a complete specification of the work to be carried out by the utility.
- More than one instance of the utility can run concurrently, as long as the utilities do not try to process any of the same volumes and mirrors.

NOTE The Asynchronous Remote Volume Mirroring utility does not check to make sure that concurrently running instances of the utility are not trying to process the same volumes and mirrors. If you choose to simultaneously run more than one instance of the Asynchronous Remote Volume Mirroring utility, you must make sure that the configuration files that you choose to run do not list the same volumes and mirrors.

Operation of the Asynchronous Remote Volume Mirroring Utility

The Asynchronous Remote Volume Mirroring utility performs steps that generate a recoverable state for multiple mirror volumes at a secondary site. The utility runs these steps to create consistent, recoverable images of a set of volumes:

- 1 On the primary storage array** – The utility reconfigures all of the participating volumes from asynchronous mirroring to synchronous mirroring. This action makes sure that the stream of write operations becomes recoverable on the secondary side.
- 2 On the primary storage array** – The utility polls all of the participating volumes until the associated mirror states all have the Optimal state. In cases where the remote link is slow or the primary host I/O activity is high, one or more mirrors are likely to be in the Unsynchronized state before they transition to the Synchronized state. By waiting until all of the mirrors have Optimal status, the utility makes sure that all of the delta logs for the affected volumes are cleared, and the secondary volumes are recoverable.
- 3 On the primary storage array** – The utility suspends the mirrored pairs for all of the participating volumes. This action causes updates to stop on the secondary side, leaving the secondary volumes in a recoverable state because they were being updated in Synchronous mode immediately before the suspension. By separating the mirrors in this manner, the primary-side applications run faster, while leaving the secondary volumes in a recoverable state. The delta log tracks changes made because of application writes on the primary side while in this state.
- 4 On the secondary storage array** – The utility generates a snapshot of each participating volume on the secondary side, which creates point-in-time images that are recoverable.
- 5 On the primary storage array** – The utility resumes the mirroring operations for all of the participating volumes. This action causes the mirrors to transition to the Synchronized state and start the process of restoring coherency between the primary site and the secondary site.

- 6 On the primary storage array** – The utility reconfigures all of the affected volumes for Asynchronous mode.

Running the Asynchronous Remote Volume Mirroring Utility

The Asynchronous Remote Volume Mirroring utility uses a command line argument that lets you specify the name of a configuration file. The configuration file contains a complete specification of the input parameters that are needed by the utility. To run the utility, enter this syntax:

```
asyncRVMUtil configuration_file -d debug_file
```

where *configuration_file* is the file that you provide as input. The configuration file specifies the Remote Volume Mirroring volumes that you want to synchronize by using the utility. When you create the configuration file, use these conditions to define the volumes in the file:

- All the primary volumes in a volume set must belong to the same storage array.
- The maximum number of volume sets that you can specify in the file is four.
- The maximum number of mirrored pairs that you can specify as part of a consistency group is eight.

The optional parameter, *-d*, lets you specify a file to which you can send information regarding how the utility runs. In this example, the file name is *debug_file*. The debug file contains trace information that can be reviewed by a Customer and Technical Support representative to determine how well the Asynchronous Remote Volume Mirroring utility has run.

NOTE Depending on the location of the configuration file and the debug file, you must specify the complete path with the file name.

To run the Asynchronous Remote Volume Mirroring utility, you must enter the `asyncRVMUtil` command from the command line. Because UNIX operating systems are case sensitive, you must type the command exactly as shown. On Windows operating systems, you can type the command in all uppercase, in all lowercase, or in mixed case.

NOTE To use the Asynchronous Remote Volume Mirroring utility, you must be managing the storage array by using the command line interface, not the graphical user interface of ISSM-EE.

Configuration Utility

The configuration file is an ASCII flat text file that provides the information for the Remote Volume Mirroring synchronization used by the Asynchronous Remote Volume Mirroring utility. The file defines the mirror volume sets to be synchronized. All of the mirror volumes in the volume sets defined in the configuration file are run collectively to create a recoverable image. If any one of the mirrors in the volume set fails, the operation is stopped for this volume set and carried on to the next volume set that is listed in the configuration file.

The configuration file supports this syntax:

```
content ::= {spec}
spec ::= logSpec | volumeSetSpec

logSpec ::= "Log" "{" {logAttribute} "}"
logAttribute ::= fileSpec
fileSpec ::= "file" "=" fileName

volumeSetSpec ::= "VolumeSet" volumeSetName
                "{" {volumeSetAttribute} "}"
volumeSetAttribute ::= timeoutSpec | mirrorSpec

timeoutSpec ::= "OptimalWaitTimeLimit" "=" integer

mirrorSpec ::= "Mirror" "{" {mirrorAttribute} "}"
mirrorAttribute ::= primarySpec | secondarySpec |
                 snapshotSpec

primarySpec ::= "Primary" "=" volumeSpec
secondarySpec ::= "Secondary" "=" volumeSpec
snapshotSpec ::= "Copy" "=" volumeSpec
volumeSpec ::= storageArrayName."volumeUserLabel"
```

In this syntax, items enclosed in double quotation marks (“ ”) are terminal symbols. Items separated by a vertical bar (|) are alternative values (enter one or the other, but not both). Items enclosed in curly braces ({ }) are optional (you can use the item zero or more times).

These definitions are provided for non-terminals in the syntax:

- *integer* – The timeout value must be an integer (decimal digits from 0–9).
- *volumeSetName* – The name of the set of volumes on which you want to run the Asynchronous Remote Volume Mirroring utility.
- *fileName* – The name of a file, using characters and conventions that are appropriate for the system on which the application is running.
- *storageArrayName* – The label that you have assigned for a storage array, as would be used in the CLI to specify the name of the storage array.
- *volumeUserLabel* – The label that you have assigned for a volume that uniquely identifies the volume within the storage array.

NOTE Names and labels can be any characters that are defined as appropriate for your operating system. The maximum length for a name or label is 30 characters. If the name or label contains special characters (as defined by the operating system) or period characters, you must enclose the name or label in double quotation marks (“ ”). You can, optionally, enclose the name or label in double quotation marks at any time.

The following items are considered syntax errors:

- More than one `logSpec` command in the input file
- Zero or more than one `fileSpec` attribute in a `logSpec` command (you must include exactly one `fileSpec` attribute in the `logSpec` command)
- More than one `timeoutSpec` attribute in a `volumeSetSpec` command
- Zero or more than one `primarySpec` attribute in a `mirrorSpec` command (you must include exactly one `primarySpec` attribute in the `mirrorSpec` command)
- Zero or more than one `secondarySpec` attribute in a `mirrorSpec` command (you must include exactly one `secondarySpec` attribute in the `mirrorSpec` command)
- Zero or more than one `snapshotSpec` attribute in a `mirrorSpec` command (you must include exactly one `snapshotSpec` attribute in the `mirrorSpec` command)

IMPORTANT In the Asynchronous Remote Volume Mirroring utility configuration file, you must specify the primary volume, the secondary volume, and the copy (snapshot) volume. The utility does not make sure that the secondary volume is correct for the Remote Volume Mirroring relationship. The utility also does not make sure that the snapshot volume is actually a snapshot for the secondary volume. *You must make sure that these volumes are correct.* If the volumes are not correct, the utility will run, but the volumes will not be consistent. For each mirror, the secondary volume and the copy volume must reside on the same storage array.

This example shows a configuration file for the Asynchronous Remote Volume Mirroring utility.

```
Log{ file="d:\rvm-consistency.log" }
VolumeSet "set1" {
  optimalWaitTimeLimit = 15
  Mirror {
    Primary = LosAngelesArray.PayrollVolume
    Secondary = NewYorkArray.PayrollVolume
    Copy = NewYorkArray.PayrollVolumeImage
  }
  Mirror {
    Primary = LosAngelesArray.PayrollVolume
    Secondary = BostonArray.PayrollVolume
    Copy = BostonArray.PayrollVolumeImage
  }
}

VolumeSet "set2" {
  Mirror {
    Primary = BostonArray.HRVolume
    Secondary = LosAngelesArray.HRVolume
    Copy = LosAngelesArray.HRVolumeImage
  }
}
```