

IBM System Storage DS3000, DS4000, and DS5000



Command Line Interface and Script Commands Programming Guide

Note

Before using this information and the product it supports, read the information in "Notices" on p. 263.

Second Edition

This edition applies to DS5000 Storage Manager Version 10.30, DS4000 Storage Manager Version 10.15, and DS3000 Storage Manager Version 2.17 software release levels and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2008.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	vii
--------------------------	------------

Tables	ix
-------------------------	-----------

About this document xi

DS3000, DS4000, and DS5000 software and firmware support	xi
Who should read this document	xi
Notices used in this document	xii
Getting information, help, and service	xii
Before you call	xii
Using the documentation.	xii
Finding Storage Manager software, controller firmware, and readme files	xii
Essential Web sites for DS3000, DS4000, and DS5000 support information	xiii
Software service and support	xiv
Hardware service and support.	xiv
Fire suppression systems	xiv
How to send your comments	xiv

Trademarks xvii

Chapter 1. About the Command Line

Interface 1

How to Use the Command Line Interface.	1
Usage Notes	1
CLI Commands	2
Command Line Terminals	3
Formatting Considerations.	6
Detailed Error Reporting	7
Exit Status	7
Usage Examples	8

Chapter 2. About the Script Commands 11

Script Command Structure	11
Script Command Synopsis	12
Recurring Syntax Elements	14
Usage Guidelines	17
Adding Comments to a Script File.	17

Chapter 3. Script Commands 19

Naming Conventions	19
Firmware Compatibility Levels	20
Formatting Rules for Script Commands	20
DS3000, DS4000, and DS5000 support and minimum firmware levels	21
Activate Enhanced Remote Mirroring Feature	31
Activate Storage Subsystem Firmware	33
Autoconfigure Storage Subsystem	33
Autoconfigure Storage Subsystem Hot Spares	35
Check Logical Drive Parity	35
Check Remote Mirror Status.	36
Clear Drive Channel Statistics	36

Clear Logical Drive Reservations	37
Clear Logical Drive Unreadable Sectors	37
Clear Storage Subsystem Configuration	38
Clear Storage Subsystem Event Log	38
Clear Storage Subsystem Firmware Pending Area.	39
Create Array	39
Create FlashCopy Logical Drive	41
Create Host	44
Create Host Group	44
Create Host Port.	45
Create iSCSI Initiator	46
Create RAID Logical Drive (Automatic Drive Select)	46
Create RAID Logical Drive (Free Capacity Base Select)	48
Create RAID Logical Drive (Manual Drive Select)	50
Create Remote Mirror	52
Create Volume Copy	54
Deactivate Remote Mirror	55
Delete Array	55
Delete Host	55
Delete Host Group	55
Delete Host Port.	56
Delete iSCSI Initiator	56
Delete LogicalDrive.	57
Diagnose Controller	57
Diagnose Remote Mirror	58
Disable Storage Subsystem Feature	59
Download Drive Firmware	59
Download Environmental Card Firmware	60
Download Storage Subsystem Firmware/NVSRAM	61
Download Storage Subsystem Drive Firmware	61
Download Storage Subsystem NVSRAM.	62
Enable Controller	62
Enable Storage Subsystem Feature Key	63
Recopy Volume Copy	63
Recover RAID Logical Drive.	64
Re-create Enhanced Remote Mirroring Repository Logical Drive.	65
Re-create FlashCopy	67
Re-create FlashCopy Collection	68
Remove Logical Drive LUN Mapping	68
Remove Remote Mirror	69
Remove Volume Copy.	69
Repair Logical Drive Parity	70
Replace Drive	70
Reset Controller	71
Reset Storage Subsystem Battery Install Date	71
Reset Storage Subsystem Diagnostic Data	72
Reset Storage Subsystem iSCSI Baseline	72
Reset Storage Subsystem Logical Drive Distribution	72
Reset Storage Subsystem RLS Baseline	73
Reset Storage Subsystem SOC Baseline	73
Resume Remote Mirror	73
Revive Array	74
Revive Drive	74
Save Controller NVSRAM	74

Save Drive Channel Fault Isolation Diagnostic Status	75	Show Logical Drive Performance Statistics	114
Save Drive Log	75	Show Logical Drive Reservations	115
Save Storage Subsystem Configuration	75	Show Storage Subsystem	115
Save Storage Subsystem Diagnostic Data	76	Show Storage Subsystem Auto Configure	116
Save Storage Subsystem Events	77	Show Storage Subsystem Host Topology	117
Save Storage Subsystem iSCSI Statistics	77	Show Storage Subsystem LUN Mappings	118
Save Storage Subsystem Performance Statistics	78	Show Storage Subsystem Negotiation Defaults	118
Save Storage Subsystem RLS Counts	78	Show Storage Subsystem Unreadable Sectors	118
Save Storage Subsystem SOC Counts	79	Show String	119
Save Storage Subsystem State Capture	79	Show Unconfigured iSCSI Initiators	119
Save Storage Subsystem Support Data	79	Show VolumeCopy	119
Set Array	80	Show VolumeCopy Source Candidates	120
Set Array Forced State	81	Show VolumeCopy Target Candidates	120
Set Controller	81	Start Array Defragment	120
Set Controller Service Action Allowed Indicator	84	Start Array Export	121
Set Drive Channel Status	84	Start Array Import	121
Set Drive Hot Spare	85	Start Array Locate	122
Set Drive Service Action Allowed Indicator	85	Start Drive Channel Fault Isolation Diagnostics	122
Set Drive State	86	Start Drive Channel Locate	123
Set Enclosure Alarm	86	Start Drive Initialize	123
Set Enclosure Identification	86	Start Drive Locate	124
Set Enclosure Service Action Allowed Indicator	87	Start Drive Reconstruction	124
Set FlashCopy Logical Drive	88	Start Enclosure Locate	124
Set Host	89	Start Enhanced Remote Mirroring Synchronization	124
Set Host Channel	90	Start iSCSI DHCP Refresh	125
Set Host Group	91	Start Logical Drive Initialization	125
Set Host Port	91	Start Storage Subsystem Locate	125
Set iSCSI Initiator	92	Stop Array Locate	126
Set iSCSI Target Properties	92	Stop Drive Channel Fault Isolation Diagnostics	126
Set Logical Drive	93	Stop Drive Channel Locate	126
Set Remote Mirror	97	Stop Drive Locate	127
Set Session	99	Stop Enclosure Locate	127
Set Storage Subsystem	100	Stop FlashCopy	127
Set Storage Subsystem Enclosure Positions	102	Stop iSCSI Session	127
Set Storage Subsystem ICMP Response	102	Stop Storage Subsystem Drive Firmware Download	128
Set Storage Subsystem iSNS Server IPv4 Address	103	Stop Storage Subsystem Locate	128
Set Storage Subsystem iSNS Server IPv6 Address	103	Stop VolumeCopy	128
Set Storage Subsystem iSNS Server Listening Port	104	Suspend Remote Mirror	129
Set Storage Subsystem iSNS Server Refresh	104	Script Commands Listed by Function	129
Set Storage Subsystem Learn Cycle	105	Controller Commands	129
Set Storage Subsystem Redundancy Mode	105	Disk Drive Commands	130
Set Storage Subsystem Time	106	Enclosure Commands	130
Set Unnamed Discovery Session	106	Host Topology Commands	131
Set VolumeCopy	106	iSCSI Commands	131
Show Array	107	Enhanced Remote Mirroring Commands	132
Show Array Export Dependencies	107	Session Command	132
Show Array Import Dependencies	108	FlashCopy Commands	132
Show Controller	108	Storage Subsystem Commands	132
Show Controller NVSRAM	109	Uncategorized Commands	134
Show Current iSCSI Sessions	110	Logical Drive Commands	134
Show Drive	110	VolumeCopy Commands	135
Show Drive Channel Stats	111	Array Commands	135
Show Drive Download Progress	111		
Show Enhanced Remote Mirroring Logical Drive Candidates	112	Chapter 4. Configuring a Storage Subsystem	137
Show Enhanced Remote Mirroring Logical Drive Synchronization Progress	112	Configuration Concepts	138
Show Host Ports	113	Controllers	138
Show Logical Drive	113	Disk Drives	141
Show Logical Drive Action Progress	114	Hot Spares	142
		Arrays	143
		Logical Drives	143

RAID Levels	145
Hosts	147
Host Groups	147
Host Bus Adapter Host Ports	147
Logical Unit Numbers	148
Configuring a Storage Subsystem.	148
Determining What Is on Your Storage Subsystem	148
Clearing the Configuration	151
Using the Auto Configure Command	151
Using the Create LogicalDrive Command	153
Modifying Your Configuration.	156
Setting the Controller Clocks	156
Setting the Storage Subsystem Password	156
Setting the Storage Subsystem Host Type	157
Setting the Storage Subsystem Cache	157
Setting the Modification Priority	161
Assigning Global Hot Spares	161
Saving a Configuration to a File	161

Chapter 5. Using the FlashCopy Premium Feature 163

How FlashCopy Works	163
Creating a FlashCopy Logical Drive	164
Creating a FlashCopy Logical Drive with User-Assigned Disk Drives	164
Creating a FlashCopy Logical Drive with Software-Assigned Disk Drives	165
Creating a FlashCopy Logical Drive by Specifying a Number of Disk Drives.	166
User-Defined Parameters	166
FlashCopy Logical Drive Names and FlashCopy Repository Logical Drive Names	168
Changing FlashCopy Logical Drive Settings	168
Stopping, Restarting, and Deleting a FlashCopy Logical Drive	169

Chapter 6. Using the Enhanced Remote Mirroring Premium Feature . . 171

How Enhanced Remote Mirroring Works	171
Mirror Repository Logical Drives	172
Mirror Relationships	172
Data Replication	173
Link Interruptions or Secondary Logical Drive Errors	173
Resynchronization	174
Creating a Remote-Mirror Pair	174
Performance Considerations	175
Enabling the Enhanced Remote Mirroring Feature	175
Activating the Enhanced Remote Mirroring Feature	175
Determining Candidates for a Remote-Mirror Pair	177
Creating a Remote-Mirror Pair	178
Changing Enhanced Remote Mirroring Settings	179
Suspending and Resuming a Mirror Relationship	179
Removing a Mirror Relationship	180
Deleting a Primary Logical Drive or a Secondary Logical Drive	180

Disabling the Enhanced Remote Mirroring Feature	181
Deactivating the Enhanced Remote Mirroring Feature	181
Interaction with Other Features	181
Storage Partitioning	181
FlashCopy Logical Drives	181
VolumeCopy	182
Dynamic Logical Drive Expansion	182
Asynchronous Enhanced Remote Mirroring Utility	182
Description of the Asynchronous Enhanced Remote Mirroring Utility	182
Operation of the Asynchronous Enhanced Remote Mirroring Utility	183
Running the Asynchronous Enhanced Remote Mirroring Utility	184
Configuration Utility	184

Chapter 7. Using the VolumeCopy Premium Feature 187

How VolumeCopy Works	187
Source Logical Drive	187
Target Logical Drive	188
VolumeCopy and Persistent Reservations	188
Storage Subsystem Performance	189
Restrictions	189
VolumeCopy Commands	189
Creating a VolumeCopy	190
Enabling the VolumeCopy Feature	190
Determining VolumeCopy Candidates	190
Creating a VolumeCopy	191
Viewing VolumeCopy Properties	192
Changing VolumeCopy Settings	192
Recopying a Logical Drive	193
Stopping a VolumeCopy.	194
Removing Copy Pairs	194
Interaction with Other Features	195
Storage Partitioning	195
FlashCopy Logical Drives	195
Enhanced Remote Mirroring	196

Chapter 8. Maintaining a Storage System 199

Routine Maintenance	199
Running a Media Scan	199
Running a Redundancy Check	200
Resetting a Controller	200
Enabling a Controller Data Transfer	200
Resetting the Battery Age	201
Removing Persistent Reservations	201
Synchronizing the Controller Clocks.	201
Locating Disk Drives	201
Relocating an Array	201
Performance Tuning	203
Monitoring the Performance	203
Changing the RAID Levels	204
Changing the Segment Size.	204
Changing the Cache Parameters	204
Defragmenting an Array.	205
Troubleshooting and Diagnostics	205
Collecting All Support Data	205

Collecting Drive Data	207
Diagnosing a Controller	207
Running Read Link Status Diagnostics	208
Collecting Switch-on-a-Chip Error Statistics	210
Recovery Operations	211
Setting the Controller Operational Mode	211
Changing the Controller Ownership	212
Initializing a Drive	212
Reconstructing a Drive	212
Initializing a Logical Drive	213
Redistributing Logical Drives	213
Replacing CRUs	213

Chapter 9. Examples of Information Returned by the Show Commands . . . 215

Show Storage Subsystem Command	215
Show Controller NVSRAM	238
Show Drive	245
Show Drive Channel Status	247
Show Logical Drives	249

Chapter 10. Example Script Files . . . 253

Configuration Script Example 1	253
Configuration Script Example 2	254

Chapter 11. Deprecated Commands and Parameters 257

Deprecated Commands	257
Deprecated Parameters	261

Appendix. Additional Documentation 263

Notices 267

Glossary 269

Index 281

Figures

1. Host and Storage Relationship	138	4. DS4200 and DS4700 Host Ports	140
2. DS3200 Host Ports	140	5. DS4800 Host Ports	141
3. DS3400 Host Ports	140		

Tables

1. Command Name Syntax Conventions	2	17. Values for the cacheFlushModifier Parameter	159
2. Command Line Terminals	3	18. Components of a FlashCopy Logical Drive	163
3. Exit Status	7	19. FlashCopy Logical Drive Commands	164
4. Configuration and Management Operations	11	20. FlashCopy Logical Drive Parameters	167
5. Object Types and Identifiers	12	21. Maximum Number of Defined Mirrors	171
6. General Form of the Script Commands	13	22. VolumeCopy Commands.	189
7. Recurring Syntax Elements	14	23. Information About Storage Subsystem Performance	203
8. DS3000, DS4000, and DS5000 commands, listed alphabetically	22	24. Support Data for the Storage Subsystem	205
9. Valid Characters for a CHAP Secret	90	25. RLS Baseline Data for the Storage Subsystem	208
10. Host Ports and the Type of Host Interfaces on Controller Modules	139	26. Deprecated Commands	257
11. Maximum Number of Disk Drives	142	27. Deprecated Parameters	261
12. Maximum Number of Disk Drives in a Array Based on Capacity	143	28. DS3000 Storage Subsystem document titles by user tasks	263
13. Logical Drive Specifications by Supported Controller Model	145	29. DS4000 Storage Subsystem document titles by user tasks	263
14. RAID Level Configurations	145	30. Storage Expansion Enclosure document titles by user tasks.	264
15. Maximum Number of HBA Host Ports per Controller.	147	31. DS3000 and DS4000–related document titles by user tasks.	264
16. Default Values for Segment Size and Cache Read Prefetch	153		

About this document

This document is a programming guide that describes the command line interface and the script commands for DS3000, DS4000™, and DS5000 Storage Manager software to configure, operate, and maintain a storage subsystem. This document describes all of the script commands, explains the purpose of the commands, shows the complete syntax of the commands, and defines the parameters for the commands.

For information about using the Storage Manager graphical user interface (GUI) or configuring your host operating systems, see the following documents:

- *IBM System Storage DS3000 Storage Manager Installation and Host Support Guide*
- *IBM System Storage DS4000 Storage Manager Installation and Host Support Guide*
- *IBM System Storage DS5000 Storage Manager Installation and Host Support Guide*

This document does not cover hardware installation or integration. For information about these topics, see the Installation and Support Guide that is appropriate for your storage subsystem. See “Additional Documentation,” on page 263 for a list of these documents.

Check the DS3000, DS4000, or DS5000 readme files for the most up-to-date information regarding hardware, software, or firmware products that might not be described in this document.

DS3000, DS4000, and DS5000 software and firmware support

The command line interface and the script commands described in this document are supported with the following software and firmware levels:

- **DS3000:** DS3000 Storage Manager Version 2.xx software in conjunction with firmware 6.xx.xx.xx and later.
- **DS4000:** DS4000 Storage Manager Version 10.xx software in conjunction with the DS4000 controller firmware version 7.xx.xx.xx and later.

Note: Some of the script commands are also supported in earlier versions of DS4000 controller firmware. Please see “DS3000, DS4000, and DS5000 support and minimum firmware levels” on page 21 for the minimum controller firmware that is required to support each script command. This table also shows which commands are supported with the DS3000, DS4000, or DS5000 products only, two of the three, or all three.

- **DS5000:** DS5000 Storage Manager Version 10.30 software in conjunction with the DS5000 controller firmware version 7.30.

Who should read this document

This document assumes that the user has a knowledge of basic storage area network (SAN) hardware and installation skills.

This document is intended for system operators, system administrators, and service personnel who are responsible for operating, maintaining, troubleshooting, and servicing a DS3000, DS4000, or DS5000 storage subsystem. Users must be familiar with computer system operation, maintenance, and repair. In addition, they should understand disk storage technology, Redundant Array of Independent Disks (RAID) concepts, networking, and Fibre Channel and iSCSI technologies. The reader must have a basic knowledge of SAN hardware functionality (controllers, drives, and hosts) and SAN cabling.

Notices used in this document

This document contains the following notices that are designed to highlight key information:

Note: These notices provide tips, guidance, or advice.

Important:

These notices provide information or advice that might help you avoid inconvenient or problem situations.

Attention:

These notices indicate possible damage to programs, devices or data. An attention notice is placed just before the instruction or situation in which damage could occur.

Getting information, help, and service

If you need help, service, or technical assistance or just want more information about IBM® products, you will find a wide variety of sources available from IBM to assist you. This section contains information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your system, and whom to call for service, if it is necessary.

Before you call

Before you call, take these steps to try to solve the problem yourself:

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system is turned on.
- Use the troubleshooting information in your system documentation, and use the diagnostic tools that come with your system.
- Check for technical information, hints, tips, and new device drivers at the IBM support Web site pages that are listed in this section.
- Use an IBM discussion forum on the IBM Web site to ask questions.

You can solve many problems without outside assistance by following the troubleshooting procedures that IBM provides in the Storage Manager online help or in the documents that are provided with your system and software. The information that comes with your system also describes the diagnostic tests that you can perform. Most subsystems, operating systems, and programs come with information that contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the information for the operating system or program.

Using the documentation

Information about your IBM system and preinstalled software, if any, is available in the documents that come with your system. This includes printed books, online documents, readme files, and help files. See the troubleshooting information in your system documentation for instructions for using the diagnostic programs. The troubleshooting information or the diagnostic programs might tell you that you need additional or updated device drivers or other software.

Finding Storage Manager software, controller firmware, and readme files

Storage Manager software and controller firmware are available on the product CD and can also be downloaded from the Web. Storage Manager readme files are also found on the Web.

Important: Before you install Storage Manager software, consult the readme file for your host operating system. Updated readme files contain the latest device driver versions, firmware levels, limitations, and other information not found in this document.

1. Go to the following Web site:

www.ibm.com/servers/storage/support/disk/

2. Click the link for your storage subsystem (for example, **DS3400** or **DS4800**).
3. When the support page for your storage subsystem opens, click the **Download** link in the Support & Downloads box. The Software and device drivers page opens.
4. In the Storage Manager section of the table, locate your operating system and version level, and click on the version link in the right hand column. The specific page for your storage subsystem and operating system version opens.
5. In the table under File details, click on the *.txt file link, and the README will open in your Web browser.

Essential Web sites for DS3000, DS4000, and DS5000 support information

The most up-to-date information about DS3000, DS4000, and DS5000 storage subsystems and Storage Manager, including documentation and the most recent software, firmware, and NVSRAM downloads, can be found at the following Web sites.

IBM System Storage™ Disk Storage Systems

Find links to software and firmware downloads, readmes, and support pages for all IBM System Storage disk storage systems, including DS3000, DS4000, and DS5000:

www.ibm.com/systems/support/storage/disk

IBM System Storage Interoperation Center (SSIC)

Find technical support information for your specific DS3000, DS4000, or DS5000 storage subsystem/host configuration, including the latest recommended firmware versions for your system, by using this interactive Web-based utility:

www.ibm.com/systems/support/storage/config/ssic/index.jsp

IBM DS3000, DS4000, or DS5000 and BladeCenter® Premium Feature Activation

Activate a DS3000, DS4000, or DS5000 premium feature by using this Web-based utility:

<https://www-912.ibm.com/PremiumFeatures/jsp/keyPrereq.jsp>

IBM System Storage DS3000, DS4000, and DS5000 Interoperability Support

Find the latest information about operating system and HBA support, clustering support, storage area network (SAN) fabric support, and Storage Manager feature support:

- **DS3000:** www.ibm.com/systems/storage/disk/ds3000/pdf/interop.pdf
- **DS4000 / DS5000:** www.ibm.com/servers/storage/disk/ds4000/interop-matrix.html

Storage Area Network (SAN) Support

Find information about using SAN switches, including links to SAN user guides and other documents:

www.ibm.com/systems/support/storage/san

Support for IBM System p™, AIX 5L™, and Linux® servers

Find the latest support information for System p, AIX®, Linux, BladeCenter, and i5/OS® servers:

www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5000025

Support for IBM System x™ servers

Find the latest support information for System x Intel- and AMD-based servers:

www.ibm.com/systems/support/supportsite.wss/brandmain?brandind=5000008

eServer™ System p and AIX Information Center

Find everything you need to know about using AIX with System p and POWER™ servers:

publib.boulder.ibm.com/infocenter/pseries/index.jsp?

Fix Central

Find fixes and updates for your system's software, hardware, and host operating system:

www.ibm.com/eserver/support/fixes

IBM System Storage products

Find information about all IBM System Storage products:

www.storage.ibm.com

IBM Publications Center

Find IBM publications:

www.ibm.com/shop/publications/order/

Software service and support

Through IBM Support Line, for a fee you can get telephone assistance with usage, configuration, and software problems. For information about which products are supported by Support Line in your country or region, go to the following Web site:

www.ibm.com/services/sl/products

For more information about the IBM Support Line and other IBM services, go to the following Web sites:

- www.ibm.com/services
- www.ibm.com/planetwide

Hardware service and support

You can receive hardware service through IBM Integrated Technology Services or through your IBM reseller, if your reseller is authorized by IBM to provide warranty service. Go to the following Web site for support telephone numbers:

www.ibm.com/planetwide

In the U.S. and Canada, hardware service and support is available 24 hours a day, 7 days a week. In the U.K., these services are available Monday through Friday, from 9 a.m. to 6 p.m.

Fire suppression systems

A fire suppression system is the responsibility of the customer. The customer's own insurance underwriter, local fire marshal, or a local building inspector, or both, should be consulted in selecting a fire suppression system that provides the correct level of coverage and protection. IBM designs and manufactures equipment to internal and external standards that require certain environments for reliable operation. Because IBM does not test any equipment for compatibility with fire suppression systems, IBM does not make compatibility claims of any kind nor does IBM provide recommendations on fire suppression systems.

How to send your comments

Your feedback is important to help us provide the highest quality information. If you have any comments about this document, you can submit them in one of the following ways:

E-mail

Submit your comments by e-mail to:

starpubs@us.ibm.com

Be sure to include the name and order number of the document and, if applicable, the specific location of the text that you are commenting on, such as a page number or table number.

Mail

Send your comments to:

International Business Machines Corporation
Information Development
Department GZW
9000 South Rita Road
Tucson, Arizona
USA
85744-0001

Be sure to include the name and order number of the document and, if applicable, the specific location of the text that you are commenting on, such as a page number or table number.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX
DS4000
eServer
FlashCopy
IBM
POWER
RS/6000
System p
System Storage

Microsoft® Windows® and Windows NT® are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX® is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Chapter 1. About the Command Line Interface

Note: Attention: IBM recommends using the Storage Manager client GUI to manage your storage subsystems. – The command-line interface does not have any mechanisms to prevent you from inadvertently making unwanted changes to the storage subsystem. Because the script commands are capable of damaging a configuration and causing loss of data access if not used correctly, IBM recommends using the Storage Manager client GUI to manage your storage subsystem configurations.

The command-line interface (CLI) is a software tool that lets storage subsystem installers, developers, and engineers configure and monitor storage subsystems. Using the CLI, you can run commands from an operating system prompt, such as the Windows command prompt, a Linux operating system console, or a Solaris operating system console.

Each command performs a specific action for managing a storage subsystem or returning information about the status of a storage subsystem. 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 subsystems. 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 subsystem. You can use the CLI both in installation sites and in development environments.

How to Use the Command Line Interface

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

A CLI command consists of these elements:

- The term `SMcli`
- The storage subsystem identifier
- Parameters
- Script commands

A CLI command takes this form:

```
SMcli storageSubsystem parameters script-commands;
```

- *SMcli* invokes the command line interface.

Note: If you issue the command from the directory or folder that does not contain the `SMcli.exe` program, you need to include the appropriate path: `directoryName/SMcli`.

- *storageSubsystem* is the name or the IP address of the storage subsystem.
- *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 subsystem configuration commands.)

Usage Notes

If you enter `SMcli` and a storage subsystem 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`. 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 SMcli string that does not have the correct syntax, parameter names, options, or terminals, the script engine returns usage information.

CLI Commands

The following table lists the conventions used in the command statement syntax to manage the storage subsystem.

Table 1. Command Name Syntax Conventions

Convention	Definition
(a b c)	Alternative (“a” or “b” or “c”)
<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
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.

```

| SMcli DNS-network-name-or-IP-address [DNS-network-name-or-IP-address]
  [-c "command; {command2};"]
  [-n storage-system-name | -w wwID]
  [-o outputfile] [-p password] [-e] [-S] [-quick]
| SMcli DNS-network-name-or-IP-address [DNS-network-name-or-IP-address]
  [-f scriptfile]
  [-n storage-system-name | -w wwID]
  [-o outputfile] [-p password] [-e] [-S] [-quick]
SMcli (-n storage-system-name | -w wwID)
  [-c "command; {command2};"]
  [-o outputfile] [-p password] [-e] [-S] [-quick]
SMcli (-n storage-system-name -w wwID)
  [-f scriptfile]
  [-o outputfile] [-p password] [-e] [-S] [-quick]
SMcli -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]
SMcli -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]
SMcli (-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]

SMcli -d [-w] [-i] [-s] [-v] [-S]
SMcli -m host-name-or-IP-address -F email-address
  [-g contactInfoFile] [-S]

```

SMcli -A [*host-name-or-IP-address* [*host-name-or-IP-address*]] [-S]

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

SMcli -?

Command Line Terminals

Table 2. Command Line Terminals

Terminal	Definition
<i>host-name-or-IP-address</i>	<p>Specifies either the host name or the Internet Protocol (IP) address (<i>xxx.xxx.xxx.xxx</i>) of an in-band managed storage subsystem or an out-of-band managed storage subsystem.</p> <ul style="list-style-type: none">• If you are managing a storage subsystem 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 subsystem is connected to the host.• If you are managing a storage subsystem 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 subsystem in the Enterprise Management Window, you can specify the storage subsystem by its user-supplied name by using the <i>-n</i> terminal.• If you have previously configured a storage subsystem in the Enterprise Management Window, you can specify the storage subsystem by its World-Wide Identifier (WWID) by using the <i>-w</i> terminal.
-A	<p>Adds a storage subsystem to the configuration file. If you do not follow the -A terminal with a <i>host-name-or-IP-address</i>, auto-discovery scans the local subnet for storage subsystems.</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.
-c	<p>Indicates that you are entering one or more script commands to run on the specified storage subsystem. 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.</p>
-d	<p>Shows the contents of the script configuration file. The file content takes this format: <i>storage-system-name host-name1 host-name2</i></p>
-e	<p>Runs the commands without performing a syntax check first.</p>
-F (uppercase)	<p>Specifies the email address from which all alerts will be sent.</p>

Table 2. Command Line Terminals (continued)

Terminal	Definition
-f (lowercase)	<p>Specifies a file name that contains script commands that you want to run on the specified storage subsystem. (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.)</p> <p>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 set session errorAction=stop command in the script file.</p>
-g	<p>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 userdata.txt file exists.</p>
-h	<p>Specifies the host name that is running the SNMP agent to which the storage subsystem is connected. Use this terminal with the -a terminal and the -x terminal.</p>
-I	<p>Specifies the type of information to be included in the email alert notifications. You can select these values:</p> <ul style="list-style-type: none"> • <i>eventOnly</i> • <i>profile</i> • <i>supportBundle</i>
-i	<p>Shows the IP address of the known storage subsystems. Use this terminal with the -d terminal. The file contents takes this format: storage-system-name IP-address1 IPaddress2</p>
-m	<p>Specifies the host name or the IP address of the email server from which email alert notifications are sent.</p>
-n	<p>Specifies the name of the storage subsystem 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 subsystem, you must use the -n terminal if more than one storage subsystem is connected to the host at the specified address. The storage subsystem name is required when the <i>host-name-or-IP-address</i> is not used. The name of the storage subsystem 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 subsystem.</p>
-o	<p>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>
-p	<p>Specifies the password for the storage subsystem on which you want to run commands. A password is not necessary under these conditions:</p> <ul style="list-style-type: none"> • A password has not been set on the storage subsystem. • The password is specified in a script file that you are running. • You specify the password by using the -c terminal and the set session password= <i>password</i> command.

Table 2. Command Line Terminals (continued)

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 <i>eventOnly</i>, the only valid value for the -q terminal is <i>everyEvent</i>. If you set the -I terminal to either the <i>profile</i> value or the <i>supportBundle</i> 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"> • <i>everyEvent</i> – 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 recreate flashCopy LogicalDrive 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 subsystems that are managed by a specific storage management station. Use the -r terminal with the -a terminal or the -x terminal. For storage subsystems that are managed out-of-band, use the <i>direct_sa</i> value. For storage subsystems that are managed in-band, use the <i>host_sa</i> value.</p>
-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 • SMcli completed successfully
-s (lowercase)	<p>Shows the alert settings in the configuration file when used with the -d terminal.</p>
-v	<p>Show the current global status of the known devices in a configuration file when used with the -d terminal.</p>
-w	<p>Specifies the WWID of the storage subsystem. This terminal is an alternate to the -n terminal. Use the -w terminal with the -d terminal to show the WWIDs of the known storage subsystems. The file content take this format: <i>storage-system-name world-wide-ID IPaddress1 IP-address2</i>.</p>
-X (uppercase)	<p>Deletes a storage subsystem from a configuration.</p>

Table 2. Command Line Terminals (continued)

Terminal	Definition
-x (lowercase)	Removes an SNMP trap destination or an email address alert destination. The <i>community</i> is the SNMP community name for the trap, and the <i>host</i> 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 storageSubsystem userLabel=\“Engineering\”;
```

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

```
-n \“My\”_StorageSubsystem
```

where “My”_StorageSubsystem is the name of the storage subsystem.

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 subsystem name to “Finance Subsystem”:

```
-c “set storageSubsystem userLabel=\“Finance\”Subsystem\”;
```

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

```
-c ‘set storageSubsystem userLabel=“Engineering”;
```

```
-n “My”_StorageSubsystem
```

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 subsystem 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 subsystem to FINANCE_|_PAYROLL, enter the following string:

```
-c set storageSubsystem 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 `excrpt.txt`. The CLI tries to place the `excrpt.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 `excrpt.txt` file in the same directory from which the CLI is running. You cannot change the file name or the location. The `excrpt.txt` file is overwritten every time that an exception occurs. If you want to save the information in the `excrpt.txt` file, you must copy the information to a new file or a new directory.

Note: In Windows, this is specified by the registry key:

`HKLM\SOFTWARE\storage\SMclient\SunJVMOption1`

In a typical installation with no changes in the default installation directory, the location is:

- **Windows operating systems:** `c:\Program Files\IBM_DS...\client\data`
- **UNIX-based operating systems:** `/var/opt/SM`

Exit Status

The following table 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.

Table 3. Exit Status (continued)

Status Value	Meaning
4	A storage subsystem was not at the specified address.
5	Addresses specify different storage subsystems.
6	A storage subsystem name does not exist for the host agent that is connected.
7	The storage subsystem name was not at the specified address.
8	The storage subsystem name was not in the configuration file.
10	A management class does not exist for the storage subsystem.
11	A storage subsystem 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 subsystem.
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.
22	The storage subsystem 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 userdata.txt file exists.
32	An invalid <i>-I</i> value in the email alert notification was specified.
33	An invalid <i>-f</i> 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 3, "Script Commands," on page 19.

This example shows how to change the name of a storage subsystem. The original name of the storage subsystem is Payroll_Subsystem. The new name is Finance_Subsystem. The storage subsystem is managed in-band through the host name ICTSANT.

- **Windows operating system:**

```
SMcli ICTSANT -n "Payroll_Subsystem" -c "set storageSubsystem userLabel=\"Finance_Subsystem\";"
```

- **UNIX operating system:**

```
SMcli ICTSANT -n 'Payroll_Subsystem' -c 'set storageSubsystem userLabel="Finance_Subsystem";'
```

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

- **Windows operating system:**

```
SMcli finance1 finance2 -c 'set session password=\"TestSubsystem\"; delete logicalDrive [\"Stocks_<_Bonds\"]; create logicalDrive driveCount[3] RAIDLEVEL=3 capacity=10GB userLabel=\"Finance\"; show storageSubsystem healthStatus;'
```

- **UNIX operating system:**

```
SMcli finance1 finance2 -c 'set session password="TestSubsystem"; delete logicalDrive ["Stocks_<Bonds"]; create logicalDrive driveCount[3] RAIDLEVEL=3 capacity=10GB userLabel="Finance"; show storageSubsystem healthStatus;'
```

This example shows how to run commands in a script file named *scriptfile.scr* on a storage subsystem 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 -n Example -f scriptfile.scr -e
```

This example shows how to run commands in a script file named *scriptfile.scr* on a storage subsystem named Example. In this example, the storage subsystem is protected by the password MySubsystem. 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_Subsystem" -o output.txt
```

- **UNIX operating system:**

```
SMcli -n Example -f scriptfile.scr -p 'My_Subsystem' -o output.txt
```

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

```
SMcli -d
```

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

```
SMcli -d -i
```

Chapter 2. About the Script Commands

You can use the script commands to configure and manage a storage subsystem. You can enter individual script commands, or you can run a file of script commands.

To invoke the Storage Manager Command-Line Interface (SMcli), use the SMcli command.

Note: When you enter an individual script command, you include it as part of a SMcli command. When you run a file of script commands, you include the file name as part of a SMcli 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 subsystem

At the storage subsystem level, the storage subsystem controllers run script commands.

The script engine and the script commands support the storage subsystem configuration and management operations that are listed in the following table:

Table 4. Configuration and Management Operations

Operation	Activities
Cache configuration	Controlling all cache parameters, both at the storage subsystem level and the individual logical drive level
Logical drive configuration and array configuration	Creating, deleting, and setting the reconstruction priority control; labeling; setting drive composition when creating logical drives; setting the segment size; and setting the media scan control
Disk drive configuration	Assigning hot spares
Controller configuration	Defining logical drive ownership, changing mode settings, defining network settings, and setting host channel IDs
General storage subsystem 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
NVSRAM configuration	Downloading and modifying the user configuration region at the bit level and the byte level, showing nonvolatile static random access memory (NVSRAM) values
Product identification	Retrieving the enclosure profile display data
Battery management	Setting the battery installation date
Firmware management	Downloading controller firmware, the environmental services monitor (ESM) firmware, and the disk 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 subsystem 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 subsystem (for example, **allLogicalDrives**). To perform a command on a specific object, use square brackets to identify the object (for example, **logicalDrive [engineering]**). Specify a subset of objects with a list of identifiers in square brackets (for example, **logicalDrives [sales engineering marketing]**). A qualifier is required if you want to include additional information to describe the objects.

The following table 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	Enclosure ID and slot ID
drivechannel	Drive channel identifier
host	User label
hostchannel	Host channel identifier
hostgroup	User label
hostport	User label
remote mirror	Primary logical drive user label
flashcopy	Logical drive user label
storagesubsystem	Not applicable
enclosure	Enclosure ID
logicalDrive	Logical drive user label or logical drive World-Wide Identifier (WWID) (set command only)
volumecopy	Target logical drive user label and, optionally, the source logical drive user label
array	Array number

Statement data is in the form of:

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

Script Command Synopsis

Because you can use the script commands to define and manage the different aspects of a storage subsystem (such as host topology, disk drive configuration, controller configuration, logical drive definitions, and array 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 subsystem. The following table 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
activate <i>object</i> { <i>statement-data</i> }	Sets up the environment so that an operation can take place or performs the operation if the environment is already set up correctly.
autoConfigure storageSubsystem { <i>statement-data</i> }	Automatically creates a configuration that is based on the parameters that are specified in the command.
check <i>object</i> { <i>statement-data</i> }	Starts an operation to report on errors in the object, which is a synchronous operation.
clear <i>object</i> { <i>statement-data</i> }	Discards the contents of some attribute of an object. This operation is destructive and cannot be reversed.
create <i>object</i> { <i>statement-data</i> }	Creates an object of the specified type.
deactivate <i>object</i> { <i>statement-data</i> }	Removes the environment for an operation.
delete <i>object</i>	Deletes a previously created object.
diagnose <i>object</i> { <i>statement-data</i> }	Runs a test and shows the results.
disable <i>object</i> { <i>statement-data</i> }	Prevents a feature from operating.
download <i>object</i> { <i>statement-data</i> }	Transfers data to the storage subsystem or to the hardware that is associated with the storage subsystem.
enable <i>object</i> { <i>statement-data</i> }	Sets a feature operate.
recopy <i>object</i> { <i>statement-data</i> }	Restarts a VolumeCopy operation by using an existing VolumeCopy pair. You can change the parameters before the operation is restarted.
recover <i>object</i> { <i>statement-data</i> }	Re-creates an object from saved configuration data and the statement parameters. (This command is similar to the create command.)
recreate <i>object</i> { <i>statement-data</i> }	Restarts a flashcopy operation by using an existing flashcopy logical drive. You can change the parameters before the operation is restarted.
remove <i>object</i> { <i>statement-data</i> }	Removes a relationship from between objects.
repair <i>object</i> { <i>statement-data</i> }	Repairs errors found by the check command.
reset <i>object</i> { <i>statement-data</i> }	Returns the hardware or an object to an initial state.
resume <i>object</i>	Starts a suspended operation. The operation starts where it left off when it was suspended.
revive <i>object</i>	Forces the object from the Failed state to the Optimal state. Use this command only as part of an error recovery procedure.
save <i>object</i> { <i>statement-data</i> }	Writes information about the object to a file.
set <i>object</i> { <i>statement-data</i> }	Changes object attributes. All changes are completed when the command returns.
show <i>object</i> { <i>statement-data</i> }	Shows information about the object.
start <i>object</i> { <i>statement-data</i> }	Starts an asynchronous operation. You can stop some operations after they have started. You can query the progress of some operations.
stop <i>object</i> { <i>statement-data</i> }	Stops an asynchronous operation.

Table 6. General Form of the Script Commands (continued)

Syntax	Description
<code>suspend <i>object</i> {<i>statement-data</i>}</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 3, “Script Commands,” on page 19

The following table 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
<i>raid-level</i>	(0 1 3 5 6)
<i>repository-raid-level</i>	(1 3 5 6)
<i>capacity-spec</i>	<i>integer-literal</i> [KB MB GB TB Bytes]
<i>segment-size-spec</i>	<i>integer-literal</i>
<i>boolean</i>	(TRUE FALSE)
<i>user-label</i>	<i>string-literal</i>
<i>user-label-list</i>	<i>user-label</i> { <i>user-label</i> }
<i>create-raid-logicalDrive-attr-value-list</i>	<i>create-raid-logicalDrive-attribute-value-pair</i> { <i>create-raid-logicalDrive-attribute-value-pair</i> }
<i>create-raid-logicalDrive-attribute-value-pair</i>	capacity= <i>capacity-spec</i> owner=(a b) cacheReadPrefetch=(TRUE FALSE) segmentSize= <i>integer-literal</i>
<i>noncontroller-enclosureID</i>	(0-99)
<i>slotID</i>	(1-32)
<i>portID</i>	(0-127)
<i>drive-spec</i>	<i>enclosureID</i> , <i>slotID</i>
<i>drive-spec-list</i>	<i>drive-spec</i> { <i>drive-spec</i> }
<i>enclosureID-list</i>	<i>enclosureID</i> { <i>enclosureID</i> }

Table 7. Recurring Syntax Elements (continued)

Recurring Syntax	Syntax Value
<i>hex-literal</i>	<i>0xhexadecimal-literal</i>
<i>array-number</i>	<i>integer-literal</i>
<i>filename</i>	<i>string-literal</i>
<i>error-action</i>	(stop continue)
<i>drive-channel-identifier</i>	(1 2 3 4)
<i>drive-channel-identifier-list</i>	<i>drive-channel-identifier</i> { <i>drive-channel-identifier</i> }
<i>host-channel-identifier</i> (all controller modules except the DS4800 (Models 82, 84, 88)-series controller modules)	(a1 a2 b1 b2)
<i>host-channel-identifier</i> (DS4800 (Models 82, 84, 88)-series controller modules only)	(a1 a2 a3 a4) (b1 b2 b3 b4)
<i>drive-type</i>	(fibre SATA SAS)
<i>feature-identifier</i>	(storagePartition2 storagePartition4 storagePartition8 storagePartition16 storagePartition64 storagePartition128 storagepartition256 storagepartition512 storagePartitionMax flashCopy remoteMirror volumeCopy)
<i>repository-spec</i>	<i>instance-based-repository-spec</i> <i>count-based-repository-spec</i>
<i>instance-based-repository-spec</i>	<p>repositoryRAIDLevel=<i>repository-raid-level</i> repositoryDrives=(<i>drive-spec-list</i>) [enclosureLossProtect=<i>boolean</i>¹] repositoryArray=<i>array-number</i> freeCapacityArea=<i>integer-literal</i>²</p> <p>Specify repositoryRAIDLevel with repositoryDrives. Do not specify the RAID level or the disk drives with the array. Do not set enclosureLossProtect when you specify a array.</p>
<i>count-based-repository-spec</i>	<p>repositoryRAIDLevel=<i>repository-raid-level</i> repositoryDriveCount=<i>integer-literal</i> [driveType=<i>drive-type</i>³] [enclosureLossProtect=<i>boolean</i>⁴]</p>
<i>wwID</i>	<i>string-literal</i>
<i>nvsram-offset</i>	<i>hex-literal</i>
<i>host-type</i>	<i>string-literal</i> <i>integer-literal</i>

Table 7. Recurring Syntax Elements (continued)

Recurring Syntax	Syntax Value
<i>nvsram-byte-setting</i>	<i>nvsram-value-> 0xhexadecimal integer-literal</i>
<i>nvsram-bit-setting</i>	<i>nvsram-mask, nvsram-value-> 0xhexadecimal, 0xhexadecimal integer-literal</i>
<i>ip-address</i> for IPv4	Four groups of four numeric characters (0-9) separated by period "." (0255).(0255).(0255).(0255)
<i>ip-address</i> for IPv6	Eight groups of four hexadecimal characters (0-9 and A-F) separated by semicolon ":" (02AF):(02AF):(02AF):(02AF): (02AF):(02AF):(02AF):(02AF)
<i>autoconfigure-logicaldrive-attr-value-list</i>	<i>autoconfigure-logicaldrive-attr-value-pair</i> { <i>autoconfigure-logicaldrive-attr-value-pair</i> }
<i>autoconfigure-logicaldrive-attr-value-pair</i>	<i>driveType=drive-type</i> ⁵ <i>raidLevel=raid-level</i> <i>arrayWidth=integer-literal</i> <i>arrayCount=integer-literal</i> <i>logicalDrivesPerArrayCount=integer-literal</i> ⁶ <i>hotSpareCount=integer-literal</i> <i>segmentSize=segment-size-spec</i> <i>cacheReadPrefetch=(TRUE FALSE)</i>
<i>create-volume-copy-attr-value-list</i>	<i>create-volume-copy-attr-value-pair</i> { <i>create-volume-copy-attr-value-pair</i> }
<i>create-volume-copy-attr-value-pair</i>	<i>copyPriority=(highest high medium low lowest)</i> <i>targetReadOnlyEnabled=boolean</i>
<i>recover-raid-logicalDrive-attr-value-list</i>	<i>recover-raid-logicalDrive-attr-value-pair</i> { <i>recover-raid-logicalDrive-attr-value-pair</i> }
<i>recover-raid-logicalDrive-attr-value-pair</i>	<i>owner=(a b)</i> <i>cacheReadPrefetch=(TRUE FALSE)</i>
<i>cache-flush-modifier-setting</i>	<i>immediate, 0, .25, .5, .75, 1, 1.5, 2, 5, 10, 20, 60, 120, 300, 1200, 3600, infinite</i>

Table 7. Recurring Syntax Elements (continued)

Recurring Syntax	Syntax Value
	<p>¹For enclosure loss protection to work, each disk drive in a array must be in a separate enclosure. If you set the enclosureLossProtect parameter to TRUE and you have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set enclosureLossProtect parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection.</p> <p>²To determine if a free capacity area exists, run the show array command.</p> <p>³The default disk drive (drive type) for DS3000 is SAS. The default disk drive (drive type) for DS4000 is fibre (Fibre Channel).</p> <p>⁴If you set the enclosureLossProtect parameter to TRUE, the storage subsystem returns an error if the controller firmware cannot find disk drives that will enable the new array to have enclosure loss protection. If you set the enclosureLossProtect parameter to FALSE, the storage subsystem performs the operation even if it means that the array might not have enclosure loss protection.</p> <p>⁵The driveType parameter is not required if only one type of disk drive is in the storage subsystem. If you use the driveType parameter, you must also use the hotSpareCount parameter and the arrayWidth parameter. If you do not use the driveType parameter, the DS4000 configuration defaults to Fibre Channel and the DS3000 configuration defaults to SAS.</p> <p>⁶The logicalDrivesPerArrayCount parameter is the number of equal-capacity logical drives per array.</p>

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. (For information about how to add comments, see ““Adding Comments to a Script File.””)

Note: While the CLI commands and the script commands are not case sensitive, user labels (such as for logical drives, 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 storageSubsystem 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 storageSubsystem 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 storageSubsystem resetConfiguration=true;
```

Chapter 3. Script Commands

Note: 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.

IBM recommends using the Storage Manager client GUI to manage your storage subsystems. – The command-line interface does not have any mechanisms to prevent you from inadvertently making unwanted changes to the storage subsystem. Therefore, IBM recommends using the Storage Manager client GUI to manage your storage subsystem configurations.

This chapter has five sections to help you use script commands:

- “Naming Conventions” lists the general formatting rules for entering the names of storage subsystem entities, such as logical drives or disk drives, with the script commands.
- “Firmware Compatibility Levels” on page 20 describes how to interpret the firmware level information.
- “Formatting Rules for Script Commands” on page 20 lists the general formatting rules that apply to the script command syntax.
- “DS3000, DS4000, and DS5000 support and minimum firmware levels” on page 21 indicates which commands you can use for DS3000, DS4000, and DS5000, or all three, and their minimum controller firmware requirements.
- “Script Commands Listed by Function” on page 129 lists the script commands organized into groups related to the physical features, the logical features, and the operational features of the storage subsystem.
- Finally, the script commands are listed alphabetically with detailed information including the command name, syntax, and parameters.

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 subsystems
 - Host groups
 - Hosts
 - Arrays
 - Logical drives
 - 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 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 subsystem, 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

Some of the script commands and the command parameters are not supported with every version of the controller firmware. Minimum controller firmware levels are listed in “DS3000, DS4000, and DS5000 support and minimum firmware levels” on page 21.

Note: For many commands, you can enable additional parameters by upgrading to a higher firmware level than the minimum.

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 enclosure 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 disk drive locations by using enclosure 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 disk 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 each time you enter a CLI command.

DS3000, DS4000, and DS5000 support and minimum firmware levels

The DS3000, DS4000, and DS5000 products share many common Storage Manager script commands. Some commands are specific to one product, two of the three products, all three of the products or require a particular minimum level of controller firmware.

The following table provides an alphabetical listing of commands and indicates which commands you can use for each, and their minimum controller firmware requirements.

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically

Storage Manager command	DS3000	DS4000	DS5000	Comments	
	Minimum firmware	Minimum firmware	Minimum firmware		
Activate Enhanced Remote Mirroring Feature	Not supported	6.1x.xx.xx	7.3x.xx.xx		
Activate Storage Subsystem Firmware	6.17.xx.xx				
Autoconfigure Storage Subsystem					
Autoconfigure Storage Subsystem Hot Spares					
Check Remote Mirror Status	Not supported				
Check Logical Drive Parity	6.17.xx.xx				
Clear Drive Channel Statistics					6.14.xx.xx adds discreteLines parameter on DS4800
Clear Storage Subsystem Configuration					7.1x.xx.xx and higher adds the all and arrays parameters
Clear Storage Subsystem Event Log					
Clear Storage Subsystem Firmware Pending Area					
Clear Logical Drive Reservations					5.4x.xx.xx
Clear Logical Drive Unreadable Sectors					
Create Array					6.1x.xx.xx
Create Host					5.2x.xx.xx
Create Host Group					
Create Host Port					
Create iSCSI Initiator		6.50.xx.xx	Not supported	Not supported	
Create RAID Logical Drive (Automatic Drive Select)		6.17.xx.xx	5.2x.xx.xx	7.3x.xx.xx	
Create RAID Logical Drive (Free Capacity Base Select)					
Create RAID Logical Drive (Manual Drive Select)					
Create Remote Mirror	Not supported	6.1x.xx.xx	7.3x.xx.xx		

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments
	Minimum firmware	Minimum firmware	Minimum firmware	
Create FlashCopy® Logical Drive	6.17.xx.xx	5.2x.xx.xx	7.3x.xx.xx	
Create Volume Copy		5.4x.xx.xx		
Deactivate Remote Mirror	Not supported	6.1x.xx.xx		
Delete Host	6.17.xx.xx	5.2x.xx.xx		
Delete Host Group				
Delete Host Port				
Delete iSCSI Initiator	6.50.xx.xx	Not supported	Not supported	

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments	
	Minimum firmware	Minimum firmware	Minimum firmware		
Delete Logical Drive	6.17.xx.xx	6.1x.xx.xx	7.3x.xx.xx	7.1x.xx.xx adds the removeArray parameter	
Delete Array					
Diagnose Controller		5.2x.xx.xx		6.1x.xx.xx adds parameters: <ul style="list-style-type: none"> • readTest • writeTest • dataLoopBackTest 6.14.xx.xx adds the discreteLinesDiagnosticTest parameter	
Diagnose Remote Mirror	Not supported	6.1x.xx.xx			
Disable Storage Subsystem Feature	6.17.xx.xx	5.2x.xx.xx		6.1x.xx.xx adds mixedDriveTypes parameter	
Download Drive Firmware		6.1x.xx.xx			
Download Environmental Card Firmware		5.2x.xx.xx			
Download Storage Subsystem Drive Firmware					
Download Storage Subsystem Firmware/NVSRAM		5.2x.xx.xx			
Enable Controller					
Enable Storage Subsystem Feature Key		6.1x.xx.xx			6.5x.xx.xx adds the featurePack parameter
Recopy Volume Copy					
Recover RAID Logical Drive				5.43.xx.xx	
Re-create Enhanced Remote Mirroring Repository Logical Drive		Not supported		6.1x.xx.xx	
Re-create FlashCopy	6.17.xx.xx	5.2x.xx.xx			
Re-create FlashCopy Collection		7.1x.xx.xx			
Remove Remote Mirror	Not supported	6.1x.xx.xx			
Remove Volume Copy	6.17.xx.xx	5.4x.xx.xx	7.3x.xx.xx		

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments	
	Minimum firmware	Minimum firmware	Minimum firmware		
Remove Logical Drive LUN Mapping	6.17.xx.xx	6.1x.xx.xx		7.3x.xx.xx and higher adds the ability to reset the battery installation dates on specific batteries that are inside the enclosure	
Repair Logical Drive Parity					
Replace Drive		7.1x.xx.xx			
Reset Controller		5.2x.xx.xx			
Reset Storage Subsystem Battery Install Date		6.1x.xx.xx			
Reset Storage Subsystem Diagnostic Data		6.16.xx.xx			
Reset Storage Subsystem iSCSI Baseline	6.50.xx.xx	Not supported	Not supported		
Reset Storage Subsystem RLS Baseline	6.17.xx.xx	5.4x.xx.xx	7.3x.xx.xx		
Reset Storage Subsystem SOC Baseline	Not supported	6.16.xx.xx			
Reset Storage Subsystem Logical Drive Distribution	6.17.xx.xx	5.2x.xx.xx			
Resume Remote Mirror	Not supported	6.1x.xx.xx			
Revive Array	6.17.xx.xx				
Revive Drive		5.43.xx.xx			
Save Controller NVSRAM		6.1x.xx.xx			
Save Drive Channel Fault Isolation Diagnostic Status	Not supported				
Save Drive Log	6.17.xx.xx	6.1x.xx.xx		7.3x.xx.xx	

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments
	Minimum firmware	Minimum firmware	Minimum firmware	
Save Storage Subsystem Configuration	6.17.xx.xx	6.1x.xx.xx		
Save Storage Subsystem Diagnostic Data		6.16.xx.xx		
Save Storage Subsystem Events		6.1x.xx.xx		
Save Storage Subsystem iSCSI Statistics	6.50.xx.xx	Not supported	Not supported	
Save Storage Subsystem Performance Statistics	6.17.xx.xx	6.1x.xx.xx	7.3x.xx.xx	
Save Storage Subsystem RLS Counts				
Save Storage Subsystem SOC Counts	Not supported	6.16.xx.xx		
Save Storage Subsystem State Capture	6.17.xx.xx	6.1x.xx.xx		
Save Storage Subsystem Support Data				
Set Array	6.17.xx.xx	6.10.xx.xx		Not supported

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments	
	Minimum firmware	Minimum firmware	Minimum firmware		
Set Array Forced State	6.17.xx.xx	7.1x.xx.xx	7.3x.xx.xx	7.30.xx.xx removed the bootp parameter and added the new Ethernet port options and the new iSCSI host-port options	
Set Controller		6.14.xx.xx			
Set Controller Service Action Allowed Indicator		6.16.xx.xx			
Set Drive State		5.2x.xx.xx			
Set Host		6.1x.xx.xx			
Set Host Channel					
Set Host Group					
Set Host Port					
Set iSCSI Initiator		6.50.xx.xx			Not supported
Set iSCSI Target Properties					
Set Remote Mirror	Not supported	6.1x.xx.xx			
Set Session	6.17.xx.xx	5.2x.xx.xx			
Set FlashCopy Logical Drive		6.1x.xx.xx			
Set Storage Subsystem Learn Cycle	Not supported				
Set Storage Subsystem	6.17.xx.xx	5.2x.xx.xx	<ul style="list-style-type: none"> 5.4x.xx.xx adds the failoverAlertDelay parameter 6.1x.xx.xx adds the alarm parameter 		
Set Storage Subsystem ICMP Response	6.50.xx.xx (DS3300 only)	7.1x.xx.xx			
Set Storage Subsystem iSNS Server IPv4 Address	6.50.xx.xx				
Set Storage Subsystem iSNS Server Listening Port					
Set Storage Subsystem iSNS Server Refresh					
Set Storage Subsystem Redundancy Mode	6.17.xx.xx	6.1x.xx.xx			
Set Storage Subsystem Time		5.4x.xx.xx			
Set Storage Subsystem Enclosure Positions		6.1x.xx.xx			
Set Enclosure Alarm	6.17.xx.xx	6.16.xx.xx	7.3x.xx.xx	For 6.14.xx.xx and 6.16.xx.xx, controller is not a valid value	

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments
	Minimum firmware	Minimum firmware	Minimum firmware	
Set Enclosure Identification	6.17.xx.xx	6.1x.xx.xx	7.3x.xx.xx	<ul style="list-style-type: none"> 6.14.xx.xx adds support for DS4800 6.16.xx.xx adds support for subsystems and enclosures that set enclosure IDs through the controller firmware
Set Enclosure Service Action Allowed Indicator				<ul style="list-style-type: none"> 6.14.xx.xx adds the powerfan and interconnect parameters 6.16.xx.xx adds the enclosure and esm parameters
Set Unnamed Discovery Session	6.50.xx.xx	7.1x.xx.xx		
Set Logical Drive	6.17.xx.xx	5.2x.xx.xx		7.1.xx.xx adds the preReadRedundancyCheck parameter
Set VolumeCopy		5.4x.xx.xx		
Show Array		6.1x.xx.xx		
Show Array Export Dependencies	Not supported	7.1x.xx.xx		
Show Array Import Dependencies				
Show Controller	6.17.xx.xx	5.4x.xx.xx		5.43.xx.xx adds the summary parameter
Show Controller NVSRAM		6.1x.xx.xx		
Show Current iSCSI Sessions	6.50.xx.xx	Not supported	Not supported	

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments	
	Minimum firmware	Minimum firmware	Minimum firmware		
Show Drive	6.17.xx.xx	5.43.xx.xx	7.3x.xx.xx	5.43.xx.xx adds the summary parameter. 6.14.xx.xx adds the connections parameter	
Show Drive Channel Stats		6.1x.xx.xx			
Show Drive Download Progress					
Show Host Ports	Not supported	5.4x.xx.xx			
Show Enhanced Remote Mirroring Logical Drive Candidates					
Show Enhanced Remote Mirroring Logical Drive Synchronization Progress					
Show Storage Subsystem	6.17.xx.xx				
Show Storage Subsystem Auto Configure		6.1x.xx.xx			
Show Storage Subsystem Host Topology		5.2x.xx.xx			
Show Storage Subsystem LUN Mappings		6.1x.xx.xx			
Show Storage Subsystem Negotiation Defaults		7.1x.xx.xx			
Show Storage Subsystem Unreadable Sectors		6.1x.xx.xx			
Show String					
Show Unconfigured iSCSI Initiators		6.50.xx.xx	Not supported	Not supported	

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments			
	Minimum firmware	Minimum firmware	Minimum firmware				
Show Logical Drive	6.17.xx.xx	5.4x.xx.xx	7.30.xx.xx	5.43.xx.xx adds the summary parameter			
Show Logical Drive Action Progress		5.43.xx.xx					
Show VolumeCopy		6.1x.xx.xx					
Show VolumeCopy Source Candidates							
Show VolumeCopy Target Candidates							
Show Logical Drive Performance Statistics							
Show Logical Drive Reservations					5.4x.xx.xx		
Start Drive Channel Fault Isolation Diagnostics	Not supported	Not supported	7.30.xx.xx				
Start Drive Channel Locate	6.17.xx.xx	6.1x.xx.xx	7.30.xx.xx				
Start Drive Initialize							
Start Drive Locate							
Start Drive Reconstruction		5.43.xx.xx					
Start iSCSI DHCP Refresh	6.50.xx.xx	Not supported	Not supported				
Start Enhanced Remote Mirroring Synchronization	Not supported	6.1x.xx.xx	7.3x.xx.xx				
Start Storage Subsystem Locate	6.17.xx.xx						
Start Enclosure Locate							
Start Array Defragment							
Start Array Export	Not supported				7.1x.xx.xx		
Start Array Import							
Start Array Locate		6.16.xx.xx					
Start Logical Drive Initialization	6.17.xx.xx	6.1x.xx.xx					
Stop Array Locate		6.16.xx.xx					
Stop Drive Channel Fault Isolation Diagnostics	Not supported	Not supported					
Stop Drive Channel Locate	6.17.xx.xx	6.1x.xx.xx					
Stop Drive Locate							
Stop iSCSI Session	6.50.xx.xx	Not supported	Not supported				

Table 8. DS3000, DS4000, and DS5000 commands, listed alphabetically (continued)

Storage Manager command	DS3000	DS4000	DS5000	Comments
	Minimum firmware	Minimum firmware	Minimum firmware	
Stop FlashCopy	6.17.xx.xx	6.1x.xx.xx	7.3x.xx.xx	
Stop Storage Subsystem Drive Firmware Download				
Stop Storage Subsystem Locate				
Stop Enclosure Locate				
Stop VolumeCopy		5.4x.xx.xx		
Suspend Remote Mirror	Not supported	6.1x.xx.xx		

Activate Enhanced Remote Mirroring Feature

This command creates the mirror repository logical drive and activates the Enhanced Remote Mirroring feature. When you use this command, you can define the mirror repository logical drive in one of three ways:

- User-defined disk drives
- User-defined array
- User-defined number of disk drives

If you choose to define a number of disk drives, the controller firmware chooses which disk drives to use for the mirror repository logical drive.

Syntax (User-Defined Disk Drives)

```
activate storageSubsystem feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
[enclosureLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Storage Subsystem)

```
activate storageSubsystem feature=remoteMirror
repositoryArray=arrayNumber[freeCapacityArea=freeCapacityIndexNumber]
```

Syntax (User-Defined Number of Disk Drives)

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

Parameters

Parameter	Description
repositoryRAIDLevel	The RAID level for the mirror repository logical drive. Valid values are 1, 3, 5, or 6.

Parameter	Description
repositoryDrives	The disk drives for the mirror repository logical drive. Specify the enclosure ID value and the slot ID value for each disk drive that you assign to the mirror repository logical drive. enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
repositoryarray	The name of the mirror repository logical drive group where the mirror repository logical drive is located. (To determine the names of the arrays in your storage subsystem, run the show storageSubsystem profile command.)
freeCapacityArea	The index number of the free space in an existing array that you want to use to create the mirror repository logical drive. Free capacity is defined as the free capacity between existing logical drives in a array. For example, a array might have the following areas: logical drive 1, free capacity, logical drive 2, free capacity, logical drive 3, free capacity. To use the free capacity following logical drive 2, you would specify: freeCapacityArea=2 Run the show array command to determine if a free capacity area exists.
repositoryDriveCount	The number of unassigned disk drives that you want to use for the mirror repository logical drive.
driveType	The type of disk drives that you want to use for the mirror repository logical drive. Valid disk drive types are fibre, SATA, or SAS. Use this parameter only when you use the repositoryDriveCount parameter. You must use this parameter only when you have more than one type of disk drive in your storage subsystem.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the mirror repository logical drive. To enforce enclosure loss protection, set this parameter to TRUE. The default value is FALSE.

Notes

If the disk 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 Enhanced Remote Mirroring is not activated. The error returns the amount of space that is needed for the mirror repository logical drive. 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 logical drives, the controller firmware returns an error message that provides the amount of space that is needed for the mirror repository logical drives. The command does not try to activate Enhanced Remote 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 disk drives, if you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection.

When the controller firmware assigns the disk drives, if you set the **enclosureLossProtect** parameter to TRUE, the storage subsystem returns an error if the controller firmware cannot provide disk drives that result in the new array having enclosure loss protection. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs the operation even if it means that the array might not have enclosure loss protection.

Activate Storage Subsystem Firmware

This command activates firmware that you have previously downloaded to the pending configuration area on the controllers in the storage subsystem.

Syntax

```
activate storageSubsystem firmware
```

Parameters

None.

Autoconfigure Storage Subsystem

This command automatically configures a storage subsystem. Before you enter the autoConfigure storageSubsystem command, run the show storageSubsystem autoConfiguration command. The show storageSubsystem autoConfiguration command returns configuration information in the form of a list of valid disk drive types, RAID levels, logical drive information, and hot spare information. (This list corresponds to the parameters for the autoConfigure storageSubsystem command.) The controllers audit the storage subsystem and then determine the highest RAID level that the storage subsystem can support and the most efficient logical drive definition for the RAID level. If the configuration that is described by the returned list is acceptable, you can enter the autoConfigure storageSubsystem 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 storageSubsystem command, the controllers set up the storage subsystem by using either the default parameters or those you selected.

Syntax

```
autoConfigure storageSubsystem  
[driveType=(fibre | SATA | SAS | )  
raidLevel=(0 | 1 | 3 | 5 | 6)  
arrayWidth=numberOfDrives  
arrayCount=numberOfArrays  
logicalDrivesPerArrayCount=numberOfLogicalDrivesPerArray  
hotSpareCount=numberOfHotSpares  
segmentSize=segmentSizeValue  
cacheReadPrefetch=(TRUE | FALSE)]
```

Parameters

Parameter	Description
driveType	The type of disk drives that you want to use for the storage subsystem. Valid disk drive types are fibre, SATA, or SAS. The driveType parameter is not required if only one type of disk drive is in the storage subsystem.
raidLevel	The RAID level of the array that contains the disk drives in the storage subsystem. Valid RAID levels are 0, 1, 3, 5, or 6.
arrayWidth	The number of disk drives in an array in the storage subsystem.

Parameter	Description
arrayCount	The number of arrays in the storage subsystem. Use integer values.
logicalDrivesPerArrayCount	The number of equal-capacity logical drives per array. Use integer values.
hotSpareCount	The number of hot spares that you want in the storage subsystem. Use integer values.
segmentSize	The amount of data (in KB) that the controller writes on a single disk drive in a logical drive before writing data on the next disk 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

Disk Drives

A array is a set of disk drives that are logically grouped together by the controllers in the storage subsystem. The number of disk drives in a array is a limitation of the controller firmware.

Disk drives do not report their exact capacity (for example, a 73-GB drive does not report 73 GB as its capacity).

Hot Spares

Hot spare disk drives can replace any failed disk drive in the storage subsystem. The hot spare must be the same type of disk drive as the disk drive that failed (that is, a SATA hot spare cannot replace a Fibre Channel disk drive). A hot spare must have capacity greater than or equal to any disk drive that can fail. If a hot spare is smaller than a failed disk drive, you cannot use the hot spare to rebuild the data from the failed disk 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk 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 disk 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 subsystem 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.

Autoconfigure Storage Subsystem Hot Spares

This command automatically defines and configures the hot spares in a storage subsystem. You can run this command at any time. This command provides the best hot spare coverage for a storage subsystem.

Syntax

```
autoConfigure storageSubsystem hotSpares
```

Parameters

None.

Notes

When you run the `autoconfigure storageSubsystem hotSpares` command, the controller firmware determines the number of hot spares to create based on the total number and type of disk drives in the storage subsystem. For Fibre Channel disk drives, SATA disk drives, and SAS disk drives, the controller firmware creates one hot spare for the storage subsystem and one additional hot spare for every 60 disk drives in the storage subsystem.

Check Logical Drive Parity

This command checks a logical drive for parity and media errors and writes the results of the check to a file.

Syntax

```
check logicalDrive [logicalDriveName]  
parity [parityErrorFile=filename]  
[mediaErrorFile=filename]  
[priority=(highest | high | medium | low | lowest)]  
[startingLBA=LBValue] [endingLBA=LBValue]  
[verbose=(TRUE | FALSE)]
```

Parameters

Parameter	Description
<code>logicalDrive</code>	The name of the specific logical drive for which you want to check parity. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").
<code>parityErrorFile</code>	The name of the file in which you want to save the parity error information. Enclose the file name in double quotation marks (" ").
<code>mediaErrorFile</code>	The name of the file in which you want to save the media error information. Enclose the file name in double quotation marks (" ").

Parameter	Description
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 logical drive 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-logical drive LUNs. Running a logical drive parity check on a very large single logical drive 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 logical drive parity check takes to complete.

Check Remote Mirror Status

This command returns the status of a remote-mirror logical drive. Use this command to determine when the status of the remote-mirror logical drive becomes Optimal.

Syntax

```
check remoteMirror localLogicalDrive [logicalDriveName] optimalStatus timeout=timeoutValue
```

Parameters

Parameter	Description
localLogicalDrive	The name of any remote-mirror logical drive. The remote-mirror logical drive can be the primary logical drive or the secondary logical drive of a remote-mirror pair. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").
timeout	The time interval within which the software can return the remote-mirror logical drive 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 Enhanced Remote Mirroring utility. (For more information, see “Asynchronous Enhanced Remote Mirroring Utility” on page 182.)

Clear Drive Channel Statistics

This command resets the statistics for all of the disk drive channels.

Syntax

```
clear allDriveChannels stats
```

Parameters

None.

Clear Logical Drive Reservations

This command clears persistent logical drive reservations.

Syntax

```
clear (allLogicalDrives | logicaldrive [logicalDriveName] |  
logicalDrives [logicalDriveName1 ... logicalDriveNameN]) reservations
```

Parameters

Parameter	Description
allLogicalDrives	The setting to clear persistent logical drive reservations on all of the logical drives in the storage subsystem.
logicalDrive or logicalDrives	The name of the specific logical drive for which you want to clear persistent logical drive reservations. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive 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.

Clear Logical Drive Unreadable Sectors

This command clears unreadable sector information from one or more logical drives.

Syntax

```
clear (allLogicalDrives | logicaldrive [logicalDriveName] |  
logicalDrives [logicalDriveName1 ... logicalDriveNameN]) unreadableSectors
```

Parameters

Parameter	Description
allLogicalDrives	The setting to clear unreadable sector information from all of the logical drives in the storage subsystem.
logicalDrive or logicalDrives	The name of the specific logical drive for which you want to clear unreadable sector information. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive 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.

Clear Storage Subsystem Configuration

Use this command to perform one of these operations:

- Clear the entire storage subsystem configuration, and return it back to the initial installation state
- Clear the configuration except for security information and identification information
- Clear array configuration information and logical drive configuration information only

Attention: Possible damage to the storage subsystem configuration – When you run this command, the existing storage subsystem configuration is deleted.

Syntax

```
clear storageSubsystem configuration [all | arrays]
```

Parameters

Parameter	Description
None	If you do not enter a parameter, this command removes all configuration information for the storage subsystem, except for information related to security and identification.
all	The setting to remove the entire configuration of the storage subsystem, including security information and identification information. Removing all configuration information returns the storage subsystem to its initial state.
arrays	The setting to remove the logical drive configuration and the array configuration. The rest of the configuration stays intact.

Notes

When you run this command, the storage subsystem becomes unresponsive, and all script processing is canceled. You must remove and re-add the storage subsystem to resume communication with the host. To remove an unresponsive storage subsystem, access the Enterprise Management Window, and select **Edit → Remove**. To re-add the storage subsystem, access the Enterprise Management Window, select **Edit → Add Storage Array**, and enter the appropriate IP addresses.

Clear Storage Subsystem Event Log

This command clears the Event Log in the storage subsystem by deleting the data in the Event Log buffer.

Attention:

Possible damage to the storage subsystem configuration – As soon as you run this command, the existing Event Log in the storage subsystem is deleted.

Syntax

```
clear storageSubsystem eventLog
```

Parameters

None.

Clear Storage Subsystem 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 subsystem configuration – As soon as you run this command, the contents of the existing pending area in the storage subsystem are deleted.

Syntax

```
clear storageSubsystem firmwarePendingArea
```

Parameters

None.

Create Array

This command creates either a free-capacity array or one logical drive on a set of unassigned disk drives.

Syntax

```
create array [userLabel]  
drives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)  
raidLevel=(0 | 1 | 3 | 5 | 6)  
[capacity=arrayCapacity  
owner=(a | b)  
cacheReadPrefetch=(TRUE | FALSE)  
segmentSize=segmentSizeValue  
usageHint=(fileSystem | dataBase | multiMedia)]  
[enclosureLossProtect=(TRUE | FALSE)]
```

Parameters

Parameter	Description
array	The alphanumeric identifier (including - and _) that you want to give the new array. Enclose the array identifier in square brackets ([]).
drives	The disk drives that you want to assign to the array that you want to create. Specify the enclosure ID and slot ID for each disk drive that you assign to the array. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
raidLevel	The RAID level of the array that contains the logical drive. Valid values are 0, 1, 3, 5, or 6.
capacity	The size of the logical drive that you are adding to the storage subsystem. Size is defined in units of bytes, KB, MB, GB, or TB. These examples show the syntax: capacity=500 bytes capacity=2 GB
owner	The setting for which controller owns the array. 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.

Parameter	Description
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 disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.
usageHint	The setting for both the cacheReadPrefetch parameter and the segmentSize parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the logical drive. Valid values are fileSystem, dataBase, or multiMedia.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the array. To enforce enclosure loss protection, set this parameter to TRUE. The default value is FALSE.

Notes

The **drives** parameter lets you choose the number of disk drives that you want to use in the array. If you choose this option, you do not need to specify the disk drives by enclosure ID and slot ID. The controllers choose the specific disk drives to use for the array.

If you do not specify a capacity by using the **capacity** parameter, all of the disk drive capacity that is available in the array 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 disk 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 subsystem 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk drives that are required to satisfy a data transfer request.

enclosure Loss Protection

For enclosure loss protection to work, each disk drive in a array must be in a separate enclosure. If you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection. enclosure loss protection is not valid when you create logical drives on existing arrays.

Create FlashCopy Logical Drive

This command creates a FlashCopy of a base logical drive. This command defines three ways to create a FlashCopy logical drive:

- User-defined disk drives
- User-defined array
- User-defined number of disk drives

If you choose to define a number of disk drives, the controller firmware chooses which disk drives to use for the FlashCopy logical drive.

Syntax (User-Defined Disk Drives)

```
create FlashCopyLogicalDrive baseLogicalDrive="baselogical driveName"
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
userLabel="FlashCopyLogicalDriveName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failFlashCopy)
enclosureLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Array)

```
create FlashCopyLogical drive baseLogical drive="baselogical driveName"
[repositoryArray=arrayNumber
freeCapacityArea=freeCapacityIndexNumber
userLabel="FlashCopyLogical driveName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failFlashCopy)
enclosureLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Number of Disk Drives)

```
create FlashCopyLogical drive baseLogical drive="baselogical driveNamerepository"
RAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
driveType=(fibre | SATA | SAS)
userLabel="FlashCopyLogical driveName"
warningThresholdPercent=percentValue
```

```

repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failFlashCopy)
enclosureLossProtect=(TRUE | FALSE)

```

A FlashCopy logical drive is a point-in-time image of a logical drive in a storage subsystem, which is created by using the create FlashCopy logical drive command and specific parameters as described in the next section.

Parameters

Parameter	Description
baseLogicalDrive	The name of the base logical drive from which you want to take a FlashCopy. Enclose the base logical drive name in double quotation marks (" ").
repositoryRAIDLevel	The RAID level for the FlashCopy repository logical drive. Valid values are 1, 3, 5, or 6.
repositoryDrives	The disk drives that you want to assign to the FlashCopy repository logical drive. Specify the enclosure ID and the slot ID for each disk drive that you assign to the FlashCopy repository logical drive. enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the list of disk drives in parentheses.
repositoryDriveCount	The number of unassigned disk drives that you want to use for the FlashCopy repository logical drive.
driveType	The type of disk drives that you want to use for the FlashCopy repository logical drive. Valid disk drive types are SCSI, fibre, or SAS.
repositoryArray	The alphanumeric identifier (including - and _) of the array where the FlashCopy repository logical drive is located.
freeCapacityArea	The index number of the free space in an existing array that you want to use to create the FlashCopy repository logical drive. Free capacity is defined as the free capacity between existing logical drives in an array. For example, an array might have the following areas: logical drive 1, free capacity, logical drive 2, free capacity, logical drive 3, free capacity. To use the free capacity following logical drive 2, you would specify: freeCapacityArea=2 Run the show array command to determine if a free capacity area exists.
userLabel	The name that you want to give the FlashCopy logical drive. Enclose the FlashCopy logical drive name in double quotation marks (" ").
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the FlashCopy repository logical drive 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 FlashCopy repository logical drive as a percentage of the base logical drive. 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 FlashCopy repository logical drive. Enclose the FlashCopy repository logical drive name in double quotation marks (" ").
repositoryFullPolicy	How you want FlashCopy processing to continue if the FlashCopy repository logical drive is full. You can choose to fail writes to the base logical drive (failBaseWrites) or fail the FlashCopy logical drive (failFlashCopy). The default value is failFlashCopy.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the FlashCopy repository logical drive. To enforce enclosure loss protection, set this parameter to TRUE. The default value is FALSE.

Notes

The logical drive that you are taking a FlashCopy of must be a standard logical drive in the storage subsystem. The maximum number of FlashCopy logical drives that you can create is one-half of the total number of logical drives 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 FlashCopy logical drive and the FlashCopy repository logical drive is to add a hyphenated suffix to the original base logical drive name. The suffix distinguishes between the FlashCopy logical drive and the FlashCopy repository logical drive. For example, if you have a base logical drive with a name of Engineering Data, the FlashCopy logical drive can have a name of Engineering Data-S1, and the FlashCopy repository logical drive can have a name of EngineeringData-R1.

If you do not choose a name for either the FlashCopy logical drive or the FlashCopy repository logical drive, the storage management software creates a default name by using the base logical drive name. An example of the FlashCopy logical drive name that the controllers might create is, if the base logical drive name is aaa and does not have a FlashCopy logical drive, the default FlashCopy logical drive name is aaa-1. If the base logical drive already has $n-1$ number of FlashCopy logical drives, the default name is aaa- n . An example of the FlashCopy repository logical drive name that the controller might create is, if the base logical drive name is aaa and does not have a FlashCopy repository logical drive, the default FlashCopy repository logical drive name is aaa-R1. If the base logical drive already has $n-1$ number of FlashCopy repository logical drives, the default name is aaa-R n .

If you do not specify the unconfigured space or free space, the FlashCopy repository logical drive is placed in the same array as the base logical drive. If the array where the base logical drive resides does not have enough space, this command fails.

When you assign the disk drives, if you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection.

When the controller firmware assigns the disk drives, if you set the **enclosureLossProtect** parameter to TRUE, the storage subsystem returns an error if the controller firmware cannot provide disk drives that result in the new array having enclosure loss protection. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs the operation even if it means the array might not have enclosure loss protection.

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 create hostGroup command.) The defaultGroup option is the host group that contains the host to which the logical drive is mapped.
hostType	The index label or the index number that identifies the host type. Use the show storageSubsystem hostTypeTable 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 subsystem and accesses the logical drives on the storage subsystem through its HBA host ports. You can define specific logical drive-to-LUN mappings to an individual host. You also can assign the host to a host group that shares access to one or more logical drives.

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 logical drives. 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 logical drives.

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.

Create Host Group

This command creates a new host group.

Syntax

```
create hostGroup userLabel="hostGroupName"
```

Parameter

Parameter	Description
<code>userLabel</code>	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 logical drives. 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 logical drives.

You can use any combination of alphanumeric characters, hyphens, and underscores for the names. Names can have a maximum of 30 characters.

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
<code>identifier</code>	The unique ID of the HBA host port. Enclose the unique ID in double quotation marks (" ").
<code>userLabel</code>	The name that you want to give to the new HBA host port. Enclose the HBA host port label in double quotation marks (" ").
<code>host</code>	The name of the host for which you are defining an HBA host port. Enclose the host name in double quotation marks (" ").
<code>hostType</code>	<i>This parameter is deprecated.</i> The index label or the index number that identifies the host type. Use the <code>show storageSubsystem 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 logical drives in a storage subsystem. 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.

Create iSCSI Initiator

This command creates a new iSCSI initiator object.

Syntax

```
create iscsiInitiator iscsiName=iSCSI-ID
userLabel="name"
host=host-name
[chapSecret=securityKey]
```

Parameters

Parameters	Description
iscsiName	The default identifier of the iSCSI initiator.
userLabel	The name that you want to use for the iSCSI initiator. Enclose the name in double quotation marks (" ").
host	The name of the host in which the iSCSI initiator is installed.
chapSecret	The security key that you want to use to authenticate a peer connection.

Notes

Challenge Handshake Authentication Protocol (CHAP) is a protocol that authenticates the peer of a connection. CHAP is based upon the peers sharing a "secret." A secret is a security key that is similar to a password.

Use the **chapSecret** parameter to set up the security keys for initiators that require a mutual authentication.

Create RAID Logical Drive (Automatic Drive Select)

This command creates a array across the disk drives in the storage subsystem, and a new logical drive in the array. The storage subsystem controllers choose the disk drives to be included in the logical drive.

Important:

If you have disk drives with different capacities, you cannot automatically create logical drives by specifying the **driveCount** parameter. If you want to create logical drives with disk drives of different capacities, see "Create RAID Logical Drive (Manual Drive Select)" on page 50.

Syntax

```
create logicalDrive driveCount=numberOfDrives
raidLevel=(0 | 1 | 3 | 5 | 6)
userLabel="logicalDriveName"
[driveType=(fibre | SATA | SAS)
capacity=logicalDriveCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia)
enclosureLossProtect=(TRUE | FALSE)
dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
driveCount	The number of unassigned disk drives that you want to use in the array.
raidLevel	The RAID level of the array that contains the logical drive. Valid values are 0, 1, 3, 5, or 6.
userLabel	The name that you want to give to the new logical drive. Enclose the new logical drive name in double quotation marks ("").
driveType	The type of disk drive that you want to use in the logical drive. You cannot mix disk drive types. Valid disk drive types are fibre, SATA, or SAS.
capacity	The size of the logical drive that you are adding to the storage subsystem. Size is defined in units of bytes, KB, MB, GB, or TB.
owner	The controller that owns the logical drive. 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 disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.
usageHint	The setting for both cacheReadPrefetch parameter and the segmentSize parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the logical drive. Valid values are fileSystem, dataBase, or multiMedia.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the array. To enforce enclosure 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 disk drives that you want to use in the array. You do not need to specify the disk drives by enclosure ID and slot ID. The controllers choose the specific disk drives to use for the array.

The **owner** parameter defines which controller owns the logical drive.

If you do not specify a capacity using the **capacity** parameter, all of the disk drive capacity that is available in the array 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 disk 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 subsystem 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk 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.

Enclosure Loss Protection

For enclosure loss protection to work, each disk drive in a array must be on a separate enclosure. If you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection.

enclosure loss protection is not valid when you create logical drives on existing arrays.

Create RAID Logical Drive (Free Capacity Base Select)

This command creates a logical drive in the free space of an array.

Syntax

```
create logicalDrive array=arrayNumber
userLabel="logicalDriveName"
[freeCapacityArea=freeCapacityIndexNumber
capacity=logicalDriveCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia) |
dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
array	The alphanumeric identifier (including - and _) for a specific array in your storage subsystem.
userLabel	The name that you want to give the new logical drive. Enclose the new logical drive name in double quotation marks (" ").
freeCapacityArea	The index number of the free space in an existing array that you want to use to create the new logical drive. Free capacity is defined as the free capacity between existing logical drives in a array. For example, a array might have the following areas: logical drive 1, free capacity, logical drive 2, free capacity, logical drive 3, free capacity. To use the free capacity following logical drive 2, you would specify: freeCapacityArea=2 Run the show array command to determine if the free capacity area exists.
capacity	The size of the logical drive that you are adding to the storage subsystem. Size is defined in units of bytes, KB, MB, GB, or TB.
owner	The controller that owns the logical drive. 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 disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.
usageHint	The settings for both the cacheReadPrefetch parameter and the segmentSize parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the logical drive. 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 logical drive. The preferred controller ownership of a logical drive is the controller that currently owns the array.

If you do not specify a capacity using the **capacity** parameter, all of the available capacity in the free capacity area of the array 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk 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 disk 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 subsystem 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.

Create RAID Logical Drive (Manual Drive Select)

This command creates a new array and logical drive and lets you specify the disk drives for the logical drive.

Important: You cannot use mixed disk drive types in the same array and logical drive. This command fails if you specify different types of disk drives for the RAID logical drive.

Syntax

```
create logicalDrive drives=(enclosureID1,slotID1...enclosureIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6)
userLabel="logicalDriveName"
[capacity=logicalDriveCapacity
owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue
usageHint=(fileSystem | dataBase | multiMedia)
enclosureLossProtect=(TRUE | FALSE)
dssPreAllocate=(TRUE | FALSE)]
```

Parameters

Parameter	Description
drives	The disk drives that you want to assign to the logical drive that you want to create. Specify the enclosure ID and slot ID for each unassigned disk drive that you want to assign to the logical drive. enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
raidLevel	The RAID level of the array that contains the logical drive. Valid values are 0, 1, 3, 5, or 6.
userLabel	The name that you want to give the new logical drive. Enclose the new logical drive name in double quotation marks (" ").
capacity	The size of the logical drive that you are adding to the storage subsystem. Size is defined in units of bytes, KB, MB, GB, or TB.
owner	The controller that owns the logical drive. 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 disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.
usageHint	The settings for both the cacheReadPrefetch parameter and the segmentSize parameter to be default values. The default values are based on the typical I/O usage pattern of the application that is using the logical drive. Valid values are fileSystem, dataBase, or multiMedia.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the repository. To enforce enclosure 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

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 logical drive. The preferred controller ownership of a logical drive is the controller that currently owns the array.

If you do not specify a capacity using the **capacity** parameter, all of the disk drive capacity that is available in the array 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk 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 disk 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 subsystem 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 enclosure loss protection to work, each disk drive in a array must be on a separate enclosure. If you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the array that you create might not have enclosure loss protection. enclosure loss protection is not valid when you create logical drives on existing arrays.

Create Remote Mirror

This command creates both the primary logical drive and the secondary logical drive for a remote-mirror pair. This command also sets the write mode (Synchronous or Asynchronous) and the synchronization priority.

Syntax

```
create remoteMirror primary="primaryLogicalDriveName"
secondary="secondaryLogicalDriveName"
(remoteStorageSubsystemName="storageSubsystemName" |
remoteStorageSubsystemWwn="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 logical drive on the local storage subsystem that you want to use for the primary logical drive. Enclose the primary logical drive name in double quotation marks (" ").
secondary	The name of an existing logical drive on the remote storage subsystem that you want to use for the secondary logical drive. Enclose the secondary logical drive name in double quotation marks (" ").
remoteStorageSubsystemName	The name of the remote storage subsystem. Enclose the remote storage subsystem name in double quotation marks (" ").
remoteStorageSubsystemWwn	The WWID of the remote storage array. Enclose the WWID in double quotation marks (" ").
remotePassword	The password for the remote storage subsystem. Use this parameter when the remote storage subsystem 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.
autoResync	The settings for automatic resynchronization between the primary logical drives and the secondary logical drives 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 logical drive and the secondary logical drive.• disabled – Automatic resynchronization is turned off. To resynchronize the primary logical drives and the secondary logical drive, you must run the resume remoteMirror command.
writeOrder	The write order for data transmission between the primary logical drive and the secondary logical drive. Valid values are preserved or notPreserved.
writeMode	How the primary logical drive writes to the secondary logical drive. 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 logical drive and the secondary logical drive, the secondary logical drive must be of equal or greater size than the primary logical drive. The RAID level of the secondary logical drive does not have to be the same as the primary logical drive.

The DS4800 controller can define a maximum of 128 remote mirrors. The DS4200 and DS4700 controllers can define a maximum of 64 remote mirrors.

Passwords are stored on each storage subsystem 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 subsystem password by using the `set storageSubsystem` command.)

Synchronization priority defines the amount of system resources that are used to synchronize the data between the primary logical drive and the secondary logical drive 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 logical drive to the secondary logical drive in the same order as the host writes to the primary logical drive. 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 logical drive has the same data as the primary logical drive.

Create Volume Copy

This command creates a VolumeCopy and starts the VolumeCopy operation.

Important: You can have a maximum of eight VolumeCopies in progress at one time. If you try to create more than eight VolumeCopies at one time, the controllers return a status of Pending until one of the VolumeCopies 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 logical drive that you want to use as the source logical drive. Enclose the source logical drive name in double quotation marks (" ").
target	The name of an existing logical drive that you want to use as the target logical drive. Enclose the target logical drive name in double quotation marks (" ").
copyPriority	The priority that VolumeCopy 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 logical drive or only read from the target logical drive. To write to the target logical drive, set this parameter to FALSE. To prevent writing to the target logical drive, 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 logical drive and the target logical drive of a VolumeCopy pair. If you select the highest priority level, the VolumeCopy uses the most system resources to perform VolumeCopy, which decreases performance for host data transfers.

Deactivate Remote Mirror

This command deactivates the Enhanced Remote Mirroring feature, disassembles the mirror repository logical drive, and releases the controller-owner of the secondary logical drive. The controller host port that is dedicated to the secondary logical drive is available for host data transfers.

Syntax

```
deactivate storageSubsystem feature=remoteMirror
```

Parameters

None.

Delete Array

This command deletes an entire array and its associated logical drives.

Attention: Possible damage to the storage subsystem configuration – All of the data in the array is lost as soon as you run this command.

Syntax

```
delete array [arrayNumber]
```

Parameter

Parameter	Description
array	The alphanumeric identifier (including - and _) of the array that you want to delete. Enclose the array identifier in square brackets ([]).

Delete Host

This command deletes a host.

Syntax

```
delete host [hostName]
```

Delete Host Group

This command deletes a host group.

Attention:

Possible damage to the storage subsystem 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 logical drives. The host group is a logical entity.

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 logical drives in a storage subsystem. If the host bus adapter has only one physical connection (one host port), the terms HBA host port and host bus adapter are synonymous.

Delete iSCSI Initiator

This command deletes a specific iSCSI initiator object.

Syntax

```
delete iscsiInitiator ("iSCSI-ID" | "name")
```

Parameters

Parameter	Description
<i>iSCSI-ID</i>	The identifier of the iSCSI initiator that you want to delete. Enclose the name in double quotation marks (" ").
<i>name</i>	The name of the iSCSI initiator that you want to delete. Enclose the name in double quotation marks (" ").

Delete LogicalDrive

This command deletes one or more standard logical drives, flashcopy logical drives, or flashcopy repository logical drives.

Attention: Possible damage to the storage subsystem configuration – All of the data in the logical drive is lost as soon as you run this command.

Syntax

```
delete (allLogicalDrives | logicalDrive [logicalDriveName] |  
logicalDrives [logicalDriveName1 ... logicalDriveNameN])  
removeArray=(TRUE | FALSE)
```

Parameters

Parameter	Description
allLogicalDrives	This parameter deletes all of the logical drives in a storage subsystem.
logicalDrive or logicalDrives	The name of the logical drive that you want to delete. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").
removeArray	Deleting the last logical drive in a array does not delete the array. You can have a stand-alone array (minus any logical drives). To remove the stand-alone array, set this parameter to TRUE. To keep stand-alone arrays intact, set this parameter to FALSE.

Notes

When you use the **allLogicalDrives** parameter, this command deletes logical drives until all of the logical drives are removed or until an error is encountered. If an error is encountered, this command does not try to delete the remaining logical drives. Deleting logical drives from different arrays is possible. All of the arrays that become empty are deleted if you set the **removeArray** parameter to TRUE.

If you want to delete an entire array, you can also use the delete array command.

Diagnose Controller

This command runs diagnostic tests on the controller. The diagnostic tests consist of loopback tests in which data is written to the disk drives and read from the disk 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 disk 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 disk 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 module. 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. This test applies only to the DS4800 (Models 82, 84, 88)-series controller modules. 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.

Diagnose Remote Mirror

This command tests the connection between the specified primary logical drives and the mirror logical drives on a storage subsystem with the Enhanced Remote Mirroring feature enabled.

Syntax

```
diagnose remoteMirror (primary [primaryLogicalDriveName] |  
primaries [primaryLogicalDriveName1 ... primaryLogicalDriveNameN])  
testID=connectivity
```

Parameters

Parameter	Description
primary or primaries	The name of the primary logical drive of the remote mirror pair that you want to test. You can enter more than one primary logical drive name. Enclose the primary logical drive names in square brackets ([]). If the primary logical drive name has special characters, you also must enclose the primary logical drive name in double quotation marks (" ").

Disable Storage Subsystem Feature

This command disables a storage subsystem feature. Run the `show storageSubsystem` command to show a list of the feature identifiers for all enabled features in the storage subsystem.

Syntax

```
disable storageSubsystem [featurePack | feature=(storagePartition2  
| storagePartition4 | storagePartition8 | storagePartition16 | storagePartition64  
| storagePartitionMax | flashcopy | remoteMirror | volumeCopy | goldKey |  
mixedDriveTypes)]
```

Parameters

None.

Notes

If you specify the **remoteMirror** parameter, this command disables the Enhanced Remote Mirroring feature and takes away the structure of the mirror repository logical drive.

Download Drive Firmware

This command downloads a firmware image to a disk drive.

Attention: Possible damage to the storage subsystem configuration – Downloading disk drive firmware incorrectly can result in damage to the disk drives or a loss of data.

Syntax

```
download drive [enclosureID,slotID] firmware file="filename"
```

Parameters

Parameter	Description
drive	The disk drive to which you want to download the firmware image. Specify the enclosure ID and slot ID for the disk drive. enclosure 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 disk drive firmware, take these precautions:

- Stop all I/O activity to the storage subsystem before you download the firmware image.
- Make sure that the firmware image file is compatible with the disk drive. If you download a firmware image file that is not compatible with the disk drive that you have selected, the disk drive might become unusable.
- Do not make any configuration changes to the storage subsystem while you download disk drive firmware. Trying to make a configuration change can cause the firmware download to fail and make the selected disk drives unusable.

You can use this command to test the firmware on one disk drive before you install the firmware on all of the disk drives in a storage subsystem. (Use the `download storageSubsystem driveFirmware` command to download firmware on all of the disk drives in the storage subsystem.) 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

Download Environmental Card Firmware

This command downloads ESM firmware.

Syntax

```
download (allEnclosures | enclosure [enclosureID])  
firmware file="filename"
```

Parameters

Parameter	Description
enclosure	The expansion drawer that contains the ESM card to which you want to load new firmware. enclosure ID values are 0 to 99. Enclose the enclosure 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 **allEnclosures** parameter downloads new firmware to all of the enclosures in the storage subsystem.
- The **enclosure** parameter downloads new firmware to a specific expansion drawer. If you need to download new firmware to more than one expansion drawer, but not all expansion drawers, you must enter this command for each expansion drawer.

Download Storage Subsystem Firmware/NVSRAM

This command downloads firmware and, optionally, NVSRAM values for the storage subsystem controller. If you want to download only NVSRAM values, use the `download storageSubsystem NVSRAM` command.

Syntax

```
download storageSubsystem 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 firmware 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 (" ").
<i>NVSRAM-filename</i>	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 downgrade 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 activateNow parameter to FALSE, you must run the <code>activate storageSubsystem firmware</code> command to activate the firmware values and the NVSRAM values at a later time.

Download Storage Subsystem Drive Firmware

This command downloads firmware images to all of the disk drives in the storage subsystem.

Syntax

```
download storageSubsystem 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 disk drives in a storage subsystem. The number of firmware image files that you can download depends on the storage subsystem. The storage management software returns an error if you try to download more firmware image files than the storage subsystem can accept.

You can schedule downloads for multiple disk drives at the same time, including multiple disk drives in a redundant array. Each firmware image file contains information about the disk drive types on which the firmware image runs. The specified firmware images can be downloaded only to a compatible disk drive. Use the download drive firmware command to download a firmware image to a specific disk drive.

The download storageSubsystem driveFirmware command blocks all I/O activity until a download try has been made for each candidate disk drive or you run the stop storageSubsystem downloadDriveFirmware command. When the download storageSubsystem driveFirmware command finishes downloading the firmware image, each candidate disk drive shows the download status for each disk drive. One of these statuses is returned:

- Successful
- Unsuccessful With Reason
- Never Attempted With Reason

Download Storage Subsystem NVSRAM

This command downloads NVSRAM values for the storage subsystem controller.

Syntax

```
download storageSubsystem 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 (" ").

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.

Enable Storage Subsystem Feature Key

This command enables a feature by using a Feature Key file.

Syntax

```
enable storageSubsystem [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 .key 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 Storage Partitioning and Enhanced Remote Mirroring. These features are combined for the convenience of the users.

A feature is an additional application to enhance the capabilities of a storage subsystem.

Recopy Volume Copy

This command reinitiates a VolumeCopy operation by using an existing VolumeCopy 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 logical drive for which you want to reinitiate a VolumeCopy operation. Enclose the target logical drive name in square brackets ([]). If the target logical drive name has special characters, you also must enclose the target logical drive name in double quotation marks (" ").
source	The name of the source logical drive for which you want to reinitiate a VolumeCopy operation. Enclose the source logical drive name in square brackets ([]). If the source logical drive name has special characters, you also must enclose the source logical drive name in double quotation marks (" ").
copyPriority	The priority that the VolumeCopy 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 logical drive or only read from the target logical drive. To write to the target logical drive, set this parameter to FALSE. To prevent writing to the target logical drive, set this parameter to TRUE.

Notes

Copy priority defines the amount of system resources that are used to copy the data between the source logical drive and the target logical drive of a VolumeCopy pair. If you select the highest priority level, the VolumeCopy uses the most system resources to perform the volume copy, which decreases performance for host data transfers.

Recover RAID Logical Drive

This command creates a RAID logical drive with the given properties without initializing any of the user data areas on the disk drives. Parameter values are derived from the Recovery Profile data file for the storage subsystem.

Syntax

```
recover logicalDrive (drive=(enclosureID,slotID) |  
drives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn) |  
Array=ArrayNumber)  
userLabel="logicalDriveName"  
capacity=logicalDriveCapacity  
offset=offsetValue  
raidLevel=(0 | 1 | 3 | 5 | 6)  
segmentSize=segmentSizeValue  
[owner=(a | b)  
cacheReadPrefetch=(TRUE | FALSE)]
```

Parameters

Parameter	Description
drive or drives	The disk drives that you want to assign to the logical drive that you want to create. Specify the enclosure ID and slot ID for each disk drive that you assign to the logical drive. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
array	The name of the array in which you want to create the new logical drive. (To determine the names of the arrays in your storage subsystem, run the show storageSubsystem profile command.)
newArray	This parameter enables the user to specify a name for a new array that is automatically created by the controller.
userLabel	The name that you want to give the new logical drive. Enclose the new logical drive name in double quotation marks (" ").
capacity	The size of the logical drive that you are adding to the storage subsystem. Size is defined in units of bytes, KB, MB, GB, or TB.
offset	The number of blocks from the start of the array to the start of the referenced logical drive.
raidLevel	The RAID level of the array that contains the disk drives. Valid values are 0, 1, 3, 5, or 6.
segmentSize	The amount of data (in KB) that the controller writes on a single disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.

Parameter	Description
owner	The controller that owns the logical drive. 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 logical drive. The preferred controller ownership of a logical drive is the controller that currently owns the array.

Segment Size

The size of a segment determines how many data blocks that the controller writes on a single disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk 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 subsystem 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.

Re-create Enhanced Remote Mirroring Repository Logical Drive

This command creates a new Enhanced Remote Mirroring repository logical drive (also called a mirror repository logical drive) by using the parameters defined for a previous mirror repository logical drive. The underlying requirement is that you have previously created a mirror repository logical drive. When you use this command, you can define the mirror repository logical drive in one of three ways: user-defined disk drives, user-defined array, or user-defined number of disk drives for the mirror

repository logical drive. If you choose to define a number of disk drives, the controller firmware chooses which disk drives to use for the mirror repository logical drive.

Syntax (User-Defined Disk Drives)

```
recreate storageSubsystem mirrorRepository
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
[enclosureLossProtect=(TRUE | FALSE)]
```

Syntax (User-Defined Array)

```
recreate storageSubsystem mirrorRepository
repositoryarray=arrayNumber[freeCapacityArea=freeCapacityIndexNumber]
```

Syntax (User-Defined Number of Disk Drives)

```
recreate storageSubsystem mirrorRepository
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
[driveType=(fibre | SATA | SAS)]
[enclosureLossProtect=(TRUE | FALSE)]
```

Parameters

Parameter	Description
repositoryRAIDLevel	The RAID level for the mirror repository logical drive. Valid values are 1, 3, 5, or 6.
repositoryDrives	The disk drives for the mirror repository logical drive. Specify the enclosure ID and slot ID for each disk drive that you assign to the mirror repository logical drive. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
repositoryArray	The name of the array where the mirror repository logical drive is located.
freeCapacityArea	The index number of the free space in an existing array that you want to use to re-create the mirror repository logical drive. Free capacity is defined as the free capacity between existing logical drives in a array. For example, a array might have the following areas: logical drive 1, free capacity, logical drive 2, free capacity, logical drive 3, free capacity. To use the free capacity following logical drive 2, you would specify: freeCapacityArea=2 Run the show array command to determine if a free capacity area exists.
repositoryDriveCount	The number of unassigned disk drives that you want to use for the mirror repository logical drive.
driveType	The type of disk drives that you want to use for the mirror repository logical drive. Valid disk drive types are fibre, SATA, or SAS.
enclosureLossProtect	The setting to enforce enclosure loss protection when you create the mirror repository logical drive. To enforce enclosure 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 logical drive that is too small, the controller firmware returns an error message, which states the amount of space that is needed for the mirror repository logical drive. The command does not try to change the mirror repository logical drive. You can re-enter the command by using the value from the error message for the storage space value of the mirror repository logical drive.

When you assign the disk drives, if you set the **enclosureLossProtect** parameter to TRUE and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs operations, but the mirror repository logical drive that you create might not have enclosure loss protection.

When the controller firmware assigns the disk drives, if you set the **enclosureLossProtect** parameter to TRUE, the storage subsystem returns an error if the controller firmware cannot provide disk drives that result in the new mirror repository logical drive having enclosure loss protection. If you set the **enclosureLossProtect** parameter to FALSE, the storage subsystem performs the operation even if it means that the mirror repository logical drive might not have enclosure loss protection.

Re-create FlashCopy

This command starts a fresh copy-on-write operation by using an existing FlashCopy logical drive.

Syntax

```
recreate FlashCopy (logical drive [logicalDriveName]  
| logical drives [logicalDriveName1 ... logicalDriveNameN]) [userLabel="FlashCopyLogicalDriveName"  
warningThresholdPercent=percentValue repositoryFullPolicy  
(failBaseWrites | failFlashCopy)]
```

Parameters

Parameter	Description
logicalDrive or logicalDrives	The name of the specific logical drive for which you want to start a fresh copy-on-write operation. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you must also enclose the logical drive name in double quotation marks (" ").
userLabel	The name of the FlashCopy logical drive. Enclose the FlashCopy logical drive name in double quotation marks (" "). If you enter more than one FlashCopy logical drive name, this command fails.
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the FlashCopy repository logical drive 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 FlashCopy repository logical drive is full. You can choose to fail writes to the base logical drive (<i>failBaseWrites</i>) or fail writes to the FlashCopy logical drive (<i>failSnapShot</i>). The default value is <i>failSnapShot</i> .

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.

Re-create FlashCopy Collection

This command restarts multiple FlashCopy logical drives in one batch operation. This command makes sure that all of the FlashCopy logical drives specified in the value are valid, and then restarts each FlashCopy logical drive. You can specify a FlashCopy logical drive or a list of FlashCopy logical drives.

Syntax

```
recreate FlashCopy collection (FlashCopyLogicalDrive [LogicalDriveName]  
| FlashCopyLogicalDrives [LogicalDriveName1 ... LogicalDriveNameN])
```

Parameters

Parameter	Description
FlashCopyLogicalDrive or FlashCopyLogicalDrives	The name of the specific FlashCopy logical drive or FlashCopy logical drives for which you want to initiate a restart. Enclose the FlashCopy logical drive name in square brackets ([]). If the FlashCopy logical drive name has special characters, you must also enclose the FlashCopy logical drive name in double quotation marks (" ").

Notes

If any one of the FlashCopy logical drives does not pass the validation check, the command fails, and the FlashCopy logical drives are not re-created.

Remove Logical Drive LUN Mapping

This command removes the logical unit number (LUN) mapping.

Syntax

```
remove (allLogicalDrives | logicalDrive [logicalDriveName] |  
logicalDrives [logicalDriveName1 ... logicalDriveNameN] | accesslogicalDrive)  
lunMapping (host="hostName" |  
hostGroup="hostGroupName" | defaultGroup))
```

Parameters

Parameter	Description
allLogicalDrives	This parameter removes the LUN mapping from all of the logical drives.
logicalDrive or logicalDrives	The name of the specific logical drive that you want to remove from the LUN mapping. You can enter more than one logical drive name. Enclose the logical drive name in double quotation marks (" ") inside of square brackets ([]).

Parameter	Description
accessLogicalDrive	This parameter removes the access logical drive.
host	The name of the host to which the logical drive is mapped. Enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group that contains the host to which the logical drive 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 logical drive is mapped.

Notes

The access logical drive is the `logicalDrive` in a SAN environment that is used for communication between the storage management software and the storage subsystem controller. The access logical drive uses a LUN address and consumes 20 MB of storage space that is not available for application data storage. An access logical drive is required only for in-band managed storage subsystems.

Attention: Removing an access logical drive can damage your configuration – The agent uses the access logical drives to communicate with a storage subsystem. If you remove an access logical drive mapping for a storage subsystem from a host that has an agent running on it, the storage management software is no longer able to manage the storage subsystem through the agent.

You must use the **host** parameter and the **hostGroup** parameter when you specify a non-access logical drive or an access logical drive. The script engine ignores the **host** parameter or the **hostGroup** parameter when you use the **allLogicalDrives** parameter or the **logicalDrives** parameter.

Remove Remote Mirror

This command removes the mirror relationship between the primary logical drive and the secondary logical drive in a remote-mirror pair.

Syntax

```
remove remoteMirror (localLogicalDrive [logicalDriveName] |
localLogicalDrives [logicalDriveName1 ... logicalDriveNameN])
```

Parameters

Parameter	Description
localLogicalDrive or localLogicalDrives	The name of the primary logical drive (the logical drive on the local storage subsystem) that you want to remove. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").

Remove Volume Copy

This command removes a VolumeCopy pair.

Syntax

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

Parameters

Parameter	Description
target	The name of the target logical drive that you want to remove. Enclose the target logical drive name in square brackets ([]). If the target logical drive name has special characters, you also must enclose the target logical drive name in double quotation marks (" ").
source	The name of the source logical drive that you want to remove. Enclose the source logical drive name in square brackets ([]). If the source logical drive name has special characters, you also must enclose the source logical drive name in double quotation marks (" ").

Repair Logical Drive Parity

This command repairs the parity errors on a logical drive.

Syntax

```
repair logicaldrive [logicalDriveName] parity  
parityErrorFile="filename"  
[verbose=(TRUE | FALSE)]
```

Parameters

Parameter	Description
logicalDrive	The name of the specific logical drive for which you want to repair parity. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive 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 logical drive parity is being repaired. To capture progress details, set this parameter to TRUE. To prevent capturing progress details, set this parameter to FALSE.

Replace Drive

This command redefines the composition of an array. You can use this command to replace a disk drive with either an unassigned disk drive or a fully integrated hot spare.

Syntax

```
replace drive [enclosureID,slotID] replacementDrive=enclosureID,slotID
```

Parameters

Parameter	Description
drive	The enclosure and the slot of the disk drive that you want to replace. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID value and the slot ID value in square brackets ([]).

Parameter	Description
replacementDrive	The enclosure and the slot of the disk drive that you want to use as a replacement disk drive. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32.

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 logical drives 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 logical drives 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 logical drives.

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.

Reset Storage Subsystem Battery Install Date

| This command resets the age of the batteries in a storage subsystem to zero days. You can reset the
| batteries for an entire storage subsystem or just the battery for a specific controller or in a specific battery
| pack.

Syntax

```
reset storageSubsystem batteryInstallDate [controller=(a | b) | batteryPack=(left | right)]
```

Parameter

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.
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 batteryPack parameter only for controllers with battery packs.

Notes

| If the command statement uses the wrong parameter, an error appears.

Reset Storage Subsystem Diagnostic Data

This command resets the NVSRAM that contains the diagnostic data for the storage subsystem. 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 automatically over 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 storageSubsystem 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 storageSubsystem diagnosticData
```

Parameters

None.

Reset Storage Subsystem iSCSI Baseline

This command resets the iSCSI baseline for the storage subsystem to 0.

Syntax

```
reset storageSubsystem iscsiStatsBaseline
```

Parameters

None.

Notes

This command resets the baseline to 0 for both controllers in the storage subsystem. The purpose of resetting both of the controller baselines is to help make sure that the controller counts are synchronized between the controllers. If one controller resets but the second controller does not reset, the host is informed that the controllers are out of synchronization. The host is informed by the time stamps that are reported with the statistics.

Reset Storage Subsystem Logical Drive Distribution

This command reassigns (moves) all of the logical drives to their preferred controller.

Syntax

```
reset storageSubsystem logicalDriveDistribution
```

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 logical drives 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 logical drives.

Reset Storage Subsystem 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 storageSubsystem RLSBaseline
```

Parameters

None.

Reset Storage Subsystem 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 storageSubsystem SOCBaseline
```

Parameters

None.

Resume Remote Mirror

This command resumes a suspended Enhanced Remote Mirroring operation.

Syntax

```
resume remoteMirror (primary [logicalDriveName] |  
primaries [logicalDriveName1 ... logicalDriveNameN])  
[writeConsistency=(TRUE | FALSE)]
```

Parameters

Parameter	Description
primary or primaries	The name of the primary logical drive for which you want to resume operation. You can enter more than one primary logical drive name. Enclose the primary logical drive name in square brackets ([]). If the primary logical drive name has special characters, you also must enclose the primary logical drive name in double quotation marks (" ").
writeConsistency	The setting to identify the logical drives in this command that are in a write-consistency group or are separate. For the logical drives to be in the same write-consistency group, set this parameter to TRUE. For the logical drives to be separate, set this parameter to FALSE.

Notes

If you set the **writeConsistency** parameter to TRUE, the logical drives must be in a write-consistency group (or groups). This command resumes all write-consistency groups that contain the logical drives. For example, if logical drives A, B, and C are in a write-consistency group and they have remote counterparts A', B', and C', the resume remoteMirror logicalDrive ["A"] writeConsistency=TRUE command resumes A-A', B-B', and C-C'.

Revive Array

This command forces the specified array and its associated failed disk 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 disk drives in the array. Never try to revive a disk drive unless you are supervised by a Customer and Technical Support representative.

Syntax

```
revive array [arrayNumber]
```

Parameter

Parameter	Description
array	The alphanumeric identifier (including - and _) of the array to be set to the Optimal state. Enclose the array identifier in square brackets ([]).

Revive Drive

This command forces the specified disk 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 disk drives in the array. Never try to revive a disk drive unless you are supervised by a Customer and Technical Support representative.

Syntax

```
revive drive [enclosureID,slotID]
```

Parameter

Parameter	Description
drive	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID value and the slot ID value in square brackets ([]).

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 (" ").

Save Drive Channel Fault Isolation Diagnostic Status

This command saves the drive channel fault isolation diagnostic data that is returned from the **start drive channel fault isolation diagnostics** command. You can save the diagnostic data to a file as standard text or as XML.

See the “Start Drive Channel Fault Isolation Diagnostics” on page 122 command for more information.

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 subsystem for each disk drive.

Syntax

```
save allDrives logFile="filename"
```

Save Storage Subsystem Configuration

This command creates a script file that you can use to create the current storage subsystem logical drive configuration.

Syntax

```
save storageSubsystem configuration file="filename"  
[(allConfig | globalSettings=(TRUE | FALSE)  
logicalDriveConfigAndSettings=(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.
logicalDriveConfigAndSettings	The setting to save the logical drive configuration settings and all of the global settings to the file. To save the logical drive configuration settings and global settings, set this parameter to TRUE. To prevent saving the logical drive 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 logical drive 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.

Save Storage Subsystem Diagnostic Data

This command saves the storage subsystem 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 storageSubsystem 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 storageSubsystem 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 (" ").

Save Storage Subsystem 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 storageSubsystem (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).

Save Storage Subsystem iSCSI Statistics

This command saves the iSCSI performance statistics of the storage subsystem to a file.

Syntax

```
save storageSubsystem iscsiStatistics [raw | baseline]
file="filename"
```

Parameters

Parameter	Description
raw	The statistics collected are all statistics from the controller start of day. Enclose the parameter in square brackets ([]).

Parameter	Description
baseline	The statistics that are collected are all statistics from the time the controllers were reset to zero using the <code>reset storageSubsystem iscsiStatsBaseline</code> command. Enclose the parameter in square brackets ([]).
file	The name of the file to which you want to save the performance statistics. Enclose the file name in double quotation marks (" ").

Notes

If you have not reset the iSCSI baseline statistics since the controller start of day, the time at the start of day is the default baseline time.

This command does not automatically append a file extension to the new file. You must specify the file extension when you enter the file name.

Save Storage Subsystem 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 storageSubsystem 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 (" ").

Save Storage Subsystem RLS Counts

This command saves the read link status (RLS) counters to a file.

Syntax

```
save storageSubsystem 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 storageSubsystem RLSBaseline` command to set all of the RLS counters to 0.
2. Run the storage subsystem for a predetermined amount of time (for instance, two hours).
3. Run the `save storageSubsystem 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.

Save Storage Subsystem 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 storageSubsystem 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 storageSubsystem SOCBaseline command to set all of the SOC counters to 0.
2. Run the storage subsystem for a predetermined amount of time (for example, two hours).
3. Run the save storageSubsystem 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.

Save Storage Subsystem State Capture

This command saves the state capture of a storage subsystem to a file.

Syntax

```
save storageSubsystem 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 (" ").

Save Storage Subsystem Support Data

This command saves the support-related information of the storage subsystem to a file. Support-related information includes these items:

- The storage subsystem profile
- The Major Event Log information
- The read ink status (RLS) data
- The NVSRAM data
- Current problems and associated recovery information
- The performance statistics for the entire storage subsystem

- The persistent registration information and the persistent reservation information
- Detailed information about the current status of the storage subsystem
- The diagnostic data for the disk drive
- A recovery profile for the storage subsystem
- The unreadable sectors that are detected on the storage subsystem
- The state capture data

Syntax

```
save storageSubsystem 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 subsystem. Enclose the file name in double quotation marks (" ").

Set Array

This command defines the properties for an array.

Syntax

```
set array [arrayNumber]
addDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6)
owner=(a | b)
availability=(online | offline)
```

Parameters

Parameter	Description
array	The alphanumeric identifier (including - and _) of the array for which you want to set properties. Enclose the array identifier in square brackets ([]).
addDrives	The disk drive, by enclosure and slot location, that you want to include in the array. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
raidLevel	The RAID level for the array. Valid values are 0, 1, 3, 5, or 6.
availability	The setting for the array, which is either online or offline.
owner	The controller that owns the array. 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 array owner.

Notes

Host I/O errors might result in arrays with more than 32 logical drives. This operation also might result in internal controller reboots because the timeout period ends before the array 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.

Set Array Forced State

This command moves an array into a Forced state. Use this command if the start array import command does not move the array to an Imported state or if the import operation does not work because of hardware errors. In a Forced state, the array can be imported, and you can then identify the hardware errors.

Syntax

```
set array [arrayNumber] forcedState
```

Parameter

Parameter	Description
<code>array</code>	The alphanumeric identifier (including - and _) of the array that you want to place in a Forced state. Enclose the array identifier in square brackets ([]).

Notes®

You can move the disk drives that comprise an array from one storage subsystem to another storage subsystem. The CLI provides three commands that let you move the disk drives. The commands are start array export, start array import, and set array forcedState.

In the Forced state, you can perform an import operation on the array.

Set Controller

This command defines the attributes for the controllers.

Syntax

```
| set controller [(a | b)]  
| availability=(online | offline | serviceMode) |  
| ethernetPort [(1 | 2)]=ethernetPortOptions  
| globalNVRAMByte [nvsramOffset]=(nvsramByteSetting | nvsramBitSetting) |  
| hostNVRAMByte [hostType, nvsramOffset]=(nvsramByteSetting | nvsramBitSetting) |  
| iscsiHostPort [(1 | 2)]=iscsiHostPortOptions  
| rloginEnabled=(TRUE | FALSE) |  
| serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
<code>controller</code>	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 controller identifier in square brackets ([]). If you do not specify a controller, the firmware for the controller returns a syntax error.
<code>availability</code>	The mode for the controller. The controller modes are online, offline, or serviceMode.

Parameter	Description
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 that you want to modify by 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.
iscsiHostPort	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, the subnet mask address, the IPv4 priority, and the IPv6 priority.
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

The following options are available for the **ethernetPort** parameter:

```
enableIPv4=(TRUE | FALSE) |
enableIPv6=(TRUE | FALSE) |
IPv6LocalAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
IPv6RoutableAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
IPv6RouterAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
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) |
duplexMode=(TRUE | FALSE) |
portSpeed=[(autoNegotiate | 10 | 100 | 1000)]
```

The following options are available for the **iscsiHostPort** parameter:

```
IPv4Address=(0-255).(0-255).(0-255).(0-255) |
IPv6LocalAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
IPv6RoutableAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
IPv6RouterAddress=(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF):(0-FFFF) |
enableIPv4=(TRUE | FALSE) |
enableIPv6=(TRUE | FALSE) |
```

```

enableIPv4Vlan=(TRUE | FALSE) |
enableIPv6Vlan=(TRUE | FALSE) |
| enableIPv4Priority=(TRUE | FALSE) |
enableIPv6Priority=(TRUE | FALSE) |
IPv4ConfigurationMethod=(static | dhcp) |
IPv6ConfigurationMethod=(static | auto) |
IPv4GatewayIP=(TRUE | FALSE) |
IPv6HopLimit=[0-255] |
IPv6NdDetectDuplicateAddress=[0-256] |
IPv6NdReachableTime=[0-65535] |
IPv6NdRetransmitTime=[0-65535] |
IPv6NdTimeOut=[0-65535] |
IPv4Priority=[0-7] |
IPv6Priority=[0-7] |
IPv4SubnetMask=(0-255).(0-255).(0-255).(0-255) |
IPv4VlanID=[1-4094] |
IPv6VlanID=[1-4094] |
maxFramePayload=[framesize] |
| tcpListeningPort=[3260, 49152-65536] |

```

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 virtual disks. The specified controller no longer has any logical drives and refuses to take ownership of any more logical drives. 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.

| The **maxFramePayload** parameter is shared between IPv4 and IPv6. The payload portion of a standard Ethernet frame is set at 1500, and a jumbo Ethernet frame is set at 9000. When using jumbo frames, all of the devices that are in the network path should be capable of handling the larger frame size.

| You must set the **enableIPv4** parameter or the **enableIPv6** parameter to `TRUE` to make sure that the specific IPv4 setting or the specific IPv6 setting is applied.

| When the **duplexMode** parameter is set to `TRUE`, the selected Ethernet port is set to full duplex. The default value is half duplex (the **duplexMode** parameter is set to `FALSE`).

| The **portSpeed** parameter is expressed as megabits per second (Mb/s).

The IPv6 address space is 128 bits. It is represented by eight 16-bit hexadecimal blocks separated by colons. You may drop leading zeros, and you may use a double colon to represent consecutive blocks of zeroes.

The default value for the **IPv6HopLimit** parameter is 64. The default value for the **IPv6NdReachableTime** parameter is 30000 milliseconds. The default value for the **IPv6NdRetransmitTime** parameter is 1000 milliseconds. The default value for the **IPv6NdTimeOut** parameter is 30000 milliseconds. The default 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 module or a controller module. If the storage subsystem does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage subsystem 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 CRU or the interconnect-battery CRU, use the set enclosure 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

This command was originally defined for use with the DS4800 (Models 82, 84, 88) controller module. This command is not supported by controller modules that were shipped before the introduction of the DS4800 (Models 82, 84, 88) controller module. The DS4200 and DS4700 controllers also support this command.

Set Drive Channel Status

This command defines how the disk drive channel performs.

Syntax

```
set driveChannel [(1 | 2 | 3 | 4)]
status=(optimal | degraded)
```

Parameters

Parameter	Description
driveChannel	The identifier number of the disk drive channel for which you want to set the status. Valid disk drive channel values are 1, 2, 3, or 4. Enclose the disk drive channel number in square brackets ([]).
status	The condition of the disk drive channel. You can set the disk drive channel status to optimal or degraded.

Notes

Use the optimal value to move a degraded disk drive channel back to the Optimal state. Use the degraded value when the disk drive channel is experiencing problems, and the storage subsystem requires additional time for data transfers.

Set Drive Hot Spare

This command assigns or unassigns one or more disk drives as a hot spare.

Syntax

```
set (drive [enclosureID,slotID] |
drives [enclosureID1,slotID1 ... enclosureIDn,slotIDn])
hotSpare=(TRUE | FALSE)
```

Parameters

Parameter	Description
drive or drives	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in square brackets ([]).
hotSpare	The setting to assign the disk drive as the hot spare. To assign the disk drive as the hot spare, set this parameter to TRUE. To remove a hot spare assignment from a disk drive, set this parameter to FALSE.

Set Drive Service Action Allowed Indicator

This command turns on or turns off the Service Action Allowed indicator light on a disk drive in expansion drawers that support the Service Action Allowed indicator light feature. If the storage subsystem does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage subsystem supports the command but is unable to turn on or turn off the indicator light, this command returns an error.

Syntax

```
set (drive [enclosureID,slotID] |
drives [enclosureID1,slotID1 ... enclosureIDn,slotIDn])
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
drive or drives	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure 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.

Set Drive State

This command sets a disk drive to the Failed state. (To return a disk drive to the Optimal state, use the revive drive command.)

Syntax

```
set drive [enclosureID,slotID] operationalState=failed
```

Parameter

Parameter	Description
drive	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in square brackets ([]).

Set Enclosure Alarm

This command turns on, turns off, or mutes the audible alarm for a specific enclosure or all of the enclosures in a storage subsystem.

Syntax

```
set (allenclosures | enclosure [enclosureID]  
alarm=(enable | disable | mute))
```

Parameters

Parameter	Description
allEnclosures	The setting to select all of the enclosures in a storage subsystem that have audible alarms that you want to turn on, turn off, or mute.
enclosure	The specific enclosure that has the audible alarm that you want to turn on, turn off, or mute. Enclosure ID values are 0 to 99. Enclose the enclosure 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.)

Set Enclosure Identification

This command sets the enclosure ID of a controller module or a expansion drawer in a storage subsystem. This command is valid only for controller modules or expansion drawers that have enclosure IDs that you can set through the controller firmware. You cannot use this command for controller modules or expansion drawers that have an enclosure ID that you set with a switch.

Syntax

```
set enclosure ["serial-number"] id=enclosureID
```

Parameters

Parameter	Description
enclosure	The serial number of the controller module, controller module, or the expansion drawer for which you are setting the enclosure 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 module enclosure ID, controller module enclosure ID, or the expansion drawer enclosure ID. Enclosure ID values are 00 through 99. You do not need to enclose the enclosure ID value in parentheses.

Notes

This command originally supported the DS4800 (Models 82, 84, 88) controller module. The DS4800 (Models 82, 84, 88)-series controller modules can connect to a variety of expansion drawers, including those whose enclosure IDs are set by switches. When connecting a DS4800 (Models 82, 84, 88)-series controller module to expansion drawers whose enclosure IDs are set by switches, valid values for enclosure IDs for the controller module are 80 through 99. This range avoids conflicts with enclosure IDs that are used for attached expansion drawers.

Set Enclosure Service Action Allowed Indicator

This command turns on or turns off the Service Action Allowed indicator light on a power-fan CRU, an interconnect-battery CRU, or an environmental services monitor (ESM) CRU. If the storage subsystem does not support the Service Action Allowed indicator light feature, this command returns an error. If the storage subsystem 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 CRU, use the set controller serviceAllowedIndicator command.

Syntax

```
set enclosure [enclosureID]
(powerFan [(left | right)] |
interconnect |
battery [(left | right)] |
esm [(left | right)])
serviceAllowedIndicator=(on | off)
```

Parameters

Parameter	Description
enclosure	The enclosure where the power-fan CRU, the interconnect-battery CRU, or the ESM CRU resides. Enclosure ID values are 0 to 99. Enclose the enclosure ID value in square brackets ([]). If you do not enter a enclosure ID value, the enclosure ID of the controller module is the default value.

Parameter	Description
powerFan	The Service Action Allowed indicator light on the power-fan CRU that you want to turn on or turn off. Valid power-fan CRU identifiers are left or right. (The values left and right are with respect to the front of the DS4800 (Models 82, 84, 88)-series controller modules.) Enclose the power-fan CRU identifier in square brackets ([]).
interconnect	The Service Action Allowed indicator light for the interconnect-battery CRU (DS4800 (Models 82, 84, 88)-series controller modules only).
battery	The Service Action Allowed indicator light for a battery. Valid battery identifiers are left or right. (The values left and right are with respect to the front of the DS4800 (Models 82, 84, 88)-series controller modules.)
esm	The Service Action Allowed indicator light for an ESM CRU. Valid ESM CRU identifiers are left or right. (The values left and right are with respect to the rear of the drive expansion enclosure.)
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 enclosure 5 with the IP address of 155.155.155.155.

```
SMcli 155.155.155.155 -c "set enclosure [5] ESM [left] serviceAllowedIndicator=on";
```

Notes

This command was originally defined for use with the DS4800 (Models 82, 84, 88) controller module. This command is not supported by controller modules that were shipped before the introduction of the DS4800 (Models 82, 84, 88) controller module.

Set FlashCopy Logical Drive

This command defines the properties for a FlashCopy logical drive and lets you rename a FlashCopy logical drive.

Syntax

```
set (logical drive [logicalDriveName] |
logicalDrives [logicalDriveName1 ... logicalDriveNameN])
userLabel="FlashCopyLogicalDriveName"
warningThresholdPercent=percentValue
repositoryFullPolicy=(failBaseWrites | failFlashCopy)
```

Parameters

Parameter	Description
logicalDrive or logicalDrives	The name of the specific FlashCopy logical drive for which you want to define properties. (You can enter more than one logical drive name if you use the logicalDrives parameter). Enclose the FlashCopy logical drive name in double quotation marks (" ") inside of square brackets ([]).
userLabel	A new name that you want to give to a FlashCopy logical drive. Enclose the new FlashCopy logical drive name in double quotation marks (" ").
warningThresholdPercent	The percentage of repository capacity at which you receive a warning that the FlashCopy 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 FlashCopy processing to continue if the FlashCopy repository logical drive is full. You can choose to fail writes to the base logical drive (failBaseWrites) or fail writes to the FlashCopy logical drive (failFlashCopy). The default value is failFlashCopy .

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 FlashCopy repository logical drive or the FlashCopy logical drive.

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 logical drive-to-LUN mappings or does not have individual logical drive-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 (" ").

Parameter	Description
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 logical drive 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 logical drive 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 show storageSubsystem hostTypeTable 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 Logical drive to LUN Mappings	Host Does Not Have Individual Logical drive to LUN Mappings
<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> .	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. The following table lists the valid characters.

Table 9. 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	{		}	~	

Set Host Channel

This command defines the loop ID for the host channel.

Syntax

```
set hostChannel [hostChannelNumber]  
preferredID=portID
```

Parameters

Parameter	Description
hostChannel	<p>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 ([]).</p> <p>For all controller modules other than the DS4800 (Models 82, 84, 88)-series controller modules, valid host channel values are a1, a2, b1, or b2.</p> <p>For the DS4800 (Models 82, 84, 88)-series controller modules, valid host channel values are a1, a2, a3, a4, b1, b2, b3, or b4.</p>
preferredID	The new port ID for the specified host channel. Port ID values are 0 to 127.

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.

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.

Set iSCSI Initiator

This command sets the attributes for an iSCSI initiator.

Syntax

```
set iscsiInitiator iscsiName=new-iSCSI-ID |  
userLabel=newName |  
host=newHostName |  
chapSecret=newSecurityKey
```

Parameters

Parameter	Description
iscsiName	The name of the iSCSI initiator for which you want to set attributes.
userLabel	The new name that you want to use for the iSCSI initiator.
host	The name of the new host to which the HBA host port is connected. Enclose the host name in double quotation marks (" ").
chapSecret	The security key that you want to use to authenticate a peer connection.

Notes

Challenge Handshake Authentication Protocol (CHAP) is a protocol that authenticates the peer of a connection. CHAP is based upon the peers sharing a "secret." A secret is a security key that is similar to a password.

Use the **chapSecret** parameter to set up the security keys for initiators that require a mutual authentication.

Set iSCSI Target Properties

This command defines properties for an iSCSI target.

Syntax

```
set iscsiTarget authenticationMethod=(none | chap) |
chapSecret=securityKey |
isnsRegistration=(TRUE | FALSE) |
targetAlias=user-label
```

Parameters

Parameter	Description
authenticationMethod	The means of authenticating your iSCSI session.
chapSecret	The security key that you want to use to authenticate a peer connection.
isnsRegistration	The means of listing the iSCSI target on the iSNS server. Set the parameter to TRUE to list it.
targetAlias	The name that you want to use for the target.

Notes

Challenge Handshake Authentication Protocol (CHAP) is a protocol that authenticates the peer of a connection. CHAP is based upon the peers sharing a “secret.” A secret is a security key that is similar to a password.

Use the **chapSecret** parameter to set up the security keys for initiators that require a mutual authentication.

Set Logical Drive

This command defines the properties for a logical drive. You can use most parameters to define properties for one or more logical drives. You also can use some parameters to define properties for only one logical drive. The syntax definitions are separated to show which parameters apply to several logical drives and which apply to only one logical drive. Also, the syntax for logical drive mapping is listed separately.

Syntax Applicable to One or More Logical Drives

```
set (allLogicalDrives | logicalDrive ["logicalDriveName"] |
logicalDrives ["logicalDriveName1" ... "logicalDriveNameN"] | logicalDrive <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 Logical Drive

```
set (logicalDrive ["logicalDriveName"] | logicalDrive <wwID>)
addCapacity=logicalDriveCapacity
[addDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)]
redundancyCheckEnabled=(TRUE | FALSE)
segmentSize=segmentSizeValue
userLabel=logicalDriveName
preReadRedundancyCheck=(TRUE | FALSE)
```

Syntax Applicable to Logical Drive Mapping

```
set (logicalDrive=["logicalDriveName"] | logicalDrive <wwID> | accesslogicalDrive)
logicalUnitNumber=LUN
(host="hostName" |
hostGroup=("hostGroupName" | defaultGroup)
```

Parameters

Parameter	Description
alllogicalDrives	The properties for all logical drives in the storage subsystem.
logicalDrive or logicalDrives	The name of the specific logical drive for which you want to define properties. (You can enter more than one logical drive name if you use the logicalDrives parameter). Enclose the logical drive name in double quotation marks (" ") inside of square brackets ([]).
logicalDrive	The WWID of the logical drive for which you are setting properties. You can use the WWID instead of the logical drive name to identify the logical drive. Enclose the WWID in angle brackets (< >).
cacheFlushModifier	The maximum amount of time that data for the logical drive 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 logical drive. 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 subsystem 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.
modificationPriority	The priority for logical drive modifications while the storage subsystem is operational. Valid values are highest, high, medium, low, or lowest.
owner	The controller that owns the logical drive. 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 logical drive 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.

Parameter	Description
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 logical drive 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 disk drives to the logical drive. Specify the enclosure ID value and the slot ID value for each disk drive that you assign to the logical drive. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in square brackets ([]). Use this parameter with the addCapacity parameter if you need to specify additional disk drives to accommodate the new size.
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 disk drive in a logical drive before writing data on the next disk drive. Valid values are 8, 16, 32, 64, 128, 256, or 512.
userLabel	The new name that you want to give an existing logical drive. Enclose the new logical drive 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 logicalDrives, for example RAID 0. To check redundancy consistency, set this parameter to TRUE. For no stripe checking, set this parameter to FALSE.
accessLogicalDrive	The logical unit number for the access logical drive. The logical unit number is the only property that you can set for the access logical drive.
logicalUnitNumber	Specific logical drive-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 logical drive is mapped. Enclose the host name in double quotation marks (" ").
hostGroup	The name of the host group to which the logical drive is mapped. Enclose the host group name in double quotation marks (" "). defaultGroup is the host group that contains the host to which the logical drive is mapped.

Notes

Host I/O errors might result in arrays with more than 32 logical drives. 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 logical drive 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 logicalDrive actionProgress` command.

Access logicalDrive

The access logical drive is the logical drive in a SAN environment that is used for in-band communication between the storage management software and the storage subsystem controller. This logical drive uses a LUN address and consumes 20 MB of storage space that is not available for application data storage. An access logical drive is required only for in-band managed storage subsystems. If you specify the **accessLogicalDrive** 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.).

Value	Description
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 subsystem 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 logical drive properties. If you select the highest priority level, the logical drive 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 disk 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 subsystem 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 disk drive in a logical drive before writing data on the next disk 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 disk drive for a single request leaves other disk drives available to simultaneously service other requests.

If the logical drive 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 disk drives in the array that are used for data transfers.) In this case, multiple disk drives are used for the same request, but each disk 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 disk drives that are required to satisfy a data transfer request.

Set Remote Mirror

This command defines the properties for a remote-mirror pair.

Syntax

```
set remoteMirror (localLogicalDrive [logicalDrive] |
localLogicalDrives [logicalDriveName1 ... logicalDriveNameN])
role=(primary | secondary)
[force=(TRUE | FALSE)]
syncPriority=(highest | high | medium | low | lowest)
autoResync=(enabled | disabled)
writeOrder=(preserved | notPreserved)
writeMode=(synchronous | asynchronous)
```

Parameters

Parameter	Description
localLogicalDrive or localLogicalDrives	The name of the primary logical drive for which you want to define properties. You can enter more than one primary logical drive name. Enclose the primary logical drive name in square brackets ([]). If the primary logical drive name has special characters, you also must enclose the primary logical drive name in double quotation marks (" ").
role	The setting for the logical drive to act as the primary logical drive or the secondary logical drive. To define the logical drive as the primary logical drive, set this parameter to primary. To define the logical drive as the secondary logical drive, set this parameter to secondary. This parameter applies only when the logical drive is part of a mirror relationship.
force	The role reversal is forced if the communications link between the storage subsystems 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.
autoResync	The settings for automatic resynchronization between the primary logical drives and the secondary logical drives 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 logical drive and the secondary logical drive.• disabled – Automatic resynchronization is turned off. To resynchronize the primary logical drives and the secondary logical drive, you must run the resume remoteMirror command.
writeOrder	This parameter defines write order for data transmission between the primary logical drive and the secondary logical drive. Valid values are preserved or notPreserved.
writeMode	This parameter defines how the primary logical drive writes to the secondary logical drive. 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 logical drives and the secondary logical drives 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 `topreserved` causes the remote-mirror pair to transmit data from the primary logicalDrive to the secondary logicalDrive in the same order as the host writes to the primary logicalDrive. 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 logicalDrive has the same data as the primary logical drive.

Set Session

This command defines how you want the current script engine session to run.

Syntax

```
set session errorAction=(stop | continue)
password="storageSubsystemPassword"
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 <code>stop</code> . (This parameter defines the action for execution errors, not syntax errors. Some error conditions might override the <code>continue</code> value.)
password	The password for the storage subsystem. 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 subsystem 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 subsystem password by using the `set storageSubsystem` 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.

Set Storage Subsystem

This command defines the properties of the storage subsystem.

Syntax

```
set storageSubsystem alarm=(enable | disable | mute)
cacheBlockSize=cacheBlockSizeValue
cacheFlushStart=cacheFlushStartSize
cacheFlushStop=cacheFlushStopSize
defaultHostType=("hostTypeName" | hostTypeIdentifier)
failoverAlertDelay=delayValue |
mediaScanRate=(disabled | 1-30) |
password="password" |
userLabel="storageSubsystemName"
```

Parameters

Parameter	Description
alarm	The setting for the audible alarm. This parameter 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.
cacheBlockSize	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).
cacheFlushStart	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.
cacheFlushStop	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 cacheFlushStart parameter.
defaultHostType	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 subsystem, run the show storageSubsystem hostTypeTable 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.
failoverAlertDelay	The failover alert delay time in minutes. The valid values for the delay time are 0 to 60 minutes. The default value is 5.
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 subsystem. Enclose the password in double quotation marks (" ").

Parameter	Description
userLabel	The name for the storage subsystem. Enclose the storage subsystem 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 subsystems 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 subsystems that require large data transfer, sequential, or high-bandwidth applications.

This parameter defines the supported cache block size for all of the logical drives in the storage subsystem. Not all controller types support all cache block sizes. For redundant configurations, this parameter includes all of the logical drives that are owned by both controllers within the storage subsystem.

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 disk drive writes that are necessary to maintain the cache level, which increases system overhead and decreases performance.

When setting storage subsystem 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 disk drive read rather than reading from the cache.

Default Host Type

When you define host types, if Storage Partitioning is enabled, the default host type affects only those logical drives that are mapped in the default group. If SANshare Storage Partitioning is not enabled, all of the hosts that are attached to the storage subsystem must run the same operating system and be compatible with the default host type.

Media Scan Rate

Media scan runs on all of the logical drives in the storage subsystem that have Optimal status, do not have modification operations in progress, and have the **mediaScanRate** parameter enabled. Use the set logicalDrive command to enable or disable the **mediaScanRate** parameter.

Password

Passwords are stored on each storage subsystem. The password can be any combination of alphanumeric characters with a maximum of 30 characters.

Set Storage Subsystem Enclosure Positions

This command defines the position of the storage expansion enclosures (EXPs) in a storage subsystem configuration. You must include all of the expansion enclosures that are in the storage subsystem when you enter this command.

Syntax

```
set storageSubsystem enclosurePositions=(controller | enclosureID-list)
```

Parameter

Parameter	Description
enclosurePositions	A list of all of the enclosure IDs. The sequence of the enclosure IDs in the list defines the positions for the controller module and the expansion drawers in a storage subsystem. Valid values are 0 to 99. Enter the enclosure ID values separated by a space. Enclose the list of enclosure ID values in parentheses. For storage subsystems where the controller module has a predefined identifier that is not in the range of valid enclosure position values, use the controller value.

Notes

This command defines the position of an enclosure in a storage subsystem by the position of the enclosure ID in the **enclosurePositions** list. For example, if you have a controller module with an ID set to 84 and expansion drawers with IDs set to 1, 12, and 50, the **enclosurePositions** sequence (84 1 12 50) places the controller module in the first position, expansion drawer 1 in the second position, expansion drawer 12 in the third position, and expansion drawer 50 in the fourth position. The **enclosurePositions** sequence (1 84 50 12) places the controller module in the second position, expansion drawer 1 in the first position, expansion drawer 50 in the third position, and expansion drawer 12 in the fourth position.

Set Storage Subsystem ICMP Response

This command returns the default values for negotiable settings for sessions and connections, which represent the starting point for the storage subsystem for negotiations.

Syntax

```
set storageSubsystem icmpPingResponse=(TRUE | FALSE)
```

Parameter

Parameter	Description
icmpPingResponse	This parameter turns on or turns off Echo Request messages. Set the parameter to TRUE to turn on Echo Request messages. Set the parameter to FALSE to turn off Echo Request messages.

Notes

The Internet Control Message Protocol (ICMP) is used by operating systems in a network to send error messages, test packets, and informational messages related to the IP, such as a requested service is not available or that a host or router could not be reached. The ICMP response command sends ICMP Echo Request messages and receives Echo Response messages to determine if a host is reachable and how long packets take to get to and from that host.

Set Storage Subsystem iSNS Server IPv4 Address

This command sets the configuration method and address for an IPv4 Internet Storage Name Service (iSNS).

Syntax

```
set storageSubsystem isnsIPv4ConfigurationMethod=[static | dhcp]
isnsIPv4Address=ip-address
```

Parameters

Parameters	Description
isnsIPv4ConfigurationMethod	The method that you want to use to define the iSNS server configuration. You can enter the IP address for the IPv4 iSNS servers by selecting static. For IPv4, you can choose to have a Dynamic Host Configuration Protocol (DHCP) server select the iSNS server IP address by entering dhcp. To enable DHCP, you must set the isnsIPv4Address parameter to 0.0.0.0.
isnsIPv4Address	The IP address that you want to use for the iSNS server. Use this parameter with the static value for IPv4 configurations. If you choose to have a DHCP server set the IP address for an IPv4 Internet iSNS server, you must set the isnsIPv4Address parameter to 0.0.0.0.

Notes

The iSNS protocol facilitates the automated discovery, management, and configuration of iSCSI devices and Fibre Channel devices on a TCP/IP network. iSNS provides intelligent storage discovery and management services comparable to those found in Fibre Channel networks, which allow a commodity IP network to function in a similar capacity as a storage area network. iSNS also facilitates a seamless integration of IP networks and Fibre Channel networks, due to its ability to emulate Fibre Channel fabric services and manage both iSCSI devices and Fibre Channel devices.

The DHCP server passes configuration parameters, such as network addresses, to IP nodes. DHCP enables a client to acquire all of the IP configuration parameters that it needs to operate. DHCP lets you automatically allocate reusable network addresses.

Set Storage Subsystem iSNS Server IPv6 Address

This command sets the IPv6 address for the iSNS server.

Syntax

```
set storageSubsystem isnsIPv6Address=ip-address
```

Parameters

Parameters	Description
isnsIPv6Address	The IPv6 address that you want to use for the iSNS server.

Notes

The iSNS protocol facilitates the automated discovery, management, and configuration of iSCSI and Fibre Channel devices on a TCP/IP network. iSNS provides intelligent storage discovery and management services comparable to those found in Fibre Channel networks, permitting a commodity IP network to function in a similar capacity as a storage area network. iSNS also facilitates a seamless integration of IP networks and Fibre Channel networks, due to its ability to emulate Fibre Channel fabric services, and manage both iSCSI devices and Fibre Channel devices. iSNS thereby provides value in any storage network that has iSCSI devices, Fibre Channel devices, or any combination thereof.

Set Storage Subsystem iSNS Server Listening Port

This command sets the iSNS server listening port.

Syntax

```
set storageSubsystem isnsListeningPort=integer
```

Parameter

Parameter	Description
isnsListeningPort	The IP address that you want to use for the iSNS server listening port. The range of values for the listening port is 49152 to 65535. The default value is 3205.

Notes

A listening port resides on the database server and is responsible for these activities:

- Listening (monitoring) for incoming client connection requests
- Managing the traffic to the server

When a client requests a network session with a server, a listener receives the actual request. If the client information matches the listener information, then the listener grants a connection to the database server.

Set Storage Subsystem iSNS Server Refresh

This command refreshes the network address information for the iSNS server. This command is valid for only IPv4.

Syntax

```
set storageSubsystem isnsServerRefresh
```

Parameters

None.

Notes

If the DHCP server is not operating at full capability, or if the DHCP server is unresponsive, the refresh operation can take between two and three minutes to complete.

The set storageSubsystem isnsServerRefresh command returns an error if you did not set the configuration method to DHCP. To set the configuration method to DHCP, use the set storageSubsystem isnsIPv4ConfigurationMethod command.

Set Storage Subsystem Learn Cycle

This command sets the learn cycle for the battery backup unit, which enables the storage management software to predict the remaining battery life. Learn cycles run at set intervals, and they store the results for software analysis.

Syntax

```
set storageArray learnCycleDate (daysToNextLearnCycle=integer-literal | day=string-literal) time=HH:MM
```

Parameters

Parameter	Description
daysToNextLearnCycle	Valid values are 0 through 7, where 0 is immediately and 7 is in seven days. The daysToNextLearnCycle parameter takes place up to seven days after the next scheduled learn cycle.
day	Valid values for the day parameter include the days of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday). Setting the day causes the next learn cycle to be scheduled on the specified day, after the currently scheduled learn cycle.
time	The time in 24-hour format; for example 8:00 a.m. is entered as 08:00. Nine o'clock p.m. is entered as 21:00, and 9:30 p.m. is entered as 21:30.

Notes

You can set the learn cycle to occur only once during a seven-day period.

The **time** parameter selects a specific time that you want to run the learn cycle. If a value is not entered, the command uses a default value of 00:00 (midnight).

If the day and time specified are in the past, the next learn cycle takes place on the next possible day specified.

Set Storage Subsystem Redundancy Mode

This command sets the redundancy mode of the storage subsystem to either simplex or duplex.

Syntax

```
set storageSubsystem redundancyMode=(simplex | duplex)
```

Parameter

Parameter	Description
redundancyMode	Use simplex mode when you have a single controller. Use duplex mode when you have two controllers.

Set Storage Subsystem Time

This command sets the clocks on both controllers in a storage subsystem by synchronizing the controller clocks with the clock of the host from which you run this command.

Syntax

```
set storageSubsystem time
```

Parameters

None.

Set Unnamed Discovery Session

This command enables the storage subsystem to participate in unnamed discovery sessions.

Syntax

```
set iscsiTarget unnamedDiscoverySession=(TRUE | FALSE)
```

Parameter

Parameter	Description
<code>unnamedDiscoverySession</code>	This parameter turns on or turns off unnamed discovery sessions. Set the parameter to TRUE to turn on unnamed discovery sessions. Set the parameter to FALSE to turn off unnamed discovery sessions.

Notes

Discovery is the process where initiators determine the targets that are available. Discovery occurs at power-on/initialization and also if the bus topology changes, for example, if an extra device is added.

An unnamed discovery session is a discovery session that is established without specifying a target ID in the login request. For unnamed discovery sessions, neither the target ID nor the target portal group ID are available to the targets.

Set VolumeCopy

This command defines the properties for a VolumeCopy 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 logical drive for which you want to define properties. Enclose the target logical drive name in square brackets ([]). If the target logical drive name has special characters, you also must enclose the target logical drive name in double quotation marks (" ").
source	The name of the source logical drive for which you want to define properties. Enclose the source logical drive name in square brackets ([]). If the source logical drive name has special characters, you also must enclose the source logical drive name in double quotation marks (" ").
copyPriority	The priority that the VolumeCopy 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 logical drive or only read from the target logical drive. To write to the target logical drive, set this parameter to FALSE. To prevent writing to the target logical drive, set this parameter to TRUE.

Notes

When you use this command, you can specify one or more of the optional parameters.

Show Array

This command returns this information about an array:

- The status (online or offline)
- The disk drive type (Fibre Channel, SATA, or SAS)
- Enclosure loss protection (yes or no)
- The current owner (the controller in slot A or the controller in slot B)
- The associated logical drives and free capacity
- The associated disk drives

Syntax

```
show array [arrayNumber]
```

Parameter

Parameter	Description
array	The alphanumeric identifier of the array (including - and _) for which you want to show information. Enclose the array identifier in square brackets ([]).

Show Array Export Dependencies

This command shows a list of dependencies for the disk drives in a array that you want to move from one storage subsystem to a second storage subsystem.

Syntax

```
show array [arrayNumber] exportDependencies
```

Parameter

Parameter	Description
array	The alphanumeric identifier (including - and _) of the array for which you want to show export dependencies. Enclose the array identifier in square brackets ([]).

Notes

This command spins up the disk drives in a array, reads the DACstore, and shows a list of import dependencies for the array. The array must be in an Exported state or a Forced state.

Show Array Import Dependencies

This command shows a list of dependencies for the disk drives in an array that you want to move from one storage subsystem to a second storage subsystem.

Syntax

```
show array [arrayNumber] importDependencies [cancelImport=(TRUE | FALSE)]
```

Parameters

Parameter	Description
array	The alphanumeric identifier (including - and _) of the array for which you want to show import dependencies. Enclose the array identifier in square brackets ([]).
cancelImport	The setting to spin the disk drives back down after the array dependencies have been read. To spin down the disk drives, set this parameter to TRUE. To let the disk drives stay spinning, set this parameter to FALSE.

Notes

This command returns the dependencies of a specific array, 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 disk drives back down.

The show array importDependencies command must be run before the start array import command.

Show Controller

For each controller in a storage subsystem, 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 logical drives (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 subsystem.
controller	The setting to return information about a specific controller in the storage subsystem. 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 subsystem.

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.

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 subsystem.
controller	The setting to return information about a specific controller in the storage subsystem. 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 show storageSubsystem hostTypeTable 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.

Show Current iSCSI Sessions

This command returns information about an iSCSI session.

Syntax

```
show iscsiInitiator iscsiSessions [iscsiInitiator | iscsiTarget]
```

Parameters

Parameter	Description
<i>iscsiInitiator</i>	The name of the iSCSI initiator for which you want to obtain session information. Enclose the iSCSI initiator name in square brackets ([]).
<i>iscsiTarget</i>	The name of the iSCSI target for which you want to obtain session information. Enclose the iSCSI target name in square brackets ([]).

Notes

If you enter this command without defining any arguments, this command returns information about all of the iSCSI sessions that are currently running. To limit the information that is returned, enter a specific iSCSI initiator or a specific iSCSI target. This command then returns information about the session for only the iSCSI initiator or the iSCSI target that you named.

Show Drive

For each disk drive in the storage subsystem, this command returns the following information:

- The total number of disk drives
- The type of disk drive (Fibre Channel, SATA, or SAS)
- Information about the basic disk drive:
 - The enclosure location and the slot location
 - The status
 - The capacity
 - The data transfer rate
 - The product ID
 - The firmware level
- Information about the disk drive channel:
 - The enclosure location and the slot location
 - The preferred channel
 - The redundant channel
- Hot spare coverage
- Details for each disk drive

Depending on the size of your storage subsystem, this information can be several pages long. In addition, the disk drive information is returned for the show storageSubsystem profile command.

Syntax

```
show (allDrives [driveType=(fibre | SATA | SAS)] |  
drive [enclosureID,slotID] |  
drives [enclosureID1,slotID1 ... enclosureIDn,slotIDn]) summary
```

Parameters

Parameter	Description
allDrives	The setting to return information about all of the disk drives in the storage subsystem.
driveType	The type of disk drive for which you want to retrieve information. Valid disk drive types are fibre, SATA, or SAS.
drive or drives	The enclosure and the slot where the disk drive resides. You can enter enclosure IDs and slot IDs for one or several disk drives. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID values and the slot ID values in parentheses.
summary	The setting to return the status, the capacity, the data transfer rate, the product ID, and the firmware version for the specified disk drives.

Notes

To determine information about the type and location of all of the disk drives in the storage subsystem, use the **allDrives** parameter.

To determine the information about the Fibre Channel, SATA, or SAS disk drives in the storage subsystem, use the **driveType** parameter.

To determine the type of disk drive in a specific location, use the **drive** parameter, and enter the enclosure ID and the slot ID for the disk drive.

Show Drive Channel Stats

This command shows the cumulative data transfer for the disk drive channel and error information. If the controller has automatically degraded a disk drive channel, this command also shows interval statistics. When you use this command, you can show information about one specific disk drive channel, several disk drive channels, or all disk drive channels.

Syntax

```
show (driveChannel [(1 | 2 | 3 | 4)] |  
driveChannels [1 2 3 4] | allDriveChannels) stats
```

Parameters

Parameter	Description
driveChannel	The identifier number of the disk drive channel for which you want to show information. Valid disk drive channel values are 1, 2, 3, 4, 5, 6, 7, or 8. Enclose the disk drive channel in square brackets ([]).
driveChannels	The identifier number of the disk drive channels for which you want to show information. Valid disk drive channel values are 1, 2, 3, 4, 5, 6, 7, or 8. Enclose the disk drive channels in square brackets ([]) with the disk drive channel value separated by a space.
allDriveChannels	The identifier that selects all of the disk drive channels.

Show Drive Download Progress

This command returns the status of firmware downloads for the disk drives that are targeted by the download drive firmware command or the download storageSubsystem 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 disk 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.

Show Enhanced Remote Mirroring Logical Drive Candidates

This command returns information about the candidate logical drives on a remote storage subsystem that you can use as secondary logical drives in a Enhanced Remote Mirroring configuration.

Syntax

```
show remoteMirror candidates primary="logicalDriveName"  
remoteStorageSubsystemName="storageSubsystemName"
```

Parameters

Parameter	Description
primary	The name of the local logical drive that you want for the primary logical drive in the remote-mirror pair. Enclose the primary logical drive name in double quotation marks (" ").
remoteStorageSubsystemName	The remote storage subsystem that contains possible logical drives for a secondary logical drive. If the remote storage subsystem name has special characters, you must also enclose the remote storage subsystem name in double quotation marks (" ").

Show Enhanced Remote Mirroring Logical Drive Synchronization Progress

This command returns the progress of data synchronization between the primary logical drive and the secondary logical drive in a Enhanced Remote Mirroring configuration. This command shows the progress as a percentage of data synchronization that has been completed.

Syntax

```
show remoteMirror (localLogicalDrive ["logicalDriveName"] |  
localLogicalDrives ["logicalDriveName1" ... "logicalDriveNameN"])  
synchronizationProgress
```

Parameter

Parameter	Description
<code>localLogicalDrive</code> or <code>localLogicalDrives</code>	The name of the primary logical drive of the remote mirror pair for which you want to check synchronization progress. Enclose the primary logical drive name in double quotation marks (" ") inside of square brackets []).

Show Host Ports

For all of the HBA host ports that are connected to a storage subsystem, this command returns this information:

- The HBA host port identifier
- The HBA host port name
- The HBA host type

Syntax

```
show allHostPorts
```

Parameters

None.

Show Logical Drive

For the logical drives in a storage subsystem, this command returns the following information:

- The number of logical drives
- The name
- The status
- The capacity
- The RAID level
- The array where the logical drive is located
- Details:
 - The logical drive ID
 - The subsystem ID
 - The disk drive type (Fibre Channel, SATA, SAS)
 - Enclosure loss protection
 - The preferred owner
 - The current owner
 - The segment size
 - The modification priority
 - The read cache status (enabled or disabled)
 - The write cache status (enabled or disabled)
 - The write cache without batteries status (enabled or disabled)
 - The write cache with mirroring status (enabled or disabled)
 - The flush write cache after time
 - The cache read prefetch setting (TRUE or FALSE)
 - The enable background media scan status (enabled or disabled)
 - The media scan with redundancy check status (enabled or disabled)
- The FlashCopy repository logical drives
- The mirror repository logical drives

- The FlashCopy logical drives
- The FlashCopy copies

Syntax

```
show (allLogicalDrives | logicalDrive [logicalDriveName] |
logicalDrives [logicalDriveName1 ... logicalDriveNameN]) summary
```

Parameters

Parameter	Description
allLogicalDrives	The setting to return information about all of the logical drives in the storage subsystem.
logicalDrive or logicalDrives	The name of the specific logical drive for which you are retrieving information. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").
summary	The setting to return a concise list of information about the logical drives.

Show Logical Drive Action Progress

For a long-running operation that is currently running on a logical drive, this command returns information about the logical drive 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 logicalDrive ["logicalDriveName"] actionProgress
```

Parameters

Parameter	Description
logicalDrive	The name of the logical drive that is running the long-running operation. Enclose the logical drive name in double quotation marks (" ") inside of square brackets ([]).

Show Logical Drive Performance Statistics

This command returns information about the performance of the logical drives in a storage subsystem.

Syntax

```
show (alllogicalDrives | logicalDrive [logicalDriveName]
logicalDrives [logicalDriveName1 ... logicalDriveNameN]) performanceStats
```

Parameters

Parameter	Description
alllogicalDrives	The setting to return performance statistics about all of the logical drives in the storage subsystem.

Parameter	Description
<code>logicalDrive</code> or <code>logicalDrives</code>	The name of the specific logical drive for which you are retrieving performance statistics. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").

Notes

Before you run the `show logicalDrive performanceStat` command, run the `set session performanceMonitorInterval` command and the `set session performanceMonitorIterations` command to define how often you collect the statistics.

Show Logical Drive Reservations

This command returns information about the logical drives that have reservations.

Syntax

```
show (allLogicalDrives | logicalDrive [logicalDriveName] |
logicalDrives [logicalDriveName1 ... logicalDriveNameN]) reservations
```

Parameters

Parameter	Description
<code>allLogicalDrives</code>	The setting to return persistent reservation information about all of the logical drives in the storage subsystem.
<code>logicalDrive</code> or <code>logicalDrives</code>	The name of the specific logical drive for which you are retrieving persistent reservation information. You can enter more than one logical drive name. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").

Show Storage Subsystem

This command returns configuration information about the storage subsystem. The parameters return lists of values for the components and features in the storage subsystem. 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 subsystem profile is shown (which is the same information as if you entered the `profile` parameter).

Syntax

```
show storageSubsystem profile batteryAge connections defaultHostType
healthStatus hostTypeTable hotSpareCoverage features time logicalDriveDistribution
summary
```

Parameters

Parameter	Description
<code>profile</code>	The parameter to show all of the properties of the logical components and the physical components that comprise the storage subsystem. The information appears in several screens.

Parameter	Description
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 subsystem.
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.
hotSpareCoverage	The parameter to show information about which logical drives of the storage subsystem have hot spare coverage and which logical drives do not.
features	The parameter to show a list of the feature identifiers for all enabled features in the storage subsystem.
time	The parameter to show the current time to which both controllers in the storage subsystem are set.
logicalDriveDistribution	The parameter to show the current controller owner for each logical drive in the storage subsystem.
summary	The parameter to show a concise list of information about the storage subsystem configuration.

Notes

The **profile** parameter shows detailed information about the storage subsystem. 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 storageSubsystem` command that looks like this example:

```
c:\...\...\client>smcli 123.45.67.89 -c "show storageSubsystem profile;" -o
"c:\\folder\\storageSubsystemprofile.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.

“v625002” on page v625002 shows the type of information that is 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.

The DS4800 (Models 82, 84, 88)-series controller modules do not support the `show storageSubsystem batteryAge` command.

Show Storage Subsystem Auto Configure

This command shows the default auto-configuration that the storage subsystem creates if you run the `autoConfigure storageSubsystem` command. If you want to determine whether the storage subsystem 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 disk 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 storageSubsystem autoConfiguration
[driveType=(fibre | SATA | SAS)
raidLevel=(0 | 1 | 3 | 5 | 6)
arrayWidth=numberOfDrives
arrayCount=numberOfLogical driveGroups
logicalDrivesPerArrayCount=numberOfLogicalDrivesPerArray
hotSpareCount=numberOfHotspares
segmentSize=segmentSizeValue
cacheReadPrefetch=(TRUE | FALSE)]
```

Parameters

Parameter	Description
driveType	The type of disk drives that you want to use for the storage subsystem. Valid disk drive types are fibre, SATA, or SAS. The driveType parameter is not required if only one type of disk drive is in the storage subsystem.
raidLevel	The RAID level of the array that contains the disk drives in the storage subsystem. Valid RAID levels are 0, 1, 3, 5, or 6.
logicalDriveGroupWidth	The number of disk drives in a array in the storage subsystem, which depends on the capacity of the disk drives. Use integer values.
logicalDriveGroupCount	The number of arrays in the storage subsystem. Use integer values.
logicalDrivesPerGroupCount	The number of equal-capacity logical drives per array. Use integer values.
hotSpareCount	The number of hot spares that you want in the storage subsystem. Use integer values.
segmentSize	The amount of data (in KB) that the controller writes on a single disk drive in a logical drive before writing data on the next disk 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.

Show Storage Subsystem Host Topology

This command returns the storage partition topology, the host type labels, and the host type index for the host storage subsystem.

Syntax

```
show storageSubsystem hostTopology
```

Parameters

None.

Show Storage Subsystem LUN Mappings

This command returns information from the storage subsystem profile about the LUN mappings in the storage subsystem. 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 storageSubsystem 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 ([]).

Show Storage Subsystem Negotiation Defaults

This statement returns information about connection-level settings that are subject to initiator-target negotiation.

Syntax

```
show storageSubsystem iscsiNegotiationDefaults
```

Parameters

None.

Notes

Information returned includes RAID controller enclosure default settings (that is, those settings that are the starting point for negotiation) and the current active settings.

Show Storage Subsystem Unreadable Sectors

This command returns a table of the addresses of all of the sectors in the storage subsystem that cannot be read. The table is organized with column headings for the following information:

1. Logical drive user label
2. LUN
3. Accessible by (host or host group)
4. Date/time
5. Logical drive-relative logical block address (hexadecimal format – 0xnxxxxxxxx)
6. Disk drive location (enclosure t, slot s)
7. Disk drive-relative logical block address (hexadecimal format – 0xnxxxxxxxx)
8. Failure type

The data is sorted first by the logical drive user label and second by the logical block address (LBA). Each entry in the table corresponds to a single sector.

Syntax

```
show storageSubsystem unreadableSectors
```

Parameters

None.

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 (" ").

Show Unconfigured iSCSI Initiators

This command returns a list of initiators that have been detected by the storage subsystem but are not yet configured into the storage subsystem topology.

Syntax

```
show storageSubsystem unconfiguredIscsiInitiators
```

Parameters

None.

Show VolumeCopy

This command returns this information about VolumeCopy operations:

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

You can retrieve information about a specific VolumeCopy pair or all of the VolumeCopy pairs in the storage subsystem.

Syntax

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

Parameters

Parameter	Description
allLogicalDrives	The setting to return information about VolumeCopy operations for all of the VolumeCopy pairs.
source	The name of the source logical drive about which you want to retrieve information. Enclose the source logical drive name in double quotation marks (" ") inside of square brackets ([]).
target	The name of the target logical drive about which you want to retrieve information. Enclose the target logical drive name in double quotation marks (" ") inside of square brackets ([]).

Show VolumeCopy Source Candidates

This command returns information about the candidate logical drives that you can use as the source for a VolumeCopy operation.

Syntax

```
show volumeCopy sourceCandidates
```

Parameters

None.

Show VolumeCopy Target Candidates

This command returns information about the candidate logical drives that you can use as the target for a VolumeCopy operation.

Syntax

```
show volumeCopy source ["sourceName"] targetCandidates
```

Parameter

Parameter	Description
source	The name of the source logical drive for which you are trying to find a candidate target logical drive. Enclose the source logical drive name in double quotation marks (" ") inside of square brackets ([]).

Start Array Defragment

This command starts a defragment operation on the specified array.

Note: Defragmenting an array starts a long-running operation that you cannot stop.

Syntax

```
start array [arrayNumber] defragment
```

Parameter

Parameter	Description
array	The alphanumeric identifier of the array (including - and _) that you want to defragment. Enclose the array identifier in square brackets ([]).

Notes

Host I/O errors might result in the arrays with more than 32 logical drives. This operation also might result in internal controller reboots because the timeout period ends before the array definition is set. If you experience this issue, quiesce the host I/O operations, and try the command again.

Start Array Export

This command moves a array into an Exported state. Then you can remove the disk drives that comprise the array and reinstall the disk drives in a different storage subsystem.

Note: Within the array, you cannot move logical drives that are associated with the premium features from one storage subsystem to another storage subsystem.

Syntax

```
start array [arrayNumber] export
```

Parameter

Parameter	Description
array	The alphanumeric identifier of the array (including - and _) that you want to export. Enclose the array identifier in square brackets ([]).

Notes

When this command is successful, you can run the start array import command to finish moving the array to a Complete state, which makes the array available to the new storage subsystem.

If this command is unsuccessful because hardware problems prevented the completion of the export, use the set array forceState command. The set array forceState command lets you use the start array import command to import a array.

After the array is in an Exported state or a Forced state, you can remove the disk drives that comprise the array from the storage subsystem. You can reinstall the disk drives in a different storage subsystem.

Start Array Import

This command moves a array into a Complete state to make a newly introduced array available to its new storage subsystem. The array must be in an Exported state or a Forced state before you run this command. Upon successfully running the command, the array is operational.

Note: Within the array, you cannot move logical drives that are associated with the premium features from one storage subsystem to another storage subsystem.

Syntax

```
start array [arrayNumber] import
```

Parameter

Parameter	Description
array	The alphanumeric identifier of the array (including - and _) that you want to import. Enclose the array identifier in square brackets ([]).

Notes

Higher-level logical drives that are specifically related to premium features (FlashCopy, Enhanced Remote Mirroring, VolumeCopy, host-to-logical drive mapping, and persistent reservations) are removed as part of the import operation.

The show array importDependencies command must be run before the start array import command.

Start Array Locate

This command identifies the disk drives that are logically grouped together to form the specified array by blinking the indicator lights on the disk drives. (Use the stop array locate command to turn off the indicator lights on the disk drives.)

Syntax

```
start array [arrayNumber] locate
```

Parameter

Parameter	Description
array	The alphanumeric identifier of the array (including - and _) for which you want to locate the disk drives that belong to that array. Enclose the array identifier in square brackets ([]).

Start Drive Channel Fault Isolation Diagnostics

This command runs the drive channel fault isolation diagnostics and stores the results.

Syntax

```
start driveChannel [(1 | 2 | 3 | 4 | 5 | 6 | 7 | 8)] controller [(a | b)] faultDiagnostics  
{testDevices=[all | controller=(a | b) | esms=[enclosureID1 (left | right), ... enclosureIDn  
(left | right)] | drives=[enclosureID1, slotID1, ... , enclosureIDn, slotIDn]] |  
dataPattern=(fixed | pseudoRandom) | patternNumber=[(0xhexadecimal | integerLiteral)] |  
maxErrorCount=integer | testIterations=integer | timeout=timeInterval}
```

Parameter

Parameter	Description
controller	The identifier letter of the controller that you want to test. Valid controller identifier values are a or b, where a is the controller in slot A, and b is the controller in slot B.
testDevices	The identifiers of the devices (controllers, ESMs, or drives) that you want to test. You can specify <i>all</i> or enter the specific identifiers for the devices that you want to diagnose.
dataPattern	The method of repeatability that you want to test.

Parameter	Description
patternNumber	The hexadecimal data pattern you want to use to run the test. This number can be any hexadecimal number between 0000 to FFFF.
maxErrorCount	The number of errors that you want to accept before terminating the test.
testIterations	The number of times that you want to repeat the test.
timeout	The length of time in minutes that you want to run the test.

Notes

Use the **save driveChannel faultDiagnostics** command and the **stop driveChannel faultDiagnostics** command in association with the **start driveChannel faultDiagnostics** command. These commands are needed to save the diagnostic test results to a file and to stop the diagnostic test.

You can also stop this command at any time by pressing **Ctrl+C**

Start Drive Channel Locate

This command identifies the expansion drawers that are connected to a specific disk drive channel by turning on the indicator lights for the expansion drawer that is connected to the disk drive channel. (Use the **stop driveChannel locate** command to turn off the indicator lights on the expansion drawer.)

Syntax

```
start driveChannel [(1 | 2 | 3 | 4)] locate
```

Parameter

Parameter	Description
driveChannel	The identifier number of the disk drive channel that you want to locate. Valid values for the identifier number for the disk drive channel are 1, 2, 3, 4, 5, 6, 7, or 8. Enclose the disk drive channel identifier number in square brackets ([]).

Start Drive Initialize

This command starts disk drive initialization.

Attention: Possible damage to the storage subsystem configuration – As soon as you enter this command, all user data is destroyed.

Syntax

```
start drive [enclosureID,slotID] initialize
```

Parameter

Parameter	Description
drive	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID value and the slot ID value in square brackets ([]).

Start Drive Locate

This command locates a disk drive by turning on an indicator light on the disk drive. (Run the stop drive locate command to turn off the indicator light on the disk drive.)

Syntax

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

Parameter

Parameter	Description
drive	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID value and the slot ID value in square brackets ([]).

Start Drive Reconstruction

This command starts reconstructing a disk drive.

Syntax

```
start drive [enclosureID,slotID] reconstruct
```

Parameter

Parameter	Description
drive	The enclosure and the slot where the disk drive resides. Enclosure ID values are 0 to 99. Slot ID values are 1 to 32. Enclose the enclosure ID value and the slot ID value in square brackets ([]).

Start Enclosure Locate

This command locates an enclosure by turning on the indicator light. (Use the stop enclosure locate command to turn off the indicator light for the enclosure.)

Syntax

```
start enclosure [enclosureID] locate
```

Parameter

Parameter	Description
enclosure	The enclosure that you want to locate. Enclosure ID values are 0 to 99. Enclose the enclosure ID value in square brackets ([]).

Start Enhanced Remote Mirroring Synchronization

This command starts Enhanced Remote Mirroring synchronization.

Syntax

```
start remoteMirror primary ["logicalDriveName"] synchronize
```

Parameter

Parameter	Description
primary	The name of the primary logical drive for which you want to start synchronization. Enclose the primary logical drive name in double quotation marks (" ") inside of square brackets ([]).

Start iSCSI DHCP Refresh

This command initiates a refresh of the DHCP parameters for the iSCSI interface. If the configuration method for the interface is not set to DHCP, the procedure returns an error.

Syntax

```
start storageSubsystem [iscsi-host-port] dhcpRefresh
```

Parameter

Parameter	Description
iscsi-host-port	The identifier of the port on the storage subsystem on which you want to refresh the DHCP parameters. Enclose the iSCSI host port name in square brackets ([]).

Notes

This operation ends the iSCSI connections for the portal and brings down the portal temporarily.

Start Logical Drive Initialization

This command starts the formatting of a logical drive in a storage subsystem.

Note: Formatting a logical drive starts a long-running operation that you cannot stop.

Syntax

```
start logicalDrive [logicalDriveName] initialize
```

Parameter

Parameter	Description
logicalDrive	The name of the logical drive for which you are starting the formatting. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you also must enclose the logical drive name in double quotation marks (" ").

Start Storage Subsystem Locate

This command locates a storage subsystem by turning on the indicator lights for the storage subsystem. (Use the stop storageSubsystem locate command to turn off the indicator lights for the storage subsystem.)

Syntax

```
start storageSubsystem locate
```

Parameters

None.

Stop Array Locate

This command turns off the indicator lights on the disk drives that were turned on by the start array locate command.

Syntax

```
stop array locate
```

Parameters

None.

Stop Drive Channel Fault Isolation Diagnostics

| This command stops the drive channel fault isolation diagnostics, which stops the start drive channel fault isolation diagnostics command before it completes.

| See the “Start Drive Channel Fault Isolation Diagnostics” on page 122 command for more information.

| Syntax

|

```
stop driveChannel faultDiagnostics
```

| Parameters

| None.

| Notes

| None.

| Minimum firmware level

| 7.3x.xx.xx

Stop Drive Channel Locate

This command turns off the indicator lights on the expansion drawers that were turned on by the start driveChannel locate command.

Syntax

```
stop driveChannel locate
```

Parameters

None.

Stop Drive Locate

This command turns off the indicator light on the disk drive that was turned on by the start drive locate command.

Syntax

```
stop drive locate
```

Parameters

None.

Stop Enclosure Locate

This command turns off the indicator light on the enclosure that was turned on by the start enclosure locate command.

Syntax

```
stop enclosure locate
```

Parameters

None.

Stop FlashCopy

This command stops a copy-on-write operation.

Syntax

```
stop flashcopy (logicaDrive [logicaDriveName] |  
logicaDrives [logicaDriveName1 ... logicaDriveNameN])
```

Parameters

Parameter	Description
logicaDrive or logicaDrives	The name of the specific logical drive for which you want to stop a copy-on-write operation. You can enter more than one logical drive name. Enclose the logical drive names in square brackets ([]). If the logical drive names have special characters, you also must enclose the logical drive names in double quotation marks (" ").

Stop iSCSI Session

This command forces the termination of an iSCSI session.

Syntax

```
stop storageSubsystem iscsiSession [integer]
```

Parameter

Parameter	Description
iscsiSession	The identifier number of the iSCSI session. Enclose the identifier number of the iSCSI session in square brackets ([]).

Stop Storage Subsystem Drive Firmware Download

This command stops a firmware download to the disk drives in a storage subsystem that was started with the download storageSubsystem driveFirmware command. This command does not stop a firmware download that is already in progress to a disk drive. This command stops all firmware downloads to disk drives that are waiting for the download.

Syntax

```
stop storageSubsystem driveFirmwareDownload
```

Parameters

None.

Stop Storage Subsystem Locate

This command turns off the indicator lights on the storage subsystem that were turned on by the start storageSubsystem locate command.

Syntax

```
stop storageSubsystem locate
```

Parameters

None.

Stop VolumeCopy

This command stops a VolumeCopy operation.

Syntax

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

Parameters

Parameter	Description
target	The name of the target logical drive for which you want to stop a VolumeCopy operation. Enclose the target logical drive name in square brackets ([]). If the target logical drive name has special characters, you also must enclose the target logical drive name in double quotation marks (" ").
source	The name of the source logical drive for which you want to stop a VolumeCopy operation. Enclose the source logical drive name in square brackets ([]). If the source logical drive name has special characters, you also must enclose the source logical drive name in double quotation marks (" ").

Suspend Remote Mirror

This command suspends a Enhanced Remote Mirroring operation.

Syntax

```
suspend remoteMirror (primary [primaryLogicalDriveName]  
primaries [primaryLogicalDriveName1 ... primaryLogicalDriveNameN])  
writeConsistency=(TRUE | FALSE)
```

Parameters

Parameter	Description
primary or primaries	The name of the logical drive for which you want to suspend operation. Enclose the logical drive name in square brackets ([]). If the logical drive name has special characters, you must also enclose the logical drive name in double quotation marks (" ").
writeConsistency	This parameter defines whether the logical drives identified in this command are in a write-consistency group or are separate. For the logical drives in the same write-consistency group, set this parameter to TRUE. For the logical drives that are separate, set this parameter to FALSE.

Notes

If you set the **writeConsistency** parameter to TRUE, the logical drives must be in a write-consistency group (or groups). This command suspends all write-consistency groups that contain the logical drives. For example, if logical drives A, B, and C are in a write-consistency group and they have remote counterparts A', B', and C', the command:

```
suspend  
remoteMirror logicalDrive ["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 logicalDrives=["A", "D"] writeConsistency=TRUE
```

suspends both write-consistency groups.

Script Commands Listed by Function

Controller Commands

“Clear Drive Channel Statistics” on page 36

“Diagnose Controller” on page 57

“Enable Controller” on page 62

“Reset Controller” on page 71

“Save Controller NVSRAM” on page 74

“Set Controller” on page 81
“Set Controller Service Action Allowed Indicator” on page 84
“Set Drive Channel Status” on page 84
“Set Host Channel” on page 90
“Show Controller” on page 108
“Show Controller NVSRAM” on page 109
“Show Drive Channel Stats” on page 111
“Start Drive Channel Locate” on page 123
“Stop Drive Channel Locate” on page 126

Disk Drive Commands

“Download Drive Firmware” on page 59
“Replace Drive” on page 70
“Revive Drive” on page 74
| “Save Drive Channel Fault Isolation Diagnostic Status” on page 75
“Save Drive Log” on page 75
“Set Drive Hot Spare” on page 85
“Set Drive Service Action Allowed Indicator” on page 85
“Set Drive State” on page 86
“Show Drive” on page 110
“Show Drive Download Progress” on page 111
“Start Drive Channel Fault Isolation Diagnostics” on page 122
“Start Drive Initialize” on page 123
“Start Drive Locate” on page 124
“Start Drive Reconstruction” on page 124
“Stop Drive Locate” on page 127

Enclosure Commands

“Download Environmental Card Firmware” on page 60
“Set Enclosure Alarm” on page 86

“Set Enclosure Identification” on page 86

“Set Enclosure Service Action Allowed Indicator” on page 87

“Start Enclosure Locate” on page 124

“Stop Enclosure Locate” on page 127

Host Topology Commands

“Create Host” on page 44

“Create Host Group” on page 44

“Create Host Port” on page 45

“Delete Host” on page 55

“Delete Host Group” on page 55

“Delete Host Port” on page 56

“Set Host” on page 89

“Set Host Group” on page 91

“Set Host Port” on page 91

“Show Host Ports” on page 113

iSCSI Commands

“Create iSCSI Initiator” on page 46

“Delete iSCSI Initiator” on page 56

“Reset Storage Subsystem iSCSI Baseline” on page 72

“Save Storage Subsystem iSCSI Statistics” on page 77

“Set iSCSI Initiator” on page 92

“Set iSCSI Target Properties” on page 92

“Show Current iSCSI Sessions” on page 110

“Show Unconfigured iSCSI Initiators” on page 119

“Start iSCSI DHCP Refresh” on page 125

“Stop iSCSI Session” on page 127

Enhanced Remote Mirroring Commands

“Activate Enhanced Remote Mirroring Feature” on page 31

“Check Remote Mirror Status” on page 36

“Create Remote Mirror” on page 52

“Deactivate Remote Mirror” on page 55

“Diagnose Remote Mirror” on page 58

“Re-create Enhanced Remote Mirroring Repository Logical Drive” on page 65

“Remove Remote Mirror” on page 69

“Resume Remote Mirror” on page 73

“Set Remote Mirror” on page 97

“Show Enhanced Remote Mirroring Logical Drive Candidates” on page 112

“Show Enhanced Remote Mirroring Logical Drive Synchronization Progress” on page 112

“Start Enhanced Remote Mirroring Synchronization” on page 124

“Suspend Remote Mirror” on page 129

Session Command

“Set Session” on page 99

FlashCopy Commands

“Create FlashCopy Logical Drive” on page 41

“Re-create FlashCopy” on page 67

“Re-create FlashCopy Collection” on page 68

“Set FlashCopy Logical Drive” on page 88

“Stop FlashCopy” on page 127

Storage Subsystem Commands

“Activate Storage Subsystem Firmware” on page 33

“Autoconfigure Storage Subsystem” on page 33

“Autoconfigure Storage Subsystem Hot Spares” on page 35

“Clear Storage Subsystem Configuration” on page 38

“Clear Storage Subsystem Event Log” on page 38

- “Clear Storage Subsystem Firmware Pending Area” on page 39
- “Disable Storage Subsystem Feature” on page 59
- “Download Storage Subsystem Drive Firmware” on page 61
- “Download Storage Subsystem Firmware/NVSRAM” on page 61
- “Download Storage Subsystem NVSRAM” on page 62
- “Enable Storage Subsystem Feature Key” on page 63
- “Reset Storage Subsystem Battery Install Date” on page 71
- “Reset Storage Subsystem Diagnostic Data” on page 72
- “Reset Storage Subsystem RLS Baseline” on page 73
- “Reset Storage Subsystem SOC Baseline” on page 73
- “Reset Storage Subsystem Logical Drive Distribution” on page 72
- “Save Storage Subsystem Configuration” on page 75
- “Save Storage Subsystem Diagnostic Data” on page 76
- “Save Storage Subsystem Events” on page 77
- “Save Storage Subsystem Performance Statistics” on page 78
- “Save Storage Subsystem RLS Counts” on page 78
- “Save Storage Subsystem SOC Counts” on page 79
- “Save Storage Subsystem State Capture” on page 79
- “Save Storage Subsystem Support Data” on page 79
- “Set Storage Subsystem” on page 100
- | “Set Storage Subsystem Learn Cycle” on page 105
- “Set Storage Subsystem Redundancy Mode” on page 105
- “Set Storage Subsystem Time” on page 106
- “Set Storage Subsystem Enclosure Positions” on page 102
- “Show Storage Subsystem” on page 115
- “Show Storage Subsystem Auto Configure” on page 116
- “Show Storage Subsystem Host Topology” on page 117
- “Show Storage Subsystem LUN Mappings” on page 118

“Show Storage Subsystem Unreadable Sectors” on page 118

“Start Storage Subsystem Locate” on page 125

“Stop Storage Subsystem Drive Firmware Download” on page 128

“Stop Storage Subsystem Locate” on page 128

Uncategorized Commands

“Set Storage Subsystem ICMP Response” on page 102

“Set Storage Subsystem iSNS Server IPv4 Address” on page 103

“Set Storage Subsystem iSNS Server IPv6 Address” on page 103

“Set Storage Subsystem iSNS Server Listening Port” on page 104

“Set Storage Subsystem iSNS Server Refresh” on page 104

“Set Unnamed Discovery Session” on page 106

“Show Storage Subsystem Negotiation Defaults” on page 118

“Show String” on page 119

Logical Drive Commands

“Check Logical Drive Parity” on page 35

“Clear Logical Drive Reservations” on page 37

“Clear Logical Drive Unreadable Sectors” on page 37

“Create RAID Logical Drive (Automatic Drive Select)” on page 46

“Create RAID Logical Drive (Free Capacity Base Select)” on page 48

“Create RAID Logical Drive (Manual Drive Select)” on page 50

“Delete LogicalDrive” on page 57

“Recover RAID Logical Drive” on page 64

“Remove Logical Drive LUN Mapping” on page 68

“Repair Logical Drive Parity” on page 70

“Set Logical Drive” on page 93

“Show Logical Drive” on page 113

“Show Logical Drive Action Progress” on page 114

“Show Logical Drive Performance Statistics” on page 114

“Show Logical Drive Reservations” on page 115

“Start Logical Drive Initialization” on page 125

VolumeCopy Commands

“Create Volume Copy” on page 54

“Recopy Volume Copy” on page 63

“Remove Volume Copy” on page 69

“Set VolumeCopy” on page 106

“Show VolumeCopy” on page 119

“Show VolumeCopy Source Candidates” on page 120

“Show VolumeCopy Target Candidates” on page 120

“Stop VolumeCopy” on page 128

Array Commands

“Create Array” on page 39

“Delete Array” on page 55

“Revive Array” on page 74

“Set Array” on page 80

“Set Array Forced State” on page 81

“Show Array” on page 107

“Show Array Export Dependencies” on page 107

“Show Array Import Dependencies” on page 108

“Start Array Defragment” on page 120

“Start Array Export” on page 121

“Start Array Import” on page 121

“Start Array Locate” on page 122

“Stop Array Locate” on page 126

Chapter 4. Configuring a Storage Subsystem

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

- Controllers
- Disk drives
- Hot spares
- Arrays
- Logical drives
- RAID technology
- Hosts
- Host groups
- Host bus adapter (HBA) host ports
- Logical unit numbers (LUNs)

Configuring a RAID storage subsystem requires caution and planning to make sure that you define the correct RAID level and configuration for your storage subsystem. You configure a storage subsystem to create logical drives, which are addressable by the hosts, from a collection of disk drives. The commands described in this chapter help you to set up and run a RAID storage subsystem. 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 subsystem. 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 3, “Script Commands,” on page 19.

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

The following figure shows a host using a computer system [3], and shows the storage subsystem with a controller module [1] and two drive expansion enclosures [2]. The storage subsystem has at least one controller and disk drives. The disk drives are grouped together to form a logical structure. The storage subsystem is physically placed in a cabinet as shown on the left in the figure.

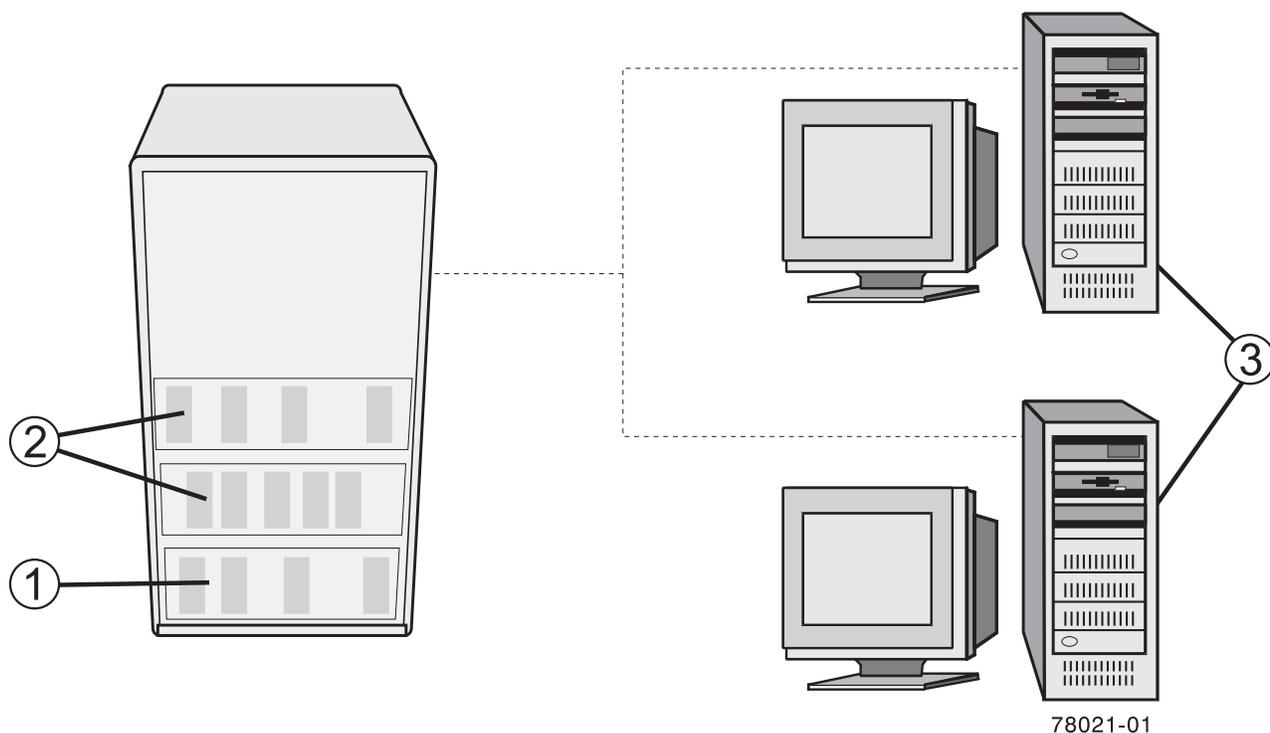


Figure 1. Host and Storage Relationship

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

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

Controllers

All storage subsystems have one or two controllers. The controllers are circuit-board assemblies that manage data and communication between the hosts and the storage subsystem. The controller manages the data flow between the hosts and the disk 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 subsystem, and the buses along which the data flows.

The controllers are located in a controller module, 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 subsystem has only one controller, the controller must be in slot A. Controllers are identified by using model numbers.

Controller models DS4400 and DS4500 use minihubs; two connected to each controller. When viewed from the rear of the controller module, 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 disk drives. When viewed from the rear of the controller module, the disk drive minihubs are numbered from left to right 4, 3, 2, 1. The script commands use these numbers to identify the disk drive channels.

Controllers manage the interface by using the controller firmware to transmit and receive commands between the hosts and the disk 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 modules and controller modules incorporate all host connections and expansion drawer 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.

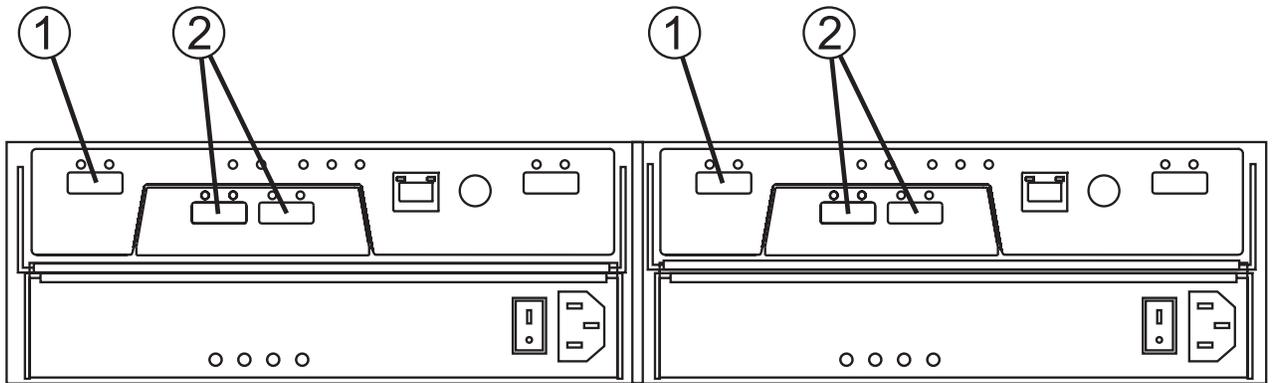
The following table lists the controller modules that do not use minihubs, the type of host port, and the number of host ports.

Table 10. Host Ports and the Type of Host Interfaces on Controller Modules

Model	Available Host Ports	Type of Host Interface
DS3200 controller module (without optional SAS expansion card)	1	SAS
DS3200 controller module (with optional SAS expansion card)	3	SAS
DS3300 controller module	2	iSCSI
DS3400 controller module	2	Fibre Channel
DS4100, DS4200, and DS4300	2	Fibre Channel
DS4700 (Model 70) controller module	2	Fibre Channel
DS4700 (Model 72) controller module	4	Fibre Channel
DS4400, DS4500, and DS4800	4	Fibre Channel

The DS4800 (Models 82, 84, 88)-series controller module has four host ports and four channel ports on each controller.

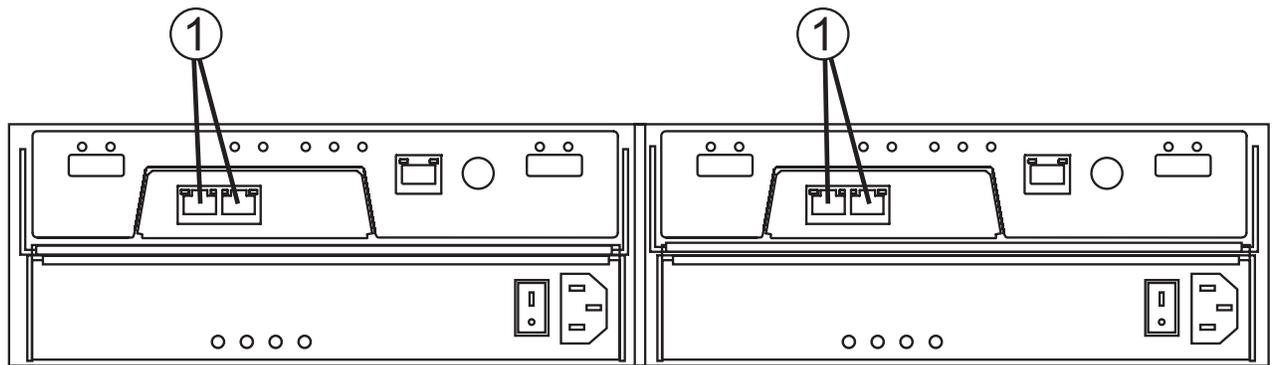
The DS3200 controller module has three host ports [1] [2], which are numbered from left to right: host port 1, host port 2, and host port 3 as shown in the following figure:



78016-20

Figure 2. DS3200 Host Ports

The DS3300 and DS3400 controller modules have two host ports [1] on each controller, which are numbered from left to right: host port 1 and host port 2 as shown in the following figure:



78053-02

Figure 3. DS3400 Host Ports

The host ports on the DS4200 and DS4700 (Model 72) controller module 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, as shown in the following figure:

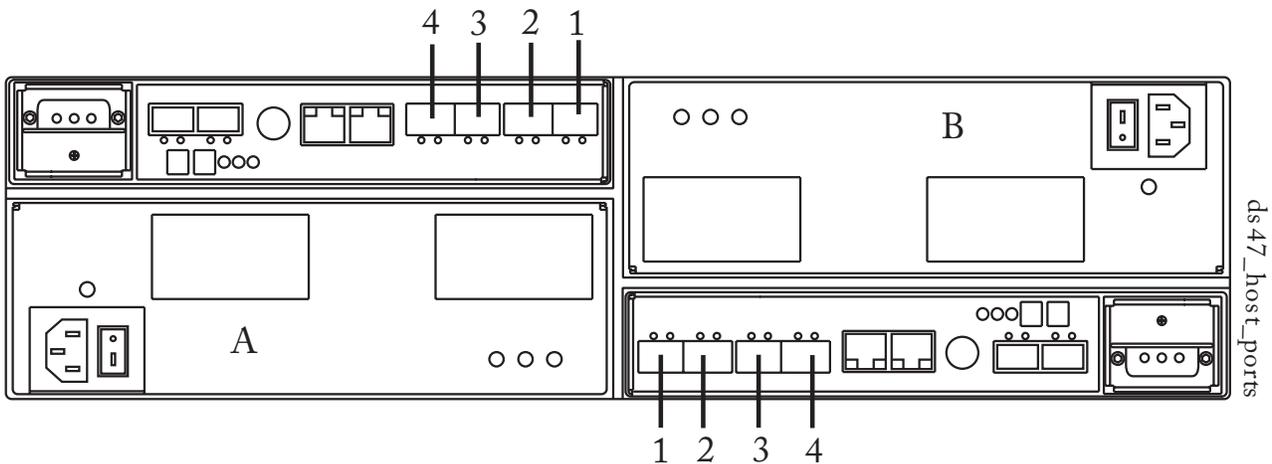


Figure 4. DS4200 and DS4700 Host Ports

The controller in the DS4800 controller module can have up to four host channels with one port for each channel; up to two drive channels with two ports per channel (for a total of four drive ports); and up to two Ethernet ports. In the DS4800 (Models 82, 84, 88) controller module, 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, and a4 (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, and b4.

On the controllers, the HBA host ports are labeled 1, 2, 3, and 4, as shown in the following figure:

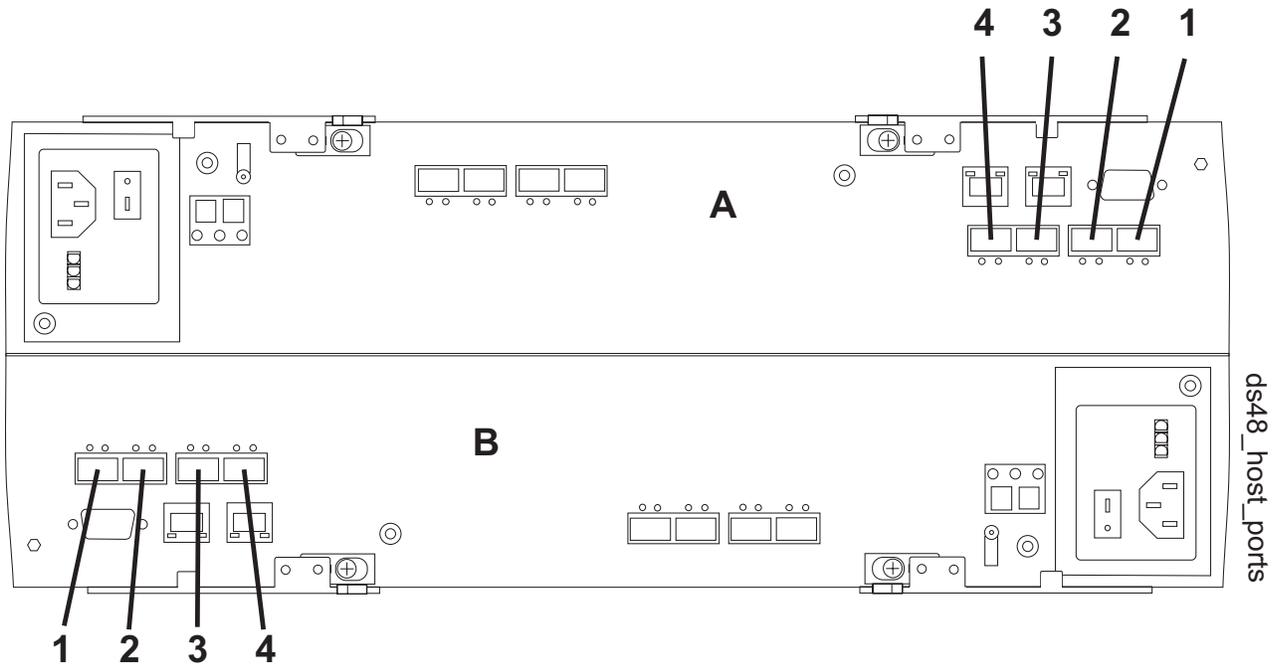


Figure 5. DS4800 Host Ports

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.

Disk Drives

The disk drives provide the actual storage of the data that is sent to the storage subsystem. The disk drives are mounted in a expansion drawer. The expansion drawer is an enclosure that, in addition to the disk drives, contains power supplies, fans, environmental cards, and other supporting components.

Disk drives are located in a storage subsystem by enclosure ID and slot ID. Enclosure ID values are 0 to 99. In older enclosures, enclosure ID values are set during installation by switches on the rear of the enclosures. In newer enclosures, enclosure ID values are set automatically when the power is applied.

The slot ID is the disk drive position in the expansion drawer. Slot ID values range from 1 to 16. A expansion drawer can contain either 10, 12, 14, or 16 disk drives. The total number of disk drives in a storage subsystem depends on the model of the controller and the type of expansion drawer.

Note: A maximum of seven expansion drawers can be on a channel when mixing 14-slot expansion drawers and 16-slot expansion drawers.

The following table lists, by controller number and expansion drawer capacity, the maximum number of disk drives in a storage subsystem.

Table 11. Maximum Number of Disk Drives

Controller Model	10-Disk Drive Expansion Drawer	12-Disk Drive Expansion Drawer	14-Disk Drive Expansion Drawer	16-Disk Drive Expansion Drawer
DS3200		48		
DS3300		48		
DS3400		48		
DS4300-SCU			14	
DS4300			112	100-110 ¹
DS4200			100-110 ¹	112
DS4700			112	112
DS4400	220		224	
DS4500	220		224	224
DS4800			224	224
DS5100			224	224
DS5300			224	224

¹
Note: There are maximum limits to the number of Fibre Channel hard drives that you can configure in redundant drive loop pairs. The maximum number of drives dictates a fixed number of supported 14-disk and 16-disk drive enclosure combinations. For more information, please refer to Chapter 1 of the *DS4000 Hard Drive and Storage Expansion Enclosure Installation and Migration Guide*.

Hot Spares

A hot spare is a disk drive that acts as a standby in the event that a disk drive containing data fails. The hot spare is a disk drive that has not been assigned to a particular array and, as such, can be used in any array. You can use the hot spare feature with RAID 1, RAID 3, RAID 5, or RAID 6.

Note: Some DS3000 and DS4000 controller firmware versions do not support RAID 6. (Firmware version 7.3x.xx.xx for the DS5000 supports RAID 6.) For more information, see the DS3000 and DS4000 controller firmware readme files or the interoperability matrices.

If a disk drive in a array fails, the controllers automatically replace the failed disk drive with a hot spare. The controllers use redundancy data to reconstruct the data from the failed disk drive onto the hot spare. To be most effective, the disk drive that you assign as a hot spare must have a capacity equal to or greater than the capacity of the largest disk drive in the storage subsystem. The hot spare must be the same type of disk drive as the disk drive that failed (for example, a Serial Advanced Technology Attachment [SATA] hot spare cannot replace a Fibre Channel hot spare).

You can assign disk drives to act as hot spares manually or have the script commands automatically assign hot spares. If you manually assign a disk drive to be a hot spare, you must identify the disk drive by enclosure 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 subsystem. The maximum number of hot spare disk drives per storage subsystem is 15.

Arrays

An array is a set of disk drives that are logically grouped together by the controllers in a storage subsystem. After you create an array, you can create one or more logical drives in the array. An array is identified by a sequence number that is defined by the controller firmware when you created the array.

To create an array, you must define the capacity and the RAID level.

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

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

The number of disk drives you can include in an array is constrained by the capacity of each disk drive. The following table relates the capacity of individual disk drives to the number of disk drives that you can include in an array.

Table 12. Maximum Number of Disk Drives in an Array Based on Capacity

	Controller firmware PRIOR to 7.10 (DS4000) or 6.xx (DS3000)	Controller firmware 7.10 or later (DS4000 / DS5000 only)
Number of drives per array	30 for all RAID levels	<ul style="list-style-type: none">RAID 3 and RAID 5: 30 drivesRAID 0 and RAID 1: Maximum of drives supported for a given storage subsystem
Maximum logical drive size	2 TB	Number of drives in an array multiplied by the size of the smallest capacity drive in the array

Note: Some of the disk drive capacity contains metadata information, so the actual usable capacity is somewhat less. Also, some storage subsystems limit the size of the array to 2 TB, and some host operating systems limit logical drive size to 2 TB or less.

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

Note: Some storage subsystems permit different drive types in the same enclosure; however, you cannot have a combination of different disk drives in the same array.

The RAID level is the level of data protection that you want to define for your storage subsystem. The RAID level that you choose affects storage capacity. When you configure your storage subsystem, 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 subsystem. (For a description of the RAID levels that you can define by using the script commands, see “RAID Levels” on page 145.)

Logical Drives

A logical drive is a logical component (object) that is the basic structure that is created on the storage subsystem to store data. A logical drive is a contiguous subsection of an array that is configured to meet application needs for data availability and I/O performance. The IBM Storage Manager client software administers a logical drive as if the logical drive is one “disk drive” for data storage. Logical drives are

identified by names or labels that users choose. The logical drive names can be any combination of alphanumeric characters, hyphens (-), and underscores (_). The maximum length of a logical drive name is 30 characters.

The script commands support the following types of logical drives:

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

Note: You must use the access logical drive only for in-band-managed storage subsystems.

- **FlashCopy logical drive** – A logical point-in-time image of another logical drive. A flashcopy logical drive 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 flashcopy repository logical drive.
- **FlashCopy repository logical drive** – A special logical drive in the storage subsystem that is created as a resource for a flashcopy logical drive. A flashcopy repository logical drive contains flashcopy data and copy-on-write data for a particular flashcopy logical drive.
- **Base logical drive** – A standard logical drive from which you create a flashcopy logical drive. The term “base logical drive” is used only to show the relationship between a standard logical drive from which you are taking the point-in-time image and a flashcopy logical drive.
- **Primary logical drive** – A standard logical drive in an Enhanced Remote Mirroring relationship. The primary logical drive accepts host data transfers and stores application data. When you first create the mirror relationship, data from the primary logical drive is copied in its entirety to the associated secondary logical drive.
- **Secondary logical drive** – A standard logical drive in an Enhanced Remote Mirroring relationship that maintains a mirror (or copy) of the data from its associated primary logical drive. The secondary logical drive 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 logical drive to a primary role.
- **Mirror repository logical drive** – A special logical drive in an Enhanced Remote Mirroring configuration that is created as a resource for each controller in both the local storage subsystem and the remote storage subsystem. The controller stores mirroring information on this logical drive, 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 subsystems.

Note: FlashCopy Logical Drive and Enhanced Remote Mirroring are premium features that you must activate before you can use them. For more information about FlashCopy logical drives, see Chapter 5, “Using the FlashCopy Premium Feature,” on page 163. For more information about Enhanced Remote Mirroring, see Chapter 6, “Using the Enhanced Remote Mirroring Premium Feature,” on page 171.

The number and capacity of the logical drives in your storage subsystem depends on the type of controller in the storage subsystem. The following table lists the maximum values for the logical drives in a storage subsystem by specific controllers.

Table 13. Logical Drive Specifications by Supported Controller Model

Specification	DS3200, DS3300, DS3400	DS4200, DS4700	DS4800	DS5000
Maximum number of logical drives per storage subsystem	256	1024	2048	2048
Maximum number of logical drives per array	256	256	256	256
Maximum logical drive size (less 12 GB for overhead)	2TB	See Table 12 on page 143		
Maximum number of disk drives per array using RAID 5	30	See Table 12 on page 143		
Maximum number of remote mirrors	N/A	64	128	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 disk drives can have a maximum logical drive size of 112 TB (224 x 500) when controller firmware version 7.10.xx.xx or later is installed in the DS4000 storage subsystem, or when controller firmware version 7.30.xx.xx is installed in the DS5000 storage subsystem.

RAID Levels

The RAID level defines a storage architecture in which the storage capacity on the disk drives in a array 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 disk 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 disk 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 disk 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 array. All redundancy information for a array is stored within the array. The following table lists the RAID levels and describes the configuration capabilities of each level.

Table 14. RAID Level Configurations

RAID Level	Configuration
0	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 disk drives in the array. If a single disk drive fails, all of the associated logical drives fail and all data is lost. RAID 0 is suited for noncritical data. It is not recommended for high-availability needs.

Table 14. RAID Level Configurations (continued)

RAID Level	Configuration
1	<p>Striping mirroring mode – RAID 1 uses disk drive mirroring to create an exact copy from one disk drive to another disk drive. A minimum of two disk 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 disk drives simultaneously. If one disk drive in a pair fails, the system can instantly switch to the other disk drive without any loss of data or service. Only half of the disk drives in the array are available for user data. If a single disk drive fails in a RAID 1 array, all of the associated logical drives become degraded, but the mirror disk drive provides access to the data. RAID 1 can survive multiple disk drive failures as long as no more than one failure occurs per mirrored pair. If a disk drive pair fails, all of the associated logical drives fail, and all data is lost.</p>
3	<p>High-bandwidth mode – RAID 3 stripes both user data and redundancy data (in the form of parity) across the disk drives. The equivalent of the capacity of one disk 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 disk drive fails in a RAID 3 array, all of the associated logical drives become degraded, but the redundancy data lets the data be reconstructed. If two or more disk drives fail, all of the associated logical drives 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 disk drives. The equivalent of the capacity of one disk 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 disk drive fails in a RAID 5 array, all of the associated logical drives become degraded, and the redundancy data permits the data to be reconstructed. If two or more disk drives fail, all of the associated logical drives 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 disk drives. A minimum of five disk drives are required for a RAID 6 array. The equivalent capacity of two disk 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 disk drive sizes. Recovery from a second disk drive failure in the same array is possible. If two disk drives fail in a RAID 6 array, all of the associated logical drives become degraded, but the redundancy data permits the data to be reconstructed. If three or more disk drives fail, all of the associated logical drives fail, and all data is lost.</p>

Note: RAID 6 is supported only with the following requirements:

- Controller firmware version 7.xx.xx.xx and higher must be installed.
- The controllers must be capable of supporting the P+Q calculation.
- You must purchase and install a premium feature key to use RAID 6 and to use dynamic RAID-level migration. (See “Set Array” on page 80 to set your array to RAID 6.)

The DS4800 storage subsystem is not supported. However, the DS4200 and DS4700 storage subsystems are supported.

Hosts

A host is a computer that is attached to the storage subsystem for accessing the logical drives in the storage subsystem. The host is attached to the storage subsystem through HBA host ports, which are connectors on host bus adapter circuit boards. You can define specific logical drive-to-LUN mappings to an individual host or assign the host to a host group that shares access to one or more logical drives. 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, Solaris, or Linux). Specifying the host type lets the controllers in the storage subsystem 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 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 logical drives. 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.

Host Bus Adapter Host Ports

A host bus adapter (HBA) provides the physical connection from the host to the storage subsystem. 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 subsystem, the Storage Manager 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 subsystem and, if you choose, change them. If you move an HBA host port, you must remap any logical drive-to-LUN mappings. Access to your data is lost until you remap the logical drives.

The maximum number of HBA host ports that you can logically define for your storage subsystem depends on the type of controller in the storage subsystem. The following table lists the maximum number of HBA host ports that you can define.

Table 15. Maximum Number of HBA Host Ports per Controller

Controller Models	Maximum Number of Host Ports
DS3200	64
DS3300	64
DS3400	64
DS4100	256
DS4300	256
DS4200	256 (before controller firmware 7.xx.xx.xx)
	1024 (with controller firmware 7.xx.xx.xx and higher)
DS4400	512
DS4500	512

Table 15. Maximum Number of HBA Host Ports per Controller (continued)

DS4700	256 (before controller firmware 7.xx.xx.xx)
	1024 (with controller firmware 7.xx.xx.xx and higher)
DS4800	512 (before controller firmware 7.xx.xx.xx)
	2048 (with controller firmware 7.xx.xx.xx and higher)
DS5100	2048 (with controller firmware 7.xx.xx.xx and higher)
DS5300	2048 (with controller firmware 7.xx.xx.xx and higher)

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 logical drives in a storage subsystem. The hosts identify the logical drives that they want to access using the LUN values. When you create a logical drive, the firmware assigns the LUN values, or you can assign LUN values when you enable the Storage Partitioning premium feature. A logical drive 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 subsystem, each host might use the same LUN to access different logical drives. The LUNs might be the same, but the logical drives 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 Subsystem

When you configure a storage subsystem, 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 array
- 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 Enhanced Remote Mirroring premium feature and the FlashCopy Logical Drive premium feature).

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

Determining What Is on Your Storage Subsystem

Even when you create a configuration on a storage subsystem that has never been configured, you still need to determine the hardware features and software features that are to be included with the storage subsystem. When you configure a storage subsystem 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 subsystem. For example, consider the case where you want to create a new array on unassigned disk drives. Before you create a new array, you must determine which disk

drives are available. The commands that are described in this section help you to determine the components and the features in your storage subsystem.

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

- A detailed profile of the components and features in the storage subsystem
- 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 logical drive in the storage subsystem

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

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

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

The `show storageSubsystem profile` command returns detailed information about the storage subsystem. 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:\...\...\client>smcli 123.45.67.89 -c "show storageSubsystem profile;"  
-o c:\folder\storageSubsystemprofile.txt
```

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

Note: 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.

Chapter 9, “Examples of Information Returned by the Show Commands,” on page 215 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 subsystem features and components, use the **summary** parameter. The command looks like this example:

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

Following is the type of information that is returned by the `show storageSubsystem` command with the **summary** parameter.

```
PROFILE FOR STORAGE SUBSYSTEM: example9 (x/xx/xx 9:58:32 AM)  
SUMMARY-----  
Number of controllers: 1  
Number of arrays: 2  
Total number of logical drives (includes an access logical drive): 3 of 2048 used  
Number of standard logical drives: 2
```

```
Number of access logical drives: 1
Number of flashcopy repositories: 0
Number of flashcopy logical drives: 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 logical drive: LUN 7 (see Mappings section for details)
Default host type: Linus (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 subsystem. The information returned by each of the show commands is the same as the information returned by the show storageSubsystem profile command, but the information is constrained to the specific component. For information about a command, refer to the page number shown.

- “Show Controller” on page 108
- “Show Drive” on page 110
- “Show Drive Channel Stats” on page 111
- “Show Storage Subsystem Host Topology” on page 117
- “Show Storage Subsystem LUN Mappings” on page 118
- “Show Logical Drive” on page 113
- “Show Array” on page 107
- “Show Logical Drive Reservations” on page 115

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

- “Show Controller” on page 108
- “Show Enhanced Remote Mirroring Logical Drive Candidates” on page 112
- “Show Storage Subsystem Auto Configure” on page 116
- “Show Storage Subsystem Unreadable Sectors” on page 118
- “Show VolumeCopy Source Candidates” on page 120
- “Show VolumeCopy Target Candidates” on page 120
- “Show Logical Drive Performance Statistics” on page 114

There are other commands that can help you learn about your storage subsystem. To see a list of the commands, see Chapter 3, “Script Commands,” on page 19. These commands are organized by the storage subsystem activities that the commands support, such as logical drive commands, host commands, enclosure commands, and others.

Clearing the Configuration

If you want to create a completely new configuration on a storage subsystem that already has an existing configuration, use the `clear storageSubsystem` configuration command. This command deletes all of the existing configuration information, including all of the arrays, logical drives, and hot spare definitions from the controller memory.

Attention: Possible damage to the storage subsystem configuration – As soon as you run this command, the existing storage subsystem configuration is deleted.

The command takes this form:

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

With controller firmware 7.xx.xx.xx and higher, this command supports two parameters that you can use to limit the amount of configuration information removed:

- **all** – Removes the entire configuration of the storage subsystem, including security information and identification information. Removing all of the configuration information returns the storage subsystem to its initial state. (This is what the command will assume if no parameters are specified.)
- **arrays** – Removes the logical drive configuration and the array configuration, but leaves the rest of the configuration intact.

Note: With controller firmware earlier than 7.xx.xx.xx,, there are no parameters. The default is the action described for the **all** parameter.

If you want to create new arrays and logical drives within the storage subsystem, you can use the `clear storageSubsystem` configuration command with the **arrays** parameter to remove existing arrays in a pre-existing configuration. This action destroys the pre-existing configuration. Use the `clear storageSubsystem` configuration command only when you create a new configuration.

Using the Auto Configure Command

The `autoConfigure storageSubsystem` command creates the arrays on a storage subsystem, the logical drives in the arrays, and the hot spares for the storage subsystem. When you use the `autoConfigure storageSubsystem` command, you define these parameters:

- The type of disk drives (Fibre, SATA, SAS)
- The RAID level
- The number of disk drives in an array
- The number of arrays
- The number of logical drives in each array
- The number of hot spares
- The size of each segment on the disk drives
- A read ahead multiplier

After you define these parameters, the Storage Manager software creates the arrays, the logical drives, and the hot spares. The controllers assign array numbers and logical drive numbers as they are created. After the Storage Manager software creates the initial configuration, you can use the `set logicalDrive` command to define logical drive labels.

Before you run the `autoConfigure storageSubsystem` command, run the `show storageSubsystem autoConfigure` command. The latter command returns a list of parameter values that the Storage Manager

software uses to automatically create a storage subsystem. 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 storageSubsystem` command. If you are satisfied with the parameter values that the `show storageSubsystem autoConfiguration` command returns, run the `autoConfigure storageSubsystem` command without new parameter values.

The `autoConfigure storageSubsystem` command takes this form:

```
autoConfigure storageSubsystem
[driveType=(fibre | SATA | SAS)
raidLevel=(0 | 1 | 3 | 5 | 6) |
arrayWidth=numberOfDrives |
arrayCount=numberOfLogicalDriveGroups |
logicalDrivesPerArrayCount=numberOfLogicalDrivesPerArray |
hotSpareCount=numberOfHotSpares |
segmentSize=segmentSizeValue |
cacheReadPrefetch=(TRUE | FALSE)]
```

When you use the `autoConfigure storageSubsystem` 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 array creation algorithms, which produce arrays with improved performance and more information about disk drive and logical drive attributes so the user can make better choices when configuring logical drives manually.

The **`arrayWidth`** parameter defines the number of unassigned disk drives that you want to use for each new array.

The **`arrayCount`** parameter defines the number of new arrays that you want in the storage subsystem.

The **`LogicalDrivesPerArrayCount`** parameter defines the number of logical drives that you want in each array.

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

The **`segmentSize`** parameter defines the amount of data, in KB, that the controller writes on a single disk drive in a logical drive before writing data on the next disk 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 disk drives that are needed to satisfy an I/O request.

Using a single disk drive for a single request leaves other disk 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 logical drive with a segment size of 8.

If the logical drive 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 disk drives in the array that are used for data storage. In this environment, multiple disk drives are used for the same request, but each disk 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 disk 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 subsystem 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.

The following table lists the default values for the segment size and cache read prefetch settings for different storage subsystem uses.

Table 16. Default Values for Segment Size and Cache Read Prefetch

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

After you have finished creating the arrays and the logical drives by using the `autoConfigure storageSubsystem` command, you can further define the properties of the logical drives in a configuration by using the `set logicalDrive` command. (For a description about how to further define your configuration, see “Modifying Your Configuration” on page 156.)

Example of the Auto Configuration Command

```
c:\...\...\client>smcli 123.45.67.89 -c "autoConfigure
storageSubsystem driveType=fibre raidLevel=5 arrayWidth=8
arrayCount=3 logicalDrivesPerArrayCount=4 hotSpareCount=2
segmentSize=8 cacheReadPrefetch=TRUE;"
```

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

Using the Create LogicalDrive Command

Use the `create logicalDrive` command to create new storage subsystem logical drives in three ways:

- Create a new logical drive while simultaneously creating a new array to which you assign the disk drives
- Create a new logical drive while simultaneously creating a new array to which the Storage Manager software assigns the disk drives
- Create a new logical drive in an existing array

You must have unassigned disk drives in the array. You do not need to assign the entire capacity of the array to a logical drive.

Creating Logical Drives with User-Assigned Disk Drives

When you create a new logical drive and assign the disk drives you want to use, the Storage Manager software creates a new array. The controller firmware assigns a array number to the new array. The command takes this form:

```
create logicalDrive drives=(enclosureID1,slotID1..enclosureIDn,slotIDn)
raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=logicalDrivesName
[capacity=logicalDrivesCapacity] [owner=(a | b)
cacheReadPrefetch=(TRUE | FALSE)
segmentSize=segmentSizeValue]
enclosureLossProtect=(TRUE | FALSE)]
```

Note: The **capacity** parameter, the **owner** parameter, the **cacheReadPrefetch** parameter, the **segmentSize** parameter, and the **enclosureLossProtect** 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 logical drive. The logical drive name can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the logical drive name is 30 characters. You must enclose the logical drive name with double quotation marks (" ").

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

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

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

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

The **cacheReadPrefetch** parameter and the **segmentSize** parameter are the same as those described for the autoConfigure storageSubsystem command.

The **enclosureLossProtect** parameter defines enclosure loss protection for the array. (For a description of how enclosure loss protection works, see "Enclosure Loss Protection" on page 156.)

Example of Creating Logical Drives with User-Assigned Disk Drives

```
c:\...\client>smcli 123.45.67.89 -c "create logicalDrive drives =(1,1 1,2 1,3 2,1 2,2 2,3) raidLevel=5
userLabel=\"Engineering_1\" capacity=20GB owner=a cacheReadPrefetch=TRUE segmentSize=128;"
```

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

18 GB x 6 disk drives = 108 GB

Because only 20 GB is assigned to the logical drive, 88 GB remains available (as unconfigured capacity) for other logical drives that a user can add to this array later.

108 GB - 20 GB subsystem 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 logical drive is 128 KB. Enclosure loss protection is set to TRUE, which prevents any operations to disk drives in the expansion drawer if the expansion drawer fails. Hot spares are not created for this new array. You must create hot spares after you run this command.

Creating Logical Drives with Software-Assigned Disk Drives

If you choose to let the Storage Manager software assign the disk drives when you create the logical drive, you need only to specify the number of disk drives that you want to use. The Storage Manager software then assigns the disk drives. The controller firmware assigns a array number to the new array. To manually create arrays and logical drives, use the create logicalDrive command:

```
create logicalDrive driveCount=numberOfDrives
raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=logicalDriveName
[driveType=(fibre | SATA | SAS)]
[capacity=logicalDriveCapacity | owner=(a | b) |
cacheReadPrefetch=(TRUE | FALSE) |
segmentSize=segmentSizeValue]
[enclosureLossProtect=(TRUE | FALSE)]
```

This command is similar to the previous create logicalDrive command in which users assign the disk 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 disk drives you want to use in the array. You do not need to enter a list of disk drives. All of the other parameters are the same. Enclosure loss protection is performed differently when the Storage Manager software assigns the disk drives than when a user assigns the disk drives. (For a description of the difference, see “Enclosure Loss Protection” on page 156)

Example of Creating Logical Drives with Software Assigned Disk Drives

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

The command in this example creates the same logical drive as the example for the previous create logicalDrive command in which a user assigns the disk drives. The difference is that a user does not know which disk drives are assigned to this array.

Creating Logical Drives in an Existing Array

If you want to add a new logical drive to an existing array, use this command:

```
create logicalDrive array=arrayNumber
userLabel=logicalDriveName
[freeCapacityArea=freeCapacityIndexNumber |
capacity=logicalDriveCapacity | 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 **array** parameter is the number of the array in which you want to create a new logical drive. If you do not know the array numbers on the storage subsystem, you can use the show allLogicalDrives summary command to get a list of the logical drives and the arrays to which the logical drives belong.

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

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

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 logicalDrive` command.

Enclosure Loss Protection

The **enclosureLossProtect** parameter is a boolean switch that you set to turn on or turn off enclosure loss protection. For enclosure loss protection to work, each disk drive in a array must be on a separate enclosure. The way in which enclosure loss protection works depends on the method that you choose to assign the disk drives for a array.

When you assign the disk drives, if you set **enclosureLossProtect=TRUE** and have selected more than one disk drive from any one enclosure, the storage subsystem returns an error. If you set **enclosureLossProtect=FALSE**, the storage subsystem performs operations, but the array that you create does not have enclosure loss protection.

When the controller firmware assigns the disk drives, if **enclosureLossProtect=TRUE** the storage subsystem posts an error if the controller firmware cannot provide disk drives that result in the new array having enclosure loss protection. If **enclosureLossProtect=FALSE**, the storage subsystem performs the operation even if it means that the array might not have enclosure loss protection.

Enclosure loss protection is not valid when creating logical drives on existing arrays.

Modifying Your Configuration

For most configurations, after you have created your initial configuration by using the `autoConfigure storageSubsystem` command or the `create logicalDrive` 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 subsystem configuration. This section describes how to modify these properties:

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

Setting the Controller Clocks

To synchronize the clocks on the controllers with the host, use the `set storageSubsystem 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:\...\...\client>smcli 123.45.67.89 -c "set storageSubsystem time;"
```

Setting the Storage Subsystem Password

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

```
set storageSubsystem password="password"
```

The **password** parameter defines a password for the storage subsystem. Passwords provide added security to a storage subsystem 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 subsystem, you can run all of the script commands. A password protects the storage subsystem from any command that the controllers consider destructive. A destructive command is any command that can change the state of the storage subsystem, such as logical drive creation; cache modification; or reset, delete, rename, or change commands.

If you have more than one storage subsystem in a storage configuration, each storage subsystem 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 storageSubsystem` command to define a password:

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

Important: Password cannot be reset without current password– If you forget the storage subsystem password, you cannot reset the password using Storage Manager. To reset the password in that situation, you must contact your IBM support representative.

Setting the Storage Subsystem Host Type

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

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

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

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 subsystem. The only time that you might need to use this parameter is if you need to change how the storage subsystem 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 subsystem. To return information about host types that are connected to the storage subsystem, use the `show storageSubsystem` 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 storageSubsystem` command:

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

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

This example shows how to define a specific default host type after displaying the host type table:

```
c:\...\...\client>smcli 123.45.67.89 -c "set storageSubsystem defaultHostType=xx;"
```

where *xx* is the host type index value. For value *xx*, refer to the previously-displayed host type table. From the table, select the appropriate value for the host server that the LUN from the storage subsystem is mapped to.

Setting the Storage Subsystem Cache

The cache is high-speed memory that holds data that is either written to the disk 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 disk drives but not yet transferred to the host. The write cache contains data from the host but not yet written to the disk drives.

The cache acts as a buffer so that data transfers between the host and the disk 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 disk 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 disk drives.

Important: Changing default cache settings is not recommended – IBM recommends that you do not change the default cache setting values. Incorrect cache settings might severely affect the storage subsystem performance.

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

- set storageSubsystem
- set logicalDrive

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

```
set storageSubsystem 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 or 16KB. The value that you use applies to the entire storage subsystem and all of the logical drives in the storage subsystem. For redundant controller configurations, this value includes all logical drives 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:\...\...\client>smcli 123.45.67.89 -c "set storageSubsystem cacheBlockSize=16;"
```

To prevent data loss or corruption, the controller periodically writes cache data to the disk 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 disk drives when data has been in the cache for a predetermined amount of time. The controller writes data to the disk drives until the amount of data in the cache drops to a stop percentage level. Use the set storageSubsystem 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:\...\...\client>smcli 123.45.67.89 -c "set storageSubsystem 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 logicalDrive 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 logical drives or for a specific logical drive in an array. The command takes this form:

```

set (allLogicalDrives | logicalDrive [logicalDriveName] |
logicalDrives [logicalDriveName1 ... logicalDriveNameN]
logicalDrive <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 disk drives. The following table lists the values for the **cacheFlushModifier** parameter.

Table 17. 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 logical drives in the storage subsystem:

```
c:\...\...\client>smcli 123.45.67.89 -c "set alllogicalDrives 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 subsystem fails.

Note: Attention: Possible loss of data – Because you can lose data when the **cacheWithoutBatteryEnabled** parameter is enabled, IBM recommends that you do not enable this parameter in a production environment.

This example shows how to set this parameter value:

```
c:\...\...\client>smcli 123.45.67.89 -c "set logicalDrive  
[\"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.

Note: Attention: Possible loss of data – If the **mirrorCacheEnabled** parameter is disabled and one of the controllers fails, you might lose data that was cached in the failed controller memory but not yet written to disk. Therefore, IBM recommends that you do not disable this parameter in a production environment.

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:\...\...\client>smcli 123.45.67.89 -c "set logicalDrive  
[\"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 disk 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:\...\...\client>smcli 123.45.67.89 -c "set logicalDrive  
[\"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 logical drive data in the cache is automatically written to the disk 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:\...\...\client>smcli 123.45.67.89 -c "set allLogicalDrives  
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 disk 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 subsystem 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:\...\...\client>smcli 123.45.67.89 -c "set logicalDrive  
[\"Engineering_1\" \"Engineering_2\"] cacheReadPrefetch=TRUE;"
```

Setting the Modification Priority

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

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

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 logicalDrive` command to define the modification priority for a logical drive. The command takes this form:

```
set (allLogicalDrives | logicalDrive [logicalDriveName] |
logicalDrives [logicalDriveName1 ... logicalDriveNameN] logicalDrive <wwID> |
accessLogicalDrive)
modificationPriority=(highest | high | medium | low | lowest)
```

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

```
c:\...\...\client>smcli 123.45.67.89 -c "set logicalDrive
[\"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 disk drives by the enclosure 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 [enclosureID,slotID] | drives
[enclosureID1,slotID1 ... enclosureIDn,slotIDn]) hotSpare=(TRUE | FALSE)
```

This example shows how to set hot spare disk drives:

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

Enter the enclosure ID and the slot ID of each disk drive that you want to use. Enclose the list in square brackets, separate the enclosure ID value and the slot ID value of a disk drive with a comma, and separate each enclosure 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 subsystems, you can save the configuration to a file by using the `save storageSubsystem` configuration command. Saving the configuration creates a script file that you can run on the command line. The command takes this form:

```
save storageSubsystem configuration file="filename"
[(allconfig | globalSettings=(TRUE | FALSE)) |
logicalDriveConfigAndSettings=(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:\...\...\client>smcli 123.45.67.89 -c "save storageSubsystem
configuration file=\"c:\\folder\\storageSubsystemconfig1.scr\";"
```

In this example, the name `folder` is the folder in which you choose to place the profile file and `storageSubsystemconfig1.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 Storage Manager software uses this extension when it creates the configuration file.

Chapter 5. Using the FlashCopy Premium Feature

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

FlashCopy logical drives provide these capabilities:

- Create a complete image of the data on a base logical drive 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 FlashCopy logical drives to be read, written, and copied
- Use the same availability characteristics of the base logical drive (such as RAID protection and redundant path failover)
- Mapping the FlashCopy logical drive and making it accessible to any host on a storage area network (SAN). You can make FlashCopy data available to secondary hosts for read access and write access by mapping the FlashCopy to the hosts
- Creating up to four FlashCopies per logical drive (the maximum number of FlashCopy logical drives is one-half of the total number of logical drives that are supported by the controller)
- Increasing the capacity of a FlashCopy logical drive

How FlashCopy Works

Three components comprise a FlashCopy logical drive: the base logical drive, the FlashCopy logical drive, and the FlashCopy repository logical drive. The following table lists the components and briefly describes what they do.

Table 18. Components of a FlashCopy Logical Drive

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

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

Note: When you first create a FlashCopy repository logical drive, briefly stop all of the write operations to the base logical drive so that a stable image of the base logical drive is available.

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

You can read, write, and copy a FlashCopy logical drive. Data written by a host to the FlashCopy logical drive is handled in the FlashCopy repository logical drive. When a write occurs to the base logical drive of a FlashCopy logical drive, the new data also overwrites the appropriate FlashCopy repository logical drive data. The following table lists the FlashCopy logical drive commands.

Table 19. FlashCopy Logical Drive Commands

Command	Description
create flashcopyLogicalDrive	This command creates a FlashCopy logical drive.
recreate flashcopy	This command starts a fresh copy-on-write operation by using an existing FlashCopy logical drive.
recreate flashcopy collection	This command restarts multiple FlashCopy logical drives as one batch operation using one or many existing FlashCopy logical drives.
set (flashcopyLogicalDrive)	This command defines the properties for a FlashCopy logical drive and lets you rename a FlashCopy logical drive.
stop flashcopy	This command stops a copy-on-write operation.

Creating a FlashCopy Logical Drive

The create FlashCopyLogicalDrive command provides three methods for defining the disk drives for your FlashCopy repository logical drive:

- Defining the disk drives for the FlashCopy repository logical drive by their enclosure IDs and their slot IDs
- Defining an array in which the FlashCopy repository logical drive resides. In addition, you can define the capacity of the FlashCopy repository logical drive.
- Defining the number of disk drives, but not specific disk drives, for the FlashCopy repository logical drive.

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

Creating a FlashCopy Logical Drive with User-Assigned Disk Drives

Creating a FlashCopy logical drive by assigning the disk drives provides flexibility in defining your configuration by letting you choose from the available disk drives in your storage subsystem. When you

choose the disk drives for your FlashCopy logical drive, you automatically create a new array. You can specify which disk drives to use and the RAID level for the new array. The command takes this form:

```
create flashcopyLogicalDrive baseLogicalDrive="baseLogicalDriveName"
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
(repositoryDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
userLabel="flashcopyLogicalDriveName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failFlashCopy)]
[enclosureLossProtect=(TRUE | FALSE)]
```

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

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

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

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

```
create flashcopyLogicalDrive baseLogicalDrive="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:\...\...\client>smcli 123.45.67.89 -c "create flashcopyLogicalDrive baseLogicalDrive=\"Mars_Spirit_4\";"
```

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

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

```
create flashcopyLogicalDrive baseLogicalDrive="Mars_Spirit_4";
```

Creating a FlashCopy Logical Drive with Software-Assigned Disk Drives

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

```
create FlashCopyLogicalDrive baseLogicalDrive="baseLogicalDriveName"
[repositoryArray=ArrayNumber
freeCapacityArea=freeCapacityIndexNumber
userLabel="FlashCopyLogicalDriveName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel=repositoryName
repositoryFullPolicy=(failBaseWrites | failFlashCopy)]
[enclosureLossProtect=(TRUE | FALSE)]
```

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

```
c:\...\...\client>smcli 123.45.67.89 -c "create
FlashCopyLogicalDrive baseLogicalDrive=\"Mars_Spirit_4\"
repositoryArray=2 freeCapacityArea=2;"
```

The command in this example creates a new FlashCopy repository logical drive in array 2. The base logical drive is Mars_Spirit_4. The size of the FlashCopy repository logical drive is 4 GB. This command also takes a FlashCopy of the base logical drive, starting the copy-on-write operation.

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

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

```
create FlashCopyLogicalDrive baseLogicalDrive="Mars_Spirit_4"
repositoryArray=2 freeCapacityArea=2;
```

Creating a FlashCopy Logical Drive by Specifying a Number of Disk Drives

With this version of the create flashcopyLogicalDrive command, you must specify the number of disk drives and the RAID level that you want for the FlashCopy repository logical drive. This version of the create flashcopyLogicalDrive command creates a new array. You must have disk drives in the storage subsystem that are not assigned to an array for this command to work.

```
create flashcopyLogicalDrive baseLogicalDrive="baseLogicalDriveName"
[repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDriveCount=numberOfDrives
driveType=(fibre | SATA | SAS)
userLabel="flashcopyLogicalDriveName"
warningThresholdPercent=percentValue
repositoryPercentOfBase=percentValue
repositoryUserLabel="repositoryName"
repositoryFullPolicy=(failBaseWrites | failFlashCopy)]
[enclosureLossProtect=(TRUE | FALSE)]
```

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

```
c:\...\...\client>smcli 123.45.67.89 -c "create flashcopyLogicalDrive
baseLogicalDrive=\"Mars_Spirit_4\" repositoryRAIDLevel=5 repositoryDriveCount=3;"
```

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

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

```
create flashcopyLogicalDrive baseLogicalDrive= "Mars_Spirit_4"
repositoryRAIDLevel=5 repositoryDriveCount=3;
```

User-Defined Parameters

Use the parameters in the create flashcopyLogicalDrive command to define the FlashCopy logical drive to suit the requirements of your storage subsystem. The following table lists the parameters and briefly describes what the parameters do.

Table 20. FlashCopy Logical Drive Parameters

Parameter	Description
driveType	The type of disk drive that you want to use for the FlashCopy repository logical drive. Valid disk drive types are fibre, SATA, or SAS. This parameter works only with the count-based repository method of defining a FlashCopy logical drive.
repositoryArray	The array in which you want to build the FlashCopy repository logical drive. The default value is to build the FlashCopy repository logical drive in the same array as the base logical drive.
freeCapacityArea	The amount of storage space that you want to use for the FlashCopy repository logical drive. Free storage space is defined in units of bytes, KB, MB, GB, or TB.
userLabel	The name that you want to give to the FlashCopy logical drive. If you do not choose a name for the FlashCopy logical drive, the software creates a default name by using the base logical drive name. For example, with a base logical drive name of Mars_Spirit_4: <ul style="list-style-type: none"> • When the base logical drive does not have a FlashCopy logical drive, the default FlashCopy logical drive name is Mars_Spirit_4-1. • When the base logical drive already has <i>n</i>-1 number of FlashCopy logical drives, the default name is Mars_Spirit_4-<i>n</i>.
repositoryUserLabel	The name that you want to give to the FlashCopy repository logical drive. If you do not choose a name for the FlashCopy repository logical drive, the software creates a default name by using the base logical drive name. For example, if the base logical drive name is Mars_Spirit_4 and does not have an associated FlashCopy repository logical drive, the default FlashCopy repository logical drive name is Mars_Spirit_4-R1. If the base logical drive already has <i>n</i> -1 number of flashcopy repository logical drives, the default name is Mars_Spirit_4-R <i>n</i> .
warningThresholdPercent	The percentage of the capacity that you will permit the FlashCopy repository logical drive to get before you receive a warning that the FlashCopy repository logical drive is nearing full. The warning value is a percentage of the total capacity of the FlashCopy repository logical drive. The default value is 50, which represents 50 percent of the total capacity. (You can change this value later by using the set flashcopyLogicalDrive command.)
repositoryPercentOfBase	The size of the FlashCopy repository logical drive as a percentage of the base logical drive size. The default value is 20, which represents 20 percent of the base logical drive size.
repositoryFullPolicy	The type of FlashCopy processing that you want to continue if the FlashCopy repository logical drive is full. You can choose to fail writes to the base logical drive (failBaseWrites) or fail writes to the FlashCopy logical drive (failFlashCopy). The default value is failFlashCopy.

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

```
c:\...\...\client>smcli 123.45.67.89 -c "create
flashcopyLogicalDrive baseLogicalDrive=\"Mars_Spirit_4\"
repositoryRAIDLevel=5 repositoryDriveCount=5
driveType=fibre userLabel=\"Mars_Spirit_4_snap1\"
repositoryUserLabel=\"Mars_Spirit_4_rep1\"
warningThresholdPercent=75 repositoryPercentOfBase=40
repositoryFullPolicy=failSnapshot;"
```

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

```
create flashcopyLogicalDrive baseLogicalDrive="Mars_Spirit_4"
repositoryRAIDLevel=5 repositoryDriveCount=5 driveType=fibre
userLabel="Mars_Spirit_4_snap1"
repositoryUserLabel="Mars_Spirit_4_rep1"
warningThresholdPercent=75 repositoryPercentOfBase=40
repositoryFullPolicy=failSnapshot;
```

FlashCopy Logical Drive Names and FlashCopy Repository Logical Drive Names

The FlashCopy logical drive names and the FlashCopy repository logical drive names can be any combination of alphanumeric characters, hyphens, and underscores. The maximum length of the logical drive names is 30 characters. You must enclose the logical drive 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 FlashCopy logical drive and the FlashCopy repository logical drive is to add a hyphenated suffix to the original base logical drive name. The suffix distinguishes between the FlashCopy logical drive and the FlashCopy repository logical drive. For example, if you have a base logical drive with a name of Engineering Data, the FlashCopy logical drive can have a name of Engineering Data-S1, and the FlashCopy repository logical drive can have a name of Engineering Data-R1.

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

- If the base logical drive name is *aaa* and does not have a FlashCopy logical drive, the default FlashCopy logical drive name is *aaa1*.
- If the base logical drive already has *n-1* number of FlashCopy logical drives, the default name is *aaa-n*.
- If the base logical drive name is *aaa* and does not have a FlashCopy repository logical drive, the default FlashCopy repository logical drive name is *aaa-R1*
- If the base logical drive already has *n-1* number of FlashCopy repository logical drives, the default name is *aaa-Rn*.

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

Changing FlashCopy Logical Drive Settings

Use the `set (FlashCopy) logicalDrive` command to change these property settings for a FlashCopy logical drive:

- The FlashCopy logical drive name
- The warning threshold percent
- The FlashCopy repository full policy

This example shows how to change a FlashCopy logical drive name.

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

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

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

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

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

Stopping, Restarting, and Deleting a FlashCopy Logical Drive

When you create a FlashCopy logical drive, copy-on-write starts running immediately. As long as a FlashCopy logical drive is enabled, storage subsystem performance is impacted by the copy-on-write operations to the associated FlashCopy repository logical drive.

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

```
c:\...\...\client>smcli 123.45.67.89 -c "stop flashcopy logicalDrives  
[\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];"
```

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

```
stop flashcopy logicalDrives [\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];
```

When you stop the copy-on-write operations for a specific FlashCopy logical drive, only that FlashCopy logical drive is disabled. All of the other FlashCopy logical drives stay in operation.

When you want to restart a copy-on-write operation, use the recreate flashcopy logicalDrive command or the recreate flashcopy collection command. The recreate flashcopy logicalDrive command starts a fresh copy-on-write operation by using an existing FlashCopy logical drive.

Note: The FlashCopy logical drive must be in either an Optimal state or a Disabled state.

When you restart a FlashCopy logical drive, these actions occur:

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

This example shows how to restart a FlashCopy logical drive:

```
c:\...\...\client>smcli 123.45.67.89 -c "recreate flashcopy  
logicalDrives [\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];"
```

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

```
recreate flashcopy logicalDrives [\"Mars_Spirit_4-2\" \"Mars_Spirit_4-3\"];
```

If you do not intend to use a FlashCopy logical drive again, you can delete the FlashCopy logical drive by using the delete logicalDrive command. When you delete a FlashCopy logical drive, the associated FlashCopy repository logical drive also is deleted.

Chapter 6. Using the Enhanced Remote Mirroring Premium Feature

The Enhanced Remote Mirroring premium feature provides for online, real-time replication of data between storage subsystems over a remote distance. In the event of a disaster or a catastrophic failure on one storage subsystem, you can promote the second storage subsystem to take over responsibility for computing services. Enhanced Remote Mirroring is designed for extended storage environments in which the storage subsystems that are used for Enhanced Remote Mirroring are maintained at separate sites. Logical drives on one storage subsystem are mirrored to logical drives on another storage subsystem 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 subsystems. Because Enhanced Remote Mirroring is storage based, it does not require any server overhead or application overhead.

You can use Enhanced Remote Mirroring for these functions:

- **Disaster recovery** – Enhanced Remote 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** – Enhanced Remote 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** – Enhanced Remote Mirroring provides the ability to have two storage subsystems back up each other by mirroring critical logical drives on each storage subsystem to logical drives on the other storage subsystem. This action lets each storage subsystem recover data from the other storage subsystem in the event of any service interruptions.

How Enhanced Remote Mirroring Works

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

Note: Enhanced Remote Mirroring is not supported in DS3000 storage subsystem configurations.

Table 21. Maximum Number of Defined Mirrors

Controller Model	Maximum Number of Defined Mirrors
DS3200, DS3300, DS3400, DS3500	Supported ONLY in a co-existence environment with DS4000, or DS5000 storage subsystems
DS4300	32
DS4700	64
DS4400	64
DS4500	64
DS4800	128
DS5100	128
DS5100	128

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

The controller owner of the primary logical drive initiates remote writes to the secondary logical drive to keep the data on the two logical drives synchronized.

The secondary logical drive maintains a mirror (or copy) of the data on its associated primary logical drive. The controller owner of the secondary logical drive receives remote writes from the controller owner of the primary logical drive but will not accept host write requests. Hosts are able to read from the secondary logical drive, 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 logical drive to a primary role. Hosts then are able to read from and write to the newly promoted logical drive, and business operations can continue.

Mirror Repository Logical Drives

A mirror repository logical drive is a special logical drive in the storage subsystem that is created as a resource for the controller owner of the primary logical drive in a remote-mirror pair. The controller stores mirroring information on this logical drive, 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 subsystems.

When you activate the Enhanced Remote Mirroring premium feature on the storage subsystem, you create two mirror repository logical drives, one for each controller in the storage subsystem. An individual mirror repository logical drive is not needed for each remote mirror.

When you create the mirror repository logical drives, you specify the location of the logical drives. You can either use existing free capacity, or you can create an array for the logical drives 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 logical drives. The required size of each logical drive is 128 MB, or 256 MB total for both mirror repository logical drives of a dual-controller storage subsystem. In previous versions of the Enhanced Remote Mirroring feature, the mirror repository logical drives 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 Enhanced Remote Mirroring premium feature on both the primary storage subsystem and the secondary storage subsystem. You must also create a secondary logical drive on the secondary site if one does not already exist. The secondary logical drive must be a standard logical drive of equal or greater capacity than the associated primary logical drive.

When secondary logical drives are available, you can establish a mirror relationship in the storage management software by identifying the primary logical drive and the storage subsystem that contains the secondary logical drive.

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

Data Replication

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

Write Modes

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

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

Metro Mirror

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

Global Copy

Global copy 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 Global copy, host write requests are written to the primary logical drive. 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 subsystem.

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

Global Mirror

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

Link Interruptions or Secondary Logical Drive Errors

When processing write requests, the primary controller might be able to write to the primary logical drive, 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 logical drive, and the primary logical drive and the secondary logical drive 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 logical drive, but remote writes do not take place.

When communication is restored between the controller owner of the primary logical drive and the controller owner of the secondary logical drive, 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 logical drive during the link interruption are copied to the secondary logical drive. 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 logical drive error on the secondary side prevents the remote write from completing. For example, an offline secondary logical drive or a failed secondary logical drive can cause the remote mirror to become unsynchronized. When the logical drive error is corrected (the secondary logical drive 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 logical drive and the secondary logical drive in a mirror relationship is managed by the controllers and is transparent to host machines and applications. When the controller owner of the primary logical drive receives a write request from a host, the controller first logs information about the write to a mirror repository logical drive. The controller then writes the data to the primary logical drive. The controller then initiates a write operation to copy the affected data to the secondary logical drive on the remote storage subsystem.

If a link interruption or a logical drive error prevents communication with the secondary storage subsystem, the controller owner of the primary logical drive 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 logical drive, but remote writes to the secondary logical drive do not take place.

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

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

Creating a Remote-Mirror Pair

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

- The secondary logical drive must be of equal or greater size than the primary logical drive.

- The RAID level of the secondary logical drive does not have to be the same as the primary logical drive.

Use these steps to create the logical drive.

1. Enable the Enhanced Remote Mirroring feature.
2. Activate the Enhanced Remote 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 logical drive performs a full synchronization in the background while processing local I/O writes to the primary logical drive and associated remote writes to the secondary logical drive. 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 logical drive 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 logical drive to the secondary logical drive 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 Enhanced Remote Mirroring Feature

The first step in creating a remote mirror is to make sure that the Enhanced Remote Mirroring feature is enabled on both storage subsystems. Because Enhanced Remote 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 storageSubsystem 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 Enhanced Remote Mirroring Feature

Activating the Enhanced Remote Mirroring feature prepares the storage subsystems 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 logical drive is automatically created for each controller in the storage subsystem. As part of the activation process, you

can decide where the mirror repository logical drives will reside, free capacity on an existing array or in a newly created array, and the RAID level for the mirror repository logical drives.

The free capacity that you select for the mirror repository logical drive must have a total of 256 MB of capacity available. Two mirror repository logical drives 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 logical drives, the firmware returns an error message that gives the amount of space needed for the mirror repository logical drives. The command does not try to activate the Enhanced Remote 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 logical drive 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 disk drives in the array.
- **RAID 6** – You must have a minimum of five disk drives in the array.

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

```
activate storageSubsystem feature=remoteMirror
```

The activate storageSubsystem feature=remoteMirror command provides three methods for defining the disk drives for your mirror repository logical drive:

- You define each disk drive for the mirror repository logical drive by its enclosure ID and its slot ID.
- You define an array in which the mirror repository logical drive resides. You can optionally define the capacity of the mirror repository logical drive.
- You define the number of disk drives, but not specific disk drives, for the mirror repository logical drive.

Activating the Enhanced Remote Mirroring Feature with User-Assigned Disk Drives

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

The command takes this form:

```
activate storageSubsystem feature=remoteMirror
repositoryRAIDLevel=(1 | 3 | 5 | 6)
repositoryDrives=(enclosureID1,slotID1 ... enclosureIDn,slotIDn)
enclosureLossProtect=(TRUE | FALSE)
```

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

```
c:\...\client>smcli 123.45.67.89 -c "activate
storageSubsystem 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 logical drive consisting of five disk drives that forms a new array. The new array has a RAID level of 5.

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

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

Activating the Enhanced Remote Mirroring Feature with Software-Assigned Disk Drives

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

The command takes this form:

```
activate storageSubsystem feature=remoteMirror
repositoryArray=arrayNumber
[freeCapacityArea=freeCapacityIndexNumber]
```

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

```
c:\...\...\client>smcli
123.45.67.89 -c "activate storageSubsystem feature=remoteMirror repositoryArray=2
freeCapacityArea=2;"
```

The command in this example creates a new mirror repository logical drive in array 2 using the second free capacity area.

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

```
activate storageSubsystem feature=remoteMirror repositoryArray=2 freeCapacityArea=2;
```

Activating the Enhanced Remote Mirroring Feature by Specifying a Number of Disk Drives

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

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

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

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

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

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

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

Determining Candidates for a Remote-Mirror Pair

All of the logical drives and disk drives on the remote storage subsystem might not be available for use as secondary logical drives. To determine which logical drives on a remote storage subsystem that you

can use as candidates for secondary logical drives, use the show remoteMirror candidates command. This command returns a list of the logical drives that you can use when creating a remote mirror.

The command takes this form:

```
c:\...\...\client>smcli 123.45.67.89 -c "show
remoteMirror candidates primary=\"logicalDriveName\"
remoteStorageSubsystemName=\"storageSubsystemName\";"
```

where *logicalDriveName* is the name of the logical drive that you want to use for the primary logical drive, and *StorageSubsystemName* is the remote storage subsystem that contains possible candidates for the secondary logical drive. Enclose both the logical drive name and the storage subsystem name in double quotation marks (" ").

Creating a Remote-Mirror Pair

When you create a new remote mirror, you must define which logical drives that you want to use for the primary (local) logical drive and the secondary (remote) logical drive. You define the primary logical drive by the name of the logical drive. You define the secondary logical drive by name with either the name or the World-Wide Identifier (WWID) of the storage subsystem on which the secondary logical drive resides. The primary logical drive name, the secondary logical drive name, and the remote storage subsystem 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="primaryLogicalDriveName"
secondary="secondaryLogicalDriveName"
(remoteStorageSubsystemName="storageSubsystemName" |
remoteStorageSubsystemWwn="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:\...\...\client>smcli 123.45.67.89 -c "create
remoteMirror primary=\"Jan_04_Account\"
secondary=\"Jan_04_Account_B\" remoteStorageSubsystemName=\"Tabor\"
remotePassword=\"jdw2ga05\" syncPriority=highest
writeMode=synchronous;"
```

The command in this example creates a remote mirror in which the primary logical drive is named Jan_04_Account on the local storage subsystem. The secondary logical drive is named Jan_04_Account_B on the remote storage subsystem that is named Tabor. The names used in this example are similar, but that is not a requirement for the logical drive names in a remote-mirror pair. In this example, the remote storage subsystem has a password that you must enter when making any change to the storage subsystem configuration. Creating a remote-mirror pair is a significant change to a storage subsystem configuration. Setting the write mode to synchronous and the synchronization priority to highest means that host write requests are written to the primary logical drive and then immediately copied to the secondary logical drive. These actions help to make sure that the data on the secondary logical drive is as accurate a copy of the data on the primary logical drive 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" remoteStorageSubsystemName="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 logical drive and the secondary logical drive by running the `show remoteMirror synchronizationProgress` command. This command shows the progress as a percentage of data synchronization that has completed.

Changing Enhanced Remote 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 logical drive 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 logical drive 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:\...\...\client>smcli 123.45.67.89 -c "set remoteMirror
localLogicalDrive [Jan_04_Account] syncPriority=medium writeOrder=notpreserved
writeMode=asynchronous;"
```

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

```
set remoteMirror localLogicalDrive [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 logical drive and a secondary logical drive in a mirror relationship without disabling the mirror relationship. Suspending a mirror relationship lets you control when the data on the primary logical drive and data on the secondary logical drive 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 logical drive is copied to the secondary logical drive. Suspending a mirror relationship is particularly useful when you want to run a backup of the data on the secondary logical drive.

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

Important: If you suspend a remote mirror that is set up in the Global mirror, 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:\...\...\client>smcli 123.45.67.89 -c "suspend remoteMirror primary Jan_04_Account
writeConsistency=false;"
```

The **writeConsistency** parameter defines whether the logical drives identified in this command are in a write-consistency group or are separate. For the logical drives in a write-consistency group, set this parameter to TRUE. For the logical drives 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 logicalDrive 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 logical drive and a secondary logical drive in a mirror relationship after the mirror has been suspended or unsynchronized.

This example shows the resume remoteMirror command:

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

Removing a Mirror Relationship

Use the remove remoteMirror command to remove the link between a primary logical drive and a secondary logical drive. (Removing a mirror relationship is similar to deleting a mirror relationship.) Removing the link between a primary logical drive and a secondary logical drive does not affect any of the existing data on either logical drive. The link between the logical drives is removed, but the primary logical drive still continues normal I/O operations. Later, you can establish the mirror relationship between the two logical drives 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:\...\...\client>smcli 123.45.67.89 -c "remove remoteMirror localLogicalDrive [Jan_04_Account];"
```

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

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

```
remove remoteMirror localLogicalDrive [Jan_04_Account];
```

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

Deleting a Primary Logical Drive or a Secondary Logical Drive

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

Attention: Possible loss of data access – Deleting a primary logical drive or a secondary logical drive permanently removes the data from the storage subsystem.

Disabling the Enhanced Remote Mirroring Feature

You disable the Enhanced Remote Mirroring feature to prevent the new mirror relationship from being created. When you disable the Enhanced Remote 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 Enhanced Remote Mirroring feature, use this command:

```
disable storageSubsystem feature=remoteMirror
```

Deactivating the Enhanced Remote Mirroring Feature

If you no longer require the Enhanced Remote Mirroring feature and you have removed all of the mirror relationships, you can deactivate the feature. Deactivating the feature reestablishes the normal use of dedicated ports on both storage subsystems and deletes both mirror repository logical drives. To deactivate the Enhanced Remote Mirroring feature, use this command:

```
deactivate storageSubsystem feature=remoteMirror
```

Interaction with Other Features

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

- Storage Partitioning
- FlashCopy Logical Drive
- VolumeCopy

When you run the Enhanced Remote 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 subsystem configuration.

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

Storage Partitioning

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

- A host
- A host group
- A logical drive-to-LUN mapping

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

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

FlashCopy Logical Drives

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

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

In this version of the storage management software, you can create FlashCopy logical drives based on the primary logical drive or secondary logical drive of a remote mirror.

VolumeCopy

The VolumeCopy premium feature copies data from one logical drive (the source logical drive) to another logical drive (the target logical drive) within a single storage subsystem. You can use this feature to perform these functions:

- Copy data from arrays that use smaller-capacity disk drives to arrays that use larger-capacity disk drives
- Back up data
- Restore FlashCopy logical drive data to the base logical drive.

You can use a primary logical drive in a remote mirror as a source logical drive or a target logical drive in a VolumeCopy. You cannot use a secondary logical drive as a source logical drive or a target logical drive.

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

Dynamic Logical Drive Expansion

A Dynamic Logical Drive Expansion (DVE) is a modification operation that increases the capacity of a standard logical drive or a FlashCopy repository logical drive. The increase in capacity is achieved by using the free capacity that is available in the array of the standard logical drive or the FlashCopy repository logical drive.

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

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

Note: Although the storage management software indicates that the logical drive has increased capacity, its usable capacity is the size of the smaller of the primary logical drive or the secondary logical drive.

You cannot perform a DVE operation on a mirror repository logical drive.

Asynchronous Enhanced Remote Mirroring Utility

This section describes the host utility to achieve periodic consistency with Asynchronous Enhanced Remote Mirroring configurations. This section also describes how to run the Asynchronous Enhanced Remote Mirroring utility.

Description of the Asynchronous Enhanced Remote Mirroring Utility

The Asynchronous Enhanced Remote Mirroring utility lets you periodically synchronize the Enhanced Remote Mirroring pairs in your storage subsystem. When defining a Enhanced Remote Mirroring configuration, you have the option to set the write modes to either Synchronous or Asynchronous. Metro mirror provides the highest level security for full data recovery from the secondary storage subsystem in the event of a disaster. Metro mirror does, however, reduce host I/O performance. Global copy 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 Global copy, you cannot make sure that a logical drive, or collection of logical drives, at a secondary site ever reach a consistent, recoverable state.

The Asynchronous Enhanced Remote Mirroring utility enables you to bring a collection of asynchronous remote logical drives 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 Enhanced Remote 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 DS5000 Storage Manager 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 logical drives and mirrors.

Note: The Asynchronous Enhanced Remote Mirroring utility does not check to make sure that concurrently running instances of the utility are not trying to process the same logical drives and mirrors. If you choose to simultaneously run more than one instance of the Asynchronous Enhanced Remote Mirroring utility, you must make sure that the configuration files that you choose to run do not list the same logical drives and mirrors.

Operation of the Asynchronous Enhanced Remote Mirroring Utility

The Asynchronous Enhanced Remote Mirroring utility performs steps that generate a recoverable state for multiple mirror logical drives at a secondary site. The utility runs these steps to create consistent, recoverable images of a set of logical drives:

1. **On the primary storage subsystem** – The utility reconfigures all of the participating logical drives 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 subsystem** – The utility polls all of the participating logical drives 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 logical drives are cleared, and the secondary logical drives are recoverable.
3. **On the primary storage subsystem** – The utility suspends the mirrored pairs for all of the participating logical drives. This action causes updates to stop on the secondary side, leaving the secondary logical drives 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 logical drives 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 subsystem** – The utility generates a flashcopy of each participating logical drive on the secondary side, which creates point-in-time images that are recoverable.
5. **On the primary storage subsystem** – The utility resumes the mirroring operations for all of the participating logical drives. 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 subsystem** – The utility reconfigures all of the affected logical drives for Asynchronous mode.

Running the Asynchronous Enhanced Remote Mirroring Utility

The Asynchronous Enhanced Remote 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 Enhanced Remote Mirroring logical drives that you want to synchronize by using the utility. When you create the configuration file, use these conditions to define the logical drives in the file:

- All the primary logical drives in a logical drive set must belong to the same storage subsystem.
- The maximum number of logical drive 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 Enhanced Remote 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 Enhanced Remote 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 Enhanced Remote Mirroring utility, you must be managing the storage subsystem by using the command line interface, not the graphical user interface of DS Storage Manager.

Configuration Utility

The configuration file is an ASCII flat text file that provides the information for the Enhanced Remote Mirroring synchronization used by the Asynchronous Enhanced Remote Mirroring utility. The file defines the mirror logical drive sets to be synchronized. All of the mirror logical drives in the logical drive sets defined in the configuration file are run collectively to create a recoverable image. If any one of the mirrors in the logical drive set fails, the operation is stopped for this logical drive set and carried on to the next logical drive set that is listed in the configuration file.

The configuration file supports this syntax:

```
content ::= {spec}
spec ::= logSpec | logicalDriveSetSpec
logSpec ::= "Log" "{" {logAttribute} "}"
logAttribute ::= fileSpec
fileSpec ::= "file" "=" fileName
LogicalDriveSetSpec ::= "LogicalDriveSet" LogicalDriveSetName
"{" {LogicalDriveSetAttribute} "}"
LogicalDriveSetAttribute ::= timeoutSpec | mirrorSpec
timeoutSpec ::= "OptimalWaitTimeLimit" "=" integer
mirrorSpec ::= "Mirror" "{" {mirrorAttribute} "}"
mirrorAttribute ::= primarySpec | secondarySpec |
FlashCopySpec
```

```

primarySpec ::= "Primary" "=" LogicalDriveSpec
secondarySpec ::= "Secondary" "=" LogicalDriveSpec
FlashCopySpec ::= "Copy" "=" logicalDriveSpec
LogicalDriveSpec ::= storageSubsystemName"."logicalDriveUserLabel

```

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).
- *logicalDriveSetName* – The name of the set of logical drives on which you want to run the Asynchronous Enhanced Remote Mirroring utility.
- *fileName* – The name of a file, using characters and conventions that are appropriate for the system on which the application is running.
- *storageSubsystemName* – The label that you have assigned for a storage subsystem, as would be used in the CLI to specify the name of the storage subsystem.
- *logicalDriveUserLabel* – The label that you have assigned for a logical drive that uniquely identifies the logicalDrive within the storage subsystem.

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 **logicalDriveSetSpec** 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 **FlashCopySpec** attribute in a **mirrorSpec** command (you must include exactly one **FlashCopySpec** attribute in the **mirrorSpec** command)

Important: In the Asynchronous Enhanced Remote Mirroring utility configuration file, you must specify the primary logical drive, the secondary logical drive, and the copy (FlashCopy) logical drive. The utility does not make sure that the secondary logical drive is correct for the Enhanced Remote Mirroring relationship. The utility also does not make sure that the FlashCopy logical drive is actually a FlashCopy for the secondary logical drive. *You must make sure that these logical drives are correct.* If the logical drives are not correct, the utility will run, but the logical drives will not be consistent. For each mirror, the secondary logical drive and the copy logical drive must reside on the same storage subsystem.

This example shows a configuration file for the Asynchronous Enhanced Remote Mirroring utility.

```

Log{ file="d:\rvm-consistency.log" }
LogicalDriveSet "set1" {
  optimalWaitTimeLimit = 15
  Mirror {
    Primary = LosAngelesSubsystem.PayrollLogicalDrive
    Secondary = NewYorkSubsystem.PayrollLogicalDrive
    Copy = NewYorkSubsystem.PayrollLogicalDriveImage
  }
}

```

```
Mirror {  
  Primary = LosAngelesSubsystem.PayrollLogicalDrive  
  Secondary = BostonSubsystem.PayrollLogicalDrive  
  Copy = BostonSubsystem.PayrollLogicalDriveImage  
}  
}  
  
LogicalDriveSet "set2" {  
  Mirror {  
    Primary = BostonSubsystem.HRLogicalDrive  
    Secondary = LosAngelesSubsystem.HRLogicalDrive  
    Copy = LosAngelesSubsystem.HRLogicalDriveImage  
  }  
}
```

Chapter 7. Using the VolumeCopy Premium Feature

The VolumeCopy premium feature lets you copy data from one logical drive (the source) to another logical drive (the target) in a single storage subsystem. You can use this feature to perform these tasks:

- Back up data
- Copy data from arrays that use smaller-capacity disk drives to arrays using greater-capacity disk drives
- Restore FlashCopy logical drive data to the associated base logical drive

How VolumeCopy Works

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

Tip: Because of this restriction, IBM strongly recommends using a FlashCopy version of the logical drive as the source for the VolumeCopy, rather than using the original logical drive as the source. If you use the FlashCopy as the source for the VolumeCopy operation, then I/O operations to the original logical drive can continue.

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

Source Logical Drive

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

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

The following are valid source logical drives:

- A standard logical drive
- A FlashCopy logical drive
- The base logical drive of a FlashCopy logical drive
- A primary logical drive that is participating in a remote-mirror pair

The following are not valid source logical drives:

- A secondary logical drive that is participating in a remote-mirror pair
- A FlashCopy repository logical drive
- A mirror repository logical drive
- A failed logical drive

- A missing logical drive
- A logical drive currently in a modification operation
- A logical drive that is holding a Small Computer System Interface-2 (SCSI-2) reservation or a persistent reservation
- A logical drive that is a source logical drive or a target logical drive in another VolumeCopy that has a status of In Progress, Pending, or Failed

Target Logical Drive

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

Attention: Possible loss of data access – A VolumeCopy overwrites data on the target logical drive. 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 logical drive.

While the VolumeCopy has a status of In Progress, Pending, or Failed, the controllers reject read and write requests to the target logical drive. After the VolumeCopy operation is finished, the target logical drive automatically becomes read only to the hosts, and write requests to the target logical drive are rejected. You can change the Read-Only attribute after the VolumeCopy has completed or has been stopped. (For more information about the Read-Only attribute, see “Viewing VolumeCopy Properties” on page 192.)

The following logical drives are valid target logical drives:

- A standard logical drive
- The base logical drive of a disabled FlashCopy logical drive or failed FlashCopy logical drive
- A primary logical drive that is participating in a remote-mirror pair

The following logical drives are not valid target logical drives:

- The base logical drive of an active FlashCopy logical drive
- A FlashCopy logical drive
- A mirror repository logical drive
- A FlashCopy repository logical drive
- A secondary logical drive in a remote-mirror pair
- A failed logical drive
- A missing logical drive
- A logical drive with a status of Degraded
- A logical drive that is currently in a modification operation
- A logical drive that is holding a SCSI-2 reservation or a persistent reservation
- A logical drive that is a source logical drive or a target logical drive in another VolumeCopy that has a status of In Progress, Pending, or Failed

VolumeCopy and Persistent Reservations

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

To determine which logical drives have reservations, run the `show (logicalDrive) reservations` command. To remove a reservation, run the `clear (logicalDrive) reservations` command.

Storage Subsystem Performance

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

These factors contribute to the performance of the storage subsystem:

- The I/O activity
- The logical drive RAID level
- The logical drive configuration (number of disk drives in the arrays and cache parameters)
- The logical drive type (flashcopy logical drives might take more time to copy than standard logical drives)

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

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

Restrictions

These restrictions apply to the source logical drive, the target logical drive, and the storage subsystem:

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

VolumeCopy Commands

The following table lists the VolumeCopy commands and briefly describes what the commands do.

Table 22. VolumeCopy Commands

Command	Description
<code>create volumeCopy</code>	Creates a VolumeCopy and starts the VolumeCopy operation.
<code>disable storageSubsystem feature volumeCopy</code>	Turns off the current VolumeCopy operation.
<code>enable storageSubsystem feature</code>	Activates the VolumeCopy premium feature.

Table 22. VolumeCopy Commands (continued)

Command	Description
recovery volumeCopy	Re-initiates a VolumeCopy operation using an existing VolumeCopy pair.
remove volumeCopy	Removes a VolumeCopy pair.
set volumeCopy	Defines the properties for a VolumeCopy pair.
show volumeCopy	Returns information about VolumeCopy operations. You can retrieve information about a specific VolumeCopy pair, or all of the VolumeCopy pairs in the storage subsystem.
show volumeCopy sourceCandidates	Returns information about the candidate logical drives that you can use as the source for a VolumeCopy operation.
show volumeCopy targetCandidates	Returns information about the candidate logical drives that you can use as the target for a VolumeCopy operation.
stop volumeCopy	Stops a VolumeCopy operation.

Creating a VolumeCopy

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

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

To create a VolumeCopy, perform these general steps:

1. Enable the VolumeCopy premium feature.
2. Determine the candidates for a VolumeCopy.
3. Create the target logical drive and the source logical drive for the VolumeCopy.

Enabling the VolumeCopy Feature

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

```
enable storageSubsystem 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 VolumeCopy Candidates

All logical drives and disk drives might not be available for use in VolumeCopy operations. To determine which candidate logical drives on the storage subsystem that you can use as a source logical drive, use the show volumeCopy sourceCandidates command. To determine which candidate logical drives on the storage subsystem that you can use as a target logical drive, use the show volumeCopy targetCandidates command. These commands return a list of the expansion drawer, slot, and capacity information for the source logical drive candidates and the target logical drive candidates. You can use the show

volumeCopy sourceCandidates command and the show volumeCopy targetCandidates command only after you have enabled the VolumeCopy premium feature.

Creating a VolumeCopy

Attention: Possible loss of data access – A VolumeCopy overwrites data on the target logical drive. Make sure that you no longer need the data or have backed up the data on the target logical drive before you start a VolumeCopy operation.

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

The command takes this form:

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

Important: Setting the targetReadOnlyEnabled attribute to false breaks the one-to-one relationship between the source and target VolumeCopy logical drives. Therefore, any data that are written to the target VolumeCopy logical drive will not be copied to the source VolumeCopy logical drive. In addition, these data will be lost if you recopy of the source VolumeCopy logical drive to the target VolumeCopy logical drive.

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

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

This example shows the create volumeCopy command:

```
c:\...\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 logical drive named Jaba_Hut to the target logical drive named Obi_1. Setting the copy priority to medium provides a compromise between how quickly the data is copied from the source logical drive to the target logical drive and the amount of processing resources that are required for data transfers to other logical drives in the storage subsystem. Setting the **targetReadOnlyEnabled** parameter to TRUE means that write requests cannot be made to the target logical drive, making sure that the data on the target logical drive 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 VolumeCopy operation is completed, the target logical drive automatically becomes read only to hosts. Any write requests to the target logical drive are rejected, unless you disable the Read-Only attribute by using the set volumeCopy command.

To view the progress of a VolumeCopy, use the show logicalDrive actionProgress command. This command returns information about the logical drive action, the percentage completed, and the time remaining until the VolumeCopy is complete.

Viewing VolumeCopy Properties

Use the `show volumeCopy` command to view information about one or more selected source logical drives or target logical drives. 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 logical drive
- The source logical drive World-Wide Identifier (WWID) or the target logical drive WWID

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

The command takes this form:

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

This example shows the `show volumeCopy` command:

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

The command in this example is requesting information about the source logical drive `JabaHut`. If you wanted information about all of the logical drives, you would use the **allLogicalDrives** parameter. You can also request information about a specific target logical drive.

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

```
show volumeCopy source ["JabaHut"];
```

Changing VolumeCopy Settings

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

- The copy priority
- The target logical drive read/write permission

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

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

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

- Using the target logical drive for backup purposes
- Copying data from one array to a larger array for greater accessibility

- Planning to use the data on the target logical drive to copy back to the base logical drive in case of a disabled FlashCopy logical drive or failed FlashCopy logical drive

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

Note: If you have set the `VolumeCopy` parameters to enable host writes to the target logical drive, the read request and the write request to the target logical drive are rejected while the `VolumeCopy` 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)
```

Important: Setting the `targetReadOnlyEnabled` attribute to `false` breaks the one-to-one relationship between the source and target `VolumeCopy` logical drives. Therefore, any data that are written to the target `VolumeCopy` logical drive will not be copied to the source `VolumeCopy` logical drive. In addition, these data will be lost if you recopy of the source `VolumeCopy` logical drive to the target `VolumeCopy` logical drive.

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

This example shows the `set volumeCopy` command:

```
c:\...\...\client>smcli 123.45.67.89 -c "set volumeCopy
target [\"0bi_1\"] copyPriority=highest
targetReadOnlyEnabled=FALSE;"
```

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

```
set volumeCopy target ["0bi_1"] copyPriority=highest targetReadOnlyEnabled=FALSE;
```

Recopying a Logical Drive

Use the `recopy volumeCopy` command to create a new `VolumeCopy` 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 logical drive. 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 logical drive while the recopy operation is running. The recopy operation might take a long time.

Tip: Because of this restriction, IBM strongly recommends using a `FlashCopy` version of the logical drive as the source for the `VolumeCopy`, rather than using the original logical drive as the source. If you use the `FlashCopy` as the source for the `VolumeCopy` operation, then I/O operations to the original logical drive can continue.

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

Attention: Possible loss of data access – The `recopy volumeCopy` command overwrites existing data on the target logical drive and makes the target logical drive read-only to hosts. The `recopy volumeCopy` command fails all of the `flashcopy` logical drives that are associated with the target logical drive, 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 subsystem resources to the `VolumeCopy` at the expense of storage subsystem 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:\...\...\client>smcli 123.45.67.89 -c "recopy volumeCopy target [\"Obi_1\"] copyPriority=highest;"
```

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

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

The command takes this form:

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

This example shows the show volumeCopy command:

```
c:\...\...\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 VolumeCopy pair from the storage subsystem configuration. All of the VolumeCopy-related information for the source logical drive and the target logical drive is removed from the storage subsystem configuration. The data on the source logical drive or the target logical drive is not deleted. Removing a VolumeCopy from the storage subsystem configuration also removes the Read-Only attribute for the target logical drive.

Important: If the VolumeCopy has a status of In Progress, you must stop the VolumeCopy before you can remove the VolumeCopy pair from the storage subsystem configuration.

The command takes this form:

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

This example shows the remove volumeCopy command:

```
c:\...\...\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 VolumeCopy premium feature while running the following premium features:

- Storage Partitioning
- FlashCopy Logical Drive
- Enhanced Remote Mirroring

When you are running the VolumeCopy 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 subsystem configuration.

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

Storage Partitioning

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

- A host
- A host group
- A logical drive-to-LUN mapping

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

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

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

FlashCopy Logical Drives

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

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

Tip: You can select a FlashCopy logical drive as the source logical drive for a VolumeCopy. This selection is a good use of this feature, because it performs complete backups without significant impact to the storage subsystem availability. Some I/O processing resources are lost to the copy operation.

Important: If you choose the base logical drive of a FlashCopy logical drive as your target logical drive, you must disable all of the FlashCopy logical drives that are associated with the base logical drive before you can select it as a target logical drive.

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

You can use the FlashCopy Logical Drive premium feature with the VolumeCopy premium feature to back up data on the same storage subsystem and to restore the data on the FlashCopy logical drive back to its original base logical drive.

Enhanced Remote Mirroring

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

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

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

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

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

The secondary logical drive 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 logical drive to a primary role. Hosts are then able to access the newly promoted logical drive, and business operations can continue.

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

Role Reversals

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

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

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

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

You can perform a role reversal by using the `set remoteMirror` command. (For information about the `set remoteMirror` command, see “Set Remote Mirror” on page 97.)

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

```
set remoteMirror role=primary
```

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

```
set remoteMirror role=secondary
```

Chapter 8. Maintaining a Storage System

Maintenance covers a broad spectrum of activity with the goal of keeping a storage subsystem operational and available to all hosts. This chapter provides descriptions of commands you can use to perform storage subsystem 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 subsystem. 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 subsystem 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 disk drive media errors before they are found during a normal read from or write to the disk drives. Any media scan errors that are detected are reported to the Event Log. The Event Log provides an early indication of an impending disk 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 logical drives.

A media scan runs on all of the logical drives in the storage subsystem 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 logical drive are reported to the Major Event Log (MEL) and handled as follows:

- **Unrecovered media error** – The disk drive could not read the requested data on its first try or on any subsequent retries. The result of this action is that for logical drives with redundancy protection, the data is reconstructed, rewritten to the disk drive, and verified, and the error is reported to the Event Log. For logical drives without redundancy protection, the error is not corrected, but it is reported to the Event Log.
- **Recovered media error** – The disk drive could not read the requested data on its first attempt. The result of this action is that the data is rewritten to the disk 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 disk 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 logical drive 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 logical drive. 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 logicalDrive
- set storageSubsystem

The set logicalDrive command enables a media scan for the logical drive. The command takes this form:

```
set (allLogicalDrives | logicalDrive [logicalDriveName] | logicalDrives
[logicalDriveName1 ... logicalDriveNameN] | logicalDrive <wwID>)
mediaScanEnabled=(TRUE | FALSE)
```

The set storageSubsystem command defines how frequently a media scan is run on a storage subsystem. The command takes this form:

```
set storageSubsystem 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 “Running a Media Scan” on page 199.) During a redundancy check, all of the data blocks in a logical drive 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 logical drives, redundancy is checked and repaired.
- For RAID 1 logical drives, the data is compared between the mirrored disk drives and data inconsistencies are repaired.
- RAID 0 logical drives have no redundancy.

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

```
set (allLogicalDrives | logicalDrive [logicalDriveName] | logicalDrives [logicalDriveName1 ...
logicalDriveNameN] | logicalDrive <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 logical drives 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 logical drives 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 logical drives.

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

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

Note: The DS4800 (Models 82, 84, 88) controller module does not support this command. The batteries in the DS4800 (Models 82, 84, 88) controller module do not require that you reset the battery age after you have replaced the batteries.

Removing Persistent Reservations

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

- Change or delete LUN mappings on a logical drive holding a reservation
- Delete arrays or logical drives that have any reservations

To determine which logical drives have reservations, use this command:

```
show (allLogicalDrives | logicalDrive [logicalDriveName]  
| logicalDrives [logicalDriveName1 ... logicalDriveNameN]) reservations
```

To clear persistent logical drive reservations, use this command:

```
clear (allLogicalDrives | logicalDrive [logicalDriveName]  
| logicalDrives [logicalDriveName1 ... logicalDriveNameN]) reservations
```

Synchronizing the Controller Clocks

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

```
set storageSubsystem time
```

Locating Disk Drives

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

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

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

```
stop drive locate
```

Relocating an Array

Array relocation describes the action of moving disk drives within the same storage subsystem. This is a supported capability; however, any relocation of storage subsystem 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 disk drives and then reinstall them into a different storage subsystem.

Hot and Cold Array Relocation

There are two methods you can use to move arrays: hot array relocation and cold array relocation.

Note: For more detailed information about relocating arrays, see one of the following documents:

- **DS3000:** *IBM System Storage DS3x00 Installation, User's and Maintenance Guide*
- **DS4000 / DS5000:** *IBM System Storage DS4000/DS5000 Hard Drive and Storage Expansion Enclosure Installation and Migration Guide*
- Hot array relocation lets you add or move storage without reconfiguring the storage subsystem and, in some cases, without rebooting. During hot array relocation, the storage subsystem power is not turned off.
- Cold array relocation requires that the power to the source storage subsystem and the destination storage subsystem be turned off before moving the arrays from one storage subsystem to another. Then the power to the storage subsystems can be turned on.

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

Note: Attention: Possible loss of data access – You must move a single array at a time, and it must go into a storage subsystem with the same level of controller firmware.

Basic Process Steps

Relocating an array from one storage subsystem to another includes these procedures:

1. Verifying the status of the storage subsystems
2. Ensuring that both storage subsystems have the latest controller firmware versions installed
3. Locating the drives in the array
4. Placing the array offline
5. Removing drives from the storage subsystem
6. Replacing an array into the new storage subsystem

Note: Attention: Check firmware versions – Ensure that both existing and new storage subsystems have the latest controller firmware versions installed.

- See the IBM Disk Support Web site to find the latest firmware versions for your storage subsystems:
www.ibm.com/systems/support/storage/disk
- Also check the following documentation for more information:
 - **DS3000:** *IBM System Storage DS3x00 Installation, User's and Maintenance Guide*
 - **DS4000 / DS5000:** *IBM System Storage DS4000/DS5000 Hard Drive and Storage Expansion Enclosure Installation and Migration Guide*

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.

Array Relocation Commands

Note: Array relocation commands are supported with the following controller firmware only:

- **DS5000:** 7.30.xx.xx
- **DS4000:** 7.1x.xx.xx
- **DS3000:** Not supported.

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

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

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

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

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

Your relocated array is now available for use.

For additional information, refer to these commands:

- “Show Array Export Dependencies” on page 107
- “Show Array Import Dependencies” on page 108
- “Start Array Export” on page 121
- “Start Array Import” on page 121

Performance Tuning

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

Monitoring the Performance

You can monitor the performance of a storage subsystem by using the `save storageSubsystem performanceStats` command. This command saves performance information to a file that you can review to help determine how well the storage subsystem is running. The following table lists the performance information that is saved to the file.

Table 23. Information About Storage Subsystem 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 logical drives that are owned by the controller • Logical drives – A list of the logical drive names • Storage subsystem 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 subsystem 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)
Cache Hit Percentage	The percentage of reads that are fulfilled by data from the cache rather than requiring an actual read from a disk 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 storageSubsystem 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 storageSubsystem 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 an array, you can define the RAID level for the logical drives in that array. You can change the RAID level later to improve performance or provide more secure protection for your data.

Note: RAID 6 is a premium feature for the DS4700 (Model 70) controller module and the DS4700 (Model 72) controller module. You must enable RAID 6 with the premium feature key before you can use the dynamic RAID migration feature.

- | RAID 6 is supported in DS4000/DS5000 storage products only, with controller firmware version
- | 7.1x.xx.xx or higher.

To change the RAID level, use this command:

```
set array [arrayNumber]
raidLevel=(0 | 1 | 3 | 5 | 6)
```

where *arrayNumber* is the number of the array for which you want to change the RAID level.

Changing the Segment Size

When you create a new logical drive, you can define the segment size for that logical drive. 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 disk drives that are needed to satisfy an I/O request. Use larger values for the segment size. Using a single disk drive for a single request leaves other disk drives available to simultaneously service other requests. If the logical drive 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 logicalDrive ([logicalDriveName] | <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 logical drive by name or by WWID. (For usage information, see "Set Logical Drive" on page 93.)

Changing the Cache Parameters

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

- `set storageSubsystem`
- `set logicalDrive`

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

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

The `set logicalDrive` 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 an Array

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

space, logical drive 2, space, logical drive 4, logical drive 5, original unused space

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

logical drive 2, logical drive 4, logical drive 5, consolidated unused space

To defragment a array, use this command:

```
start array [arrayNumber] defragment
```

where *arrayNumber* is the identifier for the array.

Note: If you need to defragment the data stored in the logical drive, you must use the appropriate defragmenting tool for the installed operating system.

Troubleshooting and Diagnostics

If a storage subsystem 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 subsystem, run the `save storageSubsystem 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. The following table lists the type of support data that is collected.

Table 24. Support Data for the Storage Subsystem

Type of Data	Description and File Name
Storage subsystem profile	A list of all components and properties of a storage subsystem. storageSubsystemProfile.txt

Table 24. Support Data for the Storage Subsystem (continued)

Type of Data	Description and File Name
Major Event Log	<p>A detailed list of errors that occur on the storage subsystem. The list is stored in reserved areas on the disk drives in the storage subsystem. The list records configuration events and failures with storage subsystem components.</p> <p>majorEventLog.txt</p>
Read link status	<p>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.</p> <p>readLinkStatus.csv</p>
Switch-on-a-chip (SOC) error statistics	<p>Information from the loop-switch ports that are connected to Fibre Channel devices.</p> <p>socStatistics.csv</p>
NVS RAM	<p>A controller file that specifies the default settings for the controllers.</p> <p>NVSRAMdata.txt</p>
Performance statistics	<p>A detailed description of how a storage subsystem is performing. Collected data includes the I/O activity of specific controllers or logical drives, the transfer rate of the controller, the current I/Os per second, and the maximum I/Os per second.</p> <p>performanceStatistics.csv</p>
Persistent reservations and persistent registrations	<p>A detailed list of logical drives on the storage subsystem and persistent reservations and persistent registrations.</p> <p>persistentRegistrations.txt</p>
Object bundle	<p>A detailed description of the status of the storage subsystem 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.</p> <p>objectBundle</p>
Drive diagnostic data	<p>A detailed list of log sense data from all of the disk drives in the storage subsystem.</p> <p>driveDiagnosticData.txt</p>
Recovery profile	<p>A detailed description of the latest recovery profile record and historical data.</p> <p>recoveryProfile.csv</p>

Table 24. Support Data for the Storage Subsystem (continued)

Type of Data	Description and File Name
Unreadable sectors	A detailed list of all of the unreadable sectors that have been logged to the storage subsystem. badBlocksData.txt
State capture data	A detailed description of the current state of the storage subsystem. stateCaptureData.dmp
Storage array	A detailed listing of the hardware components and the software components that comprise the storage subsystem configuration. storageSubsystemConfiguration.cfg

Collecting Drive Data

To gather information about all of the disk drives in a storage subsystem, 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 disk drives in the storage subsystem.

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 disk 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 disk drives. The test passes data through each controller disk 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 subsystem and any time that you that have made changes to the storage subsystem or the components that are connected to the storage subsystem (such as hubs, switches, and host adapters).

A custom data pattern file called `diagnosticsDataPattern.dpf` is included in the root directory of the installation CD. You can modify this file, but the file must have these properties to work correctly for the tests:

- The file values must be entered in hexadecimal format (00 to FF) with only one space between the values.
- The file must be no larger than 64 bytes in size. Smaller files will work, but larger files can cause an error.

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

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

The following table lists the type of data contained by the file that is generated by the `save storageSubsystem RLSBaseline` command.

Table 25. RLS Baseline Data for the Storage Subsystem

Type of Data	Description
Devices	A list of all devices on the Fibre Channel loop. The devices appear in channel order. Within each channel, the devices are sorted according to the device position in the loop.
Baseline time	The date and time when the baseline was set.
Elapsed time	The time that has elapsed from when the baseline time was set to when the read link status was gathered.

Table 25. RLS Baseline Data for the Storage Subsystem (continued)

Type of Data	Description
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>
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.

Table 25. RLS Baseline Data for the Storage Subsystem (continued)

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 subsystem. (RLS counts provide information about the Fibre Channel devices.) Reporting SOC error statistics is available only on storage subsystems that have SOC loop-switch devices that are incorporated into the controller disk 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 disk 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 statics 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 storageSubsystem SOCBaseline` command

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

After you have established the baseline for the SOC devices, you run the storage subsystem 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 storageSubsystem SOCCounts 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 subsystem 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 subsystem and returning it to an operational state. This might involve replacing a failed CRU, a failed controller, or a disk drive or restoring data or the storage subsystem to operation. For information about when it is appropriate to replace a CRU, see “Routine Maintenance” on page 199.

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 arrays to the other controller if failover protection is enabled.

Taking a controller offline can seriously impact data integrity and storage subsystem 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. Arrays and their associated logical drives 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 logical drives, application errors will occur.

Attention: **Possible loss of data access** – Placing a controller offline can cause loss of data.

Use Service mode to replace CRUs, such as a controller. Placing a controller in Service mode makes it unavailable for I/O operations and moves its arrays to the second controller without affecting the preferred path of the array. This action might significantly reduce performance. The arrays 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 logical drives 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 logical drives are not in use, or a multi-path driver is installed on all of the hosts that are using these logical drives.

In addition, if you do not have a multi-path driver, you must make appropriate operating system-specific modifications to make sure that the arrays 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 logical drive by using the set logical drive command. The command takes this form:

```
set(allLogicalDrives|logicalDrive [logicalDriveName]|logicalDrives [logicalDriveName1...logicalDriveNameN] |logicalDrive <wwID>)owner=(a|b)
```

Initializing a Drive

Attention: **Possible loss of data access** – When you initialize a disk drive, all data on the disk drive is lost.

You must initialize a disk drive when you have moved a disk drive that was previously part of a multidisk array from one storage subsystem to another. If you do not move the entire set of disk drives, the array information and the logical drive information on the disk drives that you move are incomplete. Each disk drive that you move contains only part of the information that is defined for the logical drive and the array. To be able to reuse the disk drives to create a new array and logical drive, you must delete all of the old information from the disk drives by initializing the disk drive.

When you initialize a disk drive, all of the old array information and logical drive information are deleted, and the disk drive is returned to an unassigned state. Returning a disk drive to an unassigned state adds unconfigured capacity to a storage subsystem. You can use this capacity to create additional arrays and logical drives.

To initialize a disk drive, use this command:

```
start drive [enclosureID,slotID] initialize
```

where *enclosureID* and *slotID* are the identifiers for the disk drive.

Reconstructing a Drive

If two or more of the disk drives in an array have failed, the logical drive shows a status of Failed. All of the logical drives in the array are no longer operating. To return the array to an Optimal status, you must replace the failed disk drives. Then, you must reconstruct the data on the new disk drives. The data that you reconstruct is the data as it would appear on the failed disk drives.

Important: You can use this command only when the disk drive is assigned to a RAID 1, RAID 3, RAID 5, or RAID 6 array.

To reconstruct a disk drive, use this command:

```
start drive [enclosureID,slotID] reconstruct
```

where *enclosureID* and *slotID* are the identifiers for the disk drive.

Initializing a Logical Drive

Attention: Possible loss of data – When you initialize a logical drive, all existing data on the logical drive and all of the information about the logical drive are destroyed. Do not use this command unless you want to destroy all the data in the given disk drive.

A logical drive is automatically initialized when you first create it. If the logical drive starts showing failures, you might be required to re-initialize the logical drive to correct the failure condition.

Consider these restrictions when you initialize a logical drive:

- You cannot cancel the operation after it begins.
- You cannot use this option if any modification operations are in progress on the logical drive or the array.
- You cannot change the cache parameters of the logical drive while the initialization operation is in progress.

To initialize a logical drive, use this command:

```
start logicalDrive [logicalDriveName] initialize
```

where *logicalDriveName* is the identifier for the logical drive.

Redistributing Logical Drives

When you redistribute logical drives, you return the logical drives to their preferred controller owners. The preferred controller ownership of a logical drive or a array is the controller of an active-active pair that is designated to own the logical drives. The preferred owner for a logical drive is initially designated when the logical drive is created. If the preferred controller is being replaced or undergoing a firmware download, ownership of the logical drives is automatically shifted to the other controller. That controller becomes the current owner of the logical drives. This change is considered to be a routine ownership change and is reported in the Event Log.

To redistribute logical drives to their preferred controllers, use this command:

```
reset storageSubsystem logicalDriveDistribution
```

Important: If you run this command without a multi-path driver on the hosts, stop I/O activity to the logical drives to prevent application errors.

Important: You cannot run this command if all of the logical drives are currently owned by their preferred controller, or the storage subsystem does not have defined logical drives.

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 logical drive.

Replacing CRUs

- | The DS3000, DS4000, and DS5000 controller module components, such as the controller CRUs, the power-fan CRUs, and the interconnect-battery CRUs, have a Service Action Allowed (SAA[®]) indicator light. This indicator light is a blue LED. The Service Action Allowed indicator light helps to make sure
- | that you do not remove a CRU before it is safe to do so.

Attention: Possible loss of data access – Never remove a component that has a Service Action Required indicator light on unless the Service Action Allowed (SAA) indicator light is on.

If a component fails and must be replaced, the Service Action Required indicator light on that CRU comes on to indicate that service action is required, provided no data availability dependencies or other conditions exist that dictate the CRU 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 CRU.

If the interconnect-battery CRU 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 CRU can come on, you must place controller CRU 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 CRU is removed. The Service Action Allowed indicator light comes on after the new CRU has been installed.

The ability to remove a CRU depends on the data availability dependencies of the controller module or the controller module. The Service Action Allowed indicator light does not come on if removing a CRU jeopardizes data on the expansion drawers or current I/O activity. An example of limiting when you can remove a CRU is when one controller CRU has a Service Action Required indicator light on. You cannot remove the other controller CRU (the Service Action Allowed indicator light does not come on), because doing so would jeopardize the data either on the expansion drawers or transitioning through the controllers.

A less obvious example, as in the DS4800 storage subsystem, is when the power supply for the controller CRU in slot A has failed, and the controller CRU in slot B has failed. Removing the controller CRU in slot B before replacing the failed power-fan CRU causes the controller CRU in slot A to lose power, which results in a loss of data access. This action occurs because power distribution from each power-fan CRU is through the controller CRU that is physically connected to that power-fan CRU.

So, in the preceding example, these actions occur:

- The power-fan CRU has both its Service Action Required indicator light and its Service Action Allowed indicator light on.
- The controller CRU 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 CRU has been replaced, the Service Action Allowed indicator light comes on for the controller CRU in slot B.

For more information about the Service Action Not Allowed indicator, please see the *IBM System Storage DSxxxx Storage Subsystem Installation, User's and Maintenance Guide* for your storage subsystem product.

Chapter 9. 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 subsystem.

Show Storage Subsystem Command

The show storageSubsystem command returns information about the components and the features in a storage subsystem. 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 subsystem. After you have configured a storage subsystem, save the configuration description to a file as a reference.

PROFILE FOR STORAGE SUBSYSTEM: DS4700_BOT (5/22/08 3:17:16 PM)

SUMMARY-----

Number of controllers: 2

High performance tier controllers: Enabled

Number of logical drive groups: 2

RAID 6: Enabled

Total number of logical drives used: 9
Number of standard logical drives: 8
Number of access logical drives: 1
Total number of logical drives allowed: 1024

FlashCopy Logical Drives: Enabled
Number of flashcopies used: 0
Number of flashcopies allowed: 2
Number of flashcopies allowed per base logical drive: 2

Remote Mirroring: Disabled/Deactivated
Number of mirrors used: 0
Number of mirrors allowed: 0

VolumeCopy: Enabled
Number of copies used: 0
Number of copies allowed: 1024

Number of drives: 46
Mixed drive types: Enabled
Current drive type(s): Fibre (15), Serial ATA (SATA) (31)
Total hot spare drives: 5
Standby: 5
In use: 0

Number of drive enclosures: 2
Number of drive enclosures allowed: 7

Storage Partitioning: Enabled
Number of partitions used: 1
Number of partitions allowed: 128
Number of logical drives allowed per partition: 256

Access logical drive: LUN 31,31 (see Mappings section for details)
Default host OS: Windows 2000/Server 2003/Server 2008 Non-Clustered (Host OS index 2)

Current configuration
Firmware version: 07.15.07.00
NVS RAM version: N1814D47R1010V05
EMW version: 10.15.G5.08
AMW version: 10.15.G5.08

Pending configuration
Staged firmware download supported: Yes
Firmware version: None
NVS RAM version: None
Transferred on: None
Controller enclosure audible alarm: Disabled

NVS RAM 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): 30

Failover alert delay (in minutes): 5

Feature enable identifier: 303532343600303533363800482D72AD

Feature pack: Generic
Feature pack submodel ID: 20

Storage Subsystem world-wide identifier (ID): 600A0B80002954B600000000482D72AC

CONTROLLERS-----

Number of controllers: 2

Controller in Enclosure 85, Slot A

Status: Online

Current configuration
Firmware version: 07.15.07.00
Appware version: 07.15.07.00
Bootware version: 07.15.07.00
NVS RAM version: N1814D47R1010V05

Pending configuration
Firmware version: None
Appware version: None
Bootware version: None
NVS RAM version: None
Transferred on: None

Current ID (ALPA)
On drive channel 1: 125/0x1
Replacement part number: 41Y0676

```

Board ID:          3994
Submodel ID:       20
Product ID:        1814      FAStT
Product revision:  0916
Serial number:     SN71705246
Date of manufacture: May 11, 2007
Cache
  Total data cache: 1709 MB
  Accessible data cache: 0 MB
  Processor cache:  339 MB
Date/Time:         Thu May 22 15:15:23 EDT 2008

```

```

Associated Logical Drives (* = Preferred Owner):
Array_1_R6_A*, Array_1_R6_B, Array_1_R6_C*, Array_1_R6_D, Array_2_R5_A,
Array_2_R5_B*, Array_2_R5_C, Array_2_R5_D*

```

```

Controller host name: 4700_top_A
Remote login:         Enabled*

```

```

Ethernet port:      1
MAC address:        00:a0:b8:29:54:b6
Negotiation mode:   Manual setting
  Port speed:       10/100 Mbps
  Duplex mode:      Full duplex
IPv4 settings:      Enabled
  Network configuration: Static
  IP address:       192.168.70.153
  Subnet mask:      255.255.255.0
  Gateway:          192.168.70.1
IPv6 settings:      Enabled
  Auto-configuration: Stateless
  Local IP address: FE80:0000:0000:0000:02A0:B8FF:FE29:54B6

```

```

Ethernet port:      2
MAC address:        00:a0:b8:29:54:b7
Negotiation mode:   Manual setting
  Port speed:       Unknown
  Duplex mode:      Half duplex
IPv4 settings:      Enabled
  Network configuration: Static
  IP address:       192.168.129.101
  Subnet mask:      255.255.255.0
  Gateway:          192.168.70.1
IPv6 settings:      Enabled
  Auto-configuration: Stateless
  Local IP address: FE80:0000:0000:0000:02A0:B8FF:FE29:54B7

```

*NVS RAM setting, may be overridden by DHCP/Bootp server setting

```

Drive interface:    Fibre
Channel:            1
Port:               2, 1, Out, Out
Current ID:         125/0x1
Maximum data rate:  4 Gbps
Current data rate:  4 Gbps
Data rate control:  Switch
Link status:        Up

```

```

Drive interface:    Fibre
Channel:            2
Port:               3, 4, Out, Out
Current ID:         125/0x1
Maximum data rate:  4 Gbps
Current data rate:  4 Gbps

```

```

Data rate control: Switch
Link status: Up

Host interface: Fibre
Channel: 1
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 0/0xEF
NL-Port ID: 0x010400
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Up
Topology: Fabric Attach
World-wide port identifier: 20:16:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 2
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 1/0xE8
NL-Port ID: 0x010700
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Up
Topology: Fabric Attach
World-wide port identifier: 20:26:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 3
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 2/0xE4
NL-Port ID: 0xFFFFFFFF
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Down
Topology: Not Available
World-wide port identifier: 20:36:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 4
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 3/0xE2
NL-Port ID: 0xFFFFFFFF
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Down
Topology: Not Available
World-wide port identifier: 20:46:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

```

Controller in Enclosure 85, Slot B

```

Status: Online

Current configuration
Firmware version: 07.15.07.00
Appware version: 07.15.07.00

```

```

    Bootware version: 07.15.07.00
    NVSRAM version: N1814D47R1010V05
Pending configuration
    Firmware version: None
    Appware version: None
    Bootware version: None
    NVSRAM version: None
    Transferred on: None
Current ID (ALPA)
    On drive channel 1: 124/0x2
Replacement part number: 41Y0676
Board ID: 3994
Submodel ID: 20
Product ID: 1814 FASTT
Product revision: 0916
Serial number: SN71705368
Date of manufacture: May 11, 2007
Cache
    Total data cache: 1709 MB
    Accessible data cache: 0 MB
    Processor cache: 339 MB
Date/Time: Thu May 22 15:15:28 EDT 2008

```

Associated Logical Drives (* = Preferred Owner): None

```

Controller host name: 4700_top_B
Remote login: Enabled*

```

```

Ethernet port: 1
MAC address: 00:a0:b8:29:53:ea
Negotiation mode: Manual setting
Port speed: 10/100 Mbps
Duplex mode: Full duplex
IPv4 settings: Enabled
Network configuration: Static
IP address: 192.168.70.154
Subnet mask: 255.255.255.0
Gateway: 192.168.70.1
IPv6 settings: Enabled
Auto-configuration: Stateless
Local IP address: FE80:0000:0000:0000:02A0:B8FF:FE29:53EA

```

```

Ethernet port: 2
MAC address: 00:a0:b8:29:53:eb
Negotiation mode: Manual setting
Port speed: Unknown
Duplex mode: Half duplex
IPv4 settings: Enabled
Network configuration: Static
IP address: 192.168.129.102
Subnet mask: 255.255.255.0
Gateway: 192.168.70.1
IPv6 settings: Enabled
Auto-configuration: Stateless
Local IP address: FE80:0000:0000:0000:02A0:B8FF:FE29:53EB

```

*NVSRAM setting, may be overridden by DHCP/Bootp server setting

```

Drive interface: Fibre
Channel: 1
Port: 2, 1, Out, Out
Current ID: 124/0x2
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps

```

```

Data rate control: Switch
Link status: Up

Drive interface: Fibre
Channel: 2
Port: 3, 4, Out, Out
Current ID: 124/0x2
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Switch
Link status: Up

Host interface: Fibre
Channel: 1
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 4/0xE1
NL-Port ID: 0x010200
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Up
Topology: Fabric Attach
World-wide port identifier: 20:17:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 2
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 5/0xE0
NL-Port ID: 0x010000
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Up
Topology: Fabric Attach
World-wide port identifier: 20:27:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 3
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 6/0xDC
NL-Port ID: 0xFFFFFFFF
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Down
Topology: Not Available
World-wide port identifier: 20:37:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

Host interface: Fibre
Channel: 4
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 7/0xDA
NL-Port ID: 0xFFFFFFFF
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Data rate control: Auto
Link status: Down
Topology: Not Available
World-wide port identifier: 20:47:00:a0:b8:29:54:b6
World-wide node identifier: 20:06:00:a0:b8:29:54:b6
Part type: HPFC-5700 revision 5

```

ARRAYS-----

Number of logical drive groups: 2

Name: Array_1_R6
Status: Optimal
Capacity 339.33 GB
RAID level: 6
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A

Associated logical drives and free capacity

Logical Drive	Capacity
Array_1_R6_A	10.000 GB
Free Capacity	329.330 GB

Associated drives - present (in piece order)

Enclosure	Slot
6	8
4	9
6	12
4	11
6	14
4	13
6	15

Name: Array_2_R5
Status: Optimal
Capacity 930.523 GB
RAID level: 5
Drive type: Serial ATA (SATA)
Enclosure loss protection: Yes
Current owner: Controller in slot A

Associated logical drives and free capacity

Logical Drive	Capacity
Array_2_R5_A	10.000 GB
Free Capacity	920.523 GB

Associated drives - present (in piece order)

Enclosure	Slot
85	16
6	1
4	2

STANDARD LOGICAL DRIVES-----

SUMMARY

Number of standard logical drives: 8

See other Logical Drives sub-tabs for premium feature information.

NAME	STATUS	CAPACITY	RAID LEVEL	ARRAY	DRIVE TYPE
Array_1_R6_A	Optimal	10.0 GB	6	Array_1_R6	Fibre
Array_2_R5_A	Optimal	10.0 GB	5	Array_2_R5	SATA

DETAILS

```

Logical Drive name:                Array_1_R6_A
Logical Drive status:              Optimal
Capacity:                          10.0 GB
Logical Drive world-wide identifier: 60:0a:0b:80:00:29:54:b6:00:00:a4:d4:48:2d:8a:0d
Subsystem ID (SSID):               0
Associated array:                  Array_1_R6
RAID level:                         6

Drive type:                        Fibre Channel
Enclosure loss protection:         No

Preferred owner:                   Controller in slot A
Current owner:                     Controller in slot A

Segment size:                      128 KB
Capacity reserved for future segment size changes: Yes
Maximum future segment size:       2,048 KB
Modification priority:              High

Read cache:                         Enabled
Write cache:                        Enabled
  Write cache without batteries:    Disabled
  Write cache with mirroring:       Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch:       Enabled

Enable background media scan:       Enabled
Media scan with redundancy check:   Disabled

Pre-Read redundancy check:         Disabled

```

```

Logical Drive name:                Array_2_R5_A
Logical Drive status:              Optimal
Capacity:                          10.0 GB
Logical Drive world-wide identifier: 60:0a:0b:80:00:29:53:ea:00:00:ac:29:48:2d:8b:55

```

```

Subsystem ID (SSID):          3
Associated array:             Array_2_R5
RAID level:                   5

Drive type:                   Serial ATA (SATA)
Enclosure loss protection:    Yes

Preferred owner:              Controller in slot B
Current owner:                Controller in slot A

Segment size:                 128 KB
Capacity reserved for future segment size changes: Yes
Maximum future segment size: 2,048 KB
Modification priority:        High

```

```

Read cache:                   Enabled
Write cache:                   Enabled
  Write cache without batteries: Disabled
  Write cache with mirroring:   Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch:   Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

Pre-Read redundancy check:    Disabled

```

FlashCopy REPOSITORY LOGICAL DRIVES-----

Number of flashcopy repositories: 0

FlashCopy LOGICAL DRIVES-----

Number of flashcopy logical drives: 0

COPIES-----

Number of copies: 0

MISSING LOGICAL DRIVES-----

Number of missing logical drives: 0

DRIVES-----

SUMMARY

Number of drives: 46
 Current drive types: Fibre (15), Serial ATA (SATA) (31)

(Note to reader: Some information is removed from this section of the example.)

BASIC:

TRAY, SLOT	STATUS	CAPACITY	TYPE	CURRENT DATA RATE	PRODUCT ID	FIRMWARE VERSION
4, 1	Optimal	698.638 GB	SATA	4 Gbps	ST3750640NS	3.AEH
4, 2	Optimal	465.762 GB	SATA	4 Gbps	ST3500641NS	3.AEN
4, 3	Optimal	698.638 GB	SATA	4 Gbps	ST3750640NS	3.AEH
4, 4	Optimal	465.762 GB	SATA	4 Gbps	ST3500641NS	3.AEN
4, 5	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360	K2A0AD4A
4, 6	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360	K2A0AD4A
4, 7	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360	K2A0AD4A
4, 8	Optimal	33.902 GB	Fibre	4 Gbps	MAX3036FD F	S708
4, 9	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD F	S708

4, 11	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD	F	S708
4, 12	Optimal	33.902 GB	Fibre	4 Gbps	MAX3036FD	F	S708
4, 13	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD	F	S708
4, 15	Optimal	68.366 GB	Fibre	4 Gbps	ST373554FC	F	B909
4, 16	Optimal	33.902 GB	Fibre	4 Gbps	ST336854FC	F	B90A
6, 1	Optimal	465.762 GB	SATA	4 Gbps	ST3500641NS		3.AEN
6, 2	Optimal	698.638 GB	SATA	4 Gbps	ST3750640NS		3.AEH
6, 3	Optimal	698.638 GB	SATA	4 Gbps	ST3750640NS		3.AEH
6, 4	Optimal	698.638 GB	SATA	4 Gbps	ST3750640NS		3.AEH
6, 5	Optimal	465.762 GB	SATA	4 Gbps	ST3500641NS		3.AEN
6, 6	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
6, 7	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
6, 8	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD	F	S708
6, 9	Optimal	33.902 GB	Fibre	4 Gbps	MAX3036FD	F	S708
6, 10	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
6, 11	Optimal	279.397 GB	Fibre	4 Gbps	ST3300655FC	F	B972
6, 12	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD	F	S708
6, 13	Optimal	33.902 GB	Fibre	4 Gbps	MAX3036FD	F	S708
6, 14	Optimal	68.366 GB	Fibre	4 Gbps	MAX3073FD	F	S708
6, 15	Optimal	68.366 GB	Fibre	4 Gbps	ST373554FC	F	B909
6, 16	Optimal	33.902 GB	Fibre	4 Gbps	ST336854FC	F	B90A
85, 1	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 2	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 3	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 4	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 5	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 6	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 7	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 8	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 9	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 10	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 11	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 12	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 13	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 14	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 15	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A
85, 16	Optimal	465.762 GB	SATA	4 Gbps	HDS725050KLA360		K2A0AD4A

DRIVE CHANNELS:

TRAY, SLOT	PREFERRED CHANNEL	REDUNDANT CHANNEL
4, 1	1	2
4, 2	2	1
4, 3	1	2
4, 4	2	1
4, 5	1	2
4, 6	2	1
4, 7	1	2
4, 8	2	1
4, 9	1	2
4, 11	1	2
4, 12	2	1
4, 13	1	2
4, 15	1	2
4, 16	2	1
6, 1	1	2
6, 2	2	1
6, 3	1	2
6, 4	2	1
6, 5	1	2
6, 6	2	1
6, 7	1	2
6, 8	2	1
6, 9	1	2
6, 10	2	1
6, 11	1	2
6, 12	2	1

6, 13	1	2
6, 14	2	1
6, 15	1	2
6, 16	2	1
85, 1	1	2
85, 2	2	1
85, 3	1	2
85, 4	2	1
85, 5	1	2
85, 6	2	1
85, 7	1	2
85, 8	2	1
85, 9	1	2
85, 10	2	1
85, 11	1	2
85, 12	2	1
85, 13	1	2
85, 14	2	1
85, 15	1	2
85, 16	2	1

HOT SPARE COVERAGE:

The following logical drive groups are not protected: None - All logical drive groups are protected
 Total hot spare drives: 5

Standby: 5
 In use: 0

Standby drive at enclosure 4, slot 15 (Fibre, 68.366 GB)
 Protects the following logical drive groups: Array_1_R6

Standby drive at enclosure 6, slot 11 (Fibre, 279.397 GB)
 Protects the following logical drive groups: Array_1_R6

Standby drive at enclosure 85, slot 1 (SATA, 465.762 GB)
 Protects the following logical drive groups: Array_2_R5

Standby drive at enclosure 4, slot 3 (SATA, 698.638 GB)
 Protects the following logical drive groups: Array_2_R5

Standby drive at enclosure 6, slot 3 (SATA, 698.638 GB)
 Protects the following logical drive groups: Array_2_R5

DETAILS

Drive at Enclosure 4, Slot 1

Status: Optimal

Mode: Unassigned

Raw capacity: 698.638 GB

Usable capacity: 698.138 GB

World-wide identifier: 20:00:00:a0:b8:28:4d:5c:00:00:00:00:00:00:00:00

Associated array: None

Port	Channel	ID
0	1	0/0xEF
1	2	32/0xB2

Drive path redundancy: OK

Drive type: Serial ATA (SATA)

Speed: 7200 RPM

Current data rate: 4 Gbps
Product ID: ST3750640NS 43W9715 42D0003IBM
Package version: EH56
Firmware version: 3.AEH
Serial number: 3QD052B7
Vendor: Not Available
Date of manufacture: Not Available

ATA Translator
Product ID: BR-2401-3.0
Vendor: SLI
Firmware Version: LP1156
Location: Individual Drive Canister

Drive at Enclosure 4, Slot 2

Status: Optimal
Mode: Assigned
Raw capacity: 465.762 GB
Usable capacity: 465.262 GB
World-wide identifier: 20:00:00:a0:b8:1e:26:91:00:00:00:00:00:00:00:00
Associated array: Array_2_R5

Port	Channel	ID
0	2	33/0xB1
1	1	1/0xE8

Drive path redundancy: OK
Drive type: Serial ATA (SATA)

Speed: 7200 RPM
Current data rate: 4 Gbps
Product ID: ST3500641NS 39M4557 39M0181 IBM
Package version: EN46
Firmware version: 3.AEN
Serial number: 3PM0GL34
Vendor: Not Available
Date of manufacture: Not Available

ATA Translator
Product ID: BR-2401-3.0
Vendor: SLI
Firmware Version: LP1145
Location: Individual Drive Canister

(Note to reader: Some information is removed from this section of the example.)

Drive at Enclosure 4, Slot 8

Status: Optimal
Mode: Unassigned
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
World-wide identifier: 50:00:00:e0:12:69:8a:40:00:00:00:00:00:00:00:00
Associated array: None

Port	Channel	ID
0	2	39/0xA9
1	1	7/0xDA

Drive path redundancy: OK
Drive type: Fibre Channel

Speed: 15000 RPM
Current data rate: 4 Gbps
Product ID: MAX3036FD F
Firmware version: S708
Serial number: DVR9P67001G3
Vendor: IBM-SSG
Date of manufacture: Not Available

Drive at Enclosure 4, Slot 9

Status: Optimal

Mode: Assigned
Raw capacity: 68.366 GB
Usable capacity: 67.866 GB
World-wide identifier: 50:00:00:e0:11:9b:9d:30:00:00:00:00:00:00:00:00
Associated array: Array_1_R6

Port	Channel	ID
0	1	8/0xD9
1	2	16/0xCD

Drive path redundancy: OK
Drive type: Fibre Channel

Speed: 15000 RPM
Current data rate: 4 Gbps
Product ID: MAX3073FD F
Firmware version: S708
Serial number: DVF9P6100017
Vendor: IBM-SSG
Date of manufacture: Not Available

(Note to reader: Some information is removed from this section of the example.)

Drive at Enclosure 85, Slot 1

Status: Optimal

Mode: Hot spare standby
Raw capacity: 465.762 GB
Usable capacity: 465.262 GB
World-wide identifier: 20:00:00:a0:b8:28:2f:3d:00:00:00:00:00:00:00:00
Associated array: None

Port	Channel	ID
0	1	32/0xB2
1	2	40/0xA7

Drive path redundancy: OK
Drive type: Serial ATA (SATA)

Speed: 7200 RPM
Current data rate: 4 Gbps
Product ID: HDS725050KLA360 IBM
Package version: AD4A

Firmware version: K2A0AD4A
Serial number: KRVN67ZAJ3D0XF
Vendor: Not Available
Date of manufacture: Not Available

ATA Translator
Product ID: BR-2401-3.0
Vendor: SLI
Firmware Version: LP1131b
Location: Individual Drive Canister

(Note to reader: Some information is removed from this section of the example.)

DRIVE CHANNELS-----

SUMMARY

CHANNEL	PORT	STATUS	CTRL A LINK	CTRL B LINK
1	2,1,Out,Out	Optimal	Up	Up
2	3,4,Out,Out	Optimal	Up	Up

DETAILS

DRIVE CHANNEL 1

Port: 2, 1, Out, Out
Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 4 Gbps
Rate Control: Switch
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 30
Connected to: A, Port 2
Attached drives: 14
Drive enclosure: 4 (14 drives)
Connected to: A, Port 1
Attached drives: 16
Drive enclosure: 6 (16 drives)

CUMULATIVE ERROR COUNTS

Controller A

Baseline time set: 5/22/08 3:24:24 PM
Sample period (hh:mm:ss): 00:00:00
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: N/A
Total I/O count: 0

Controller B

Baseline time set: 5/22/08 3:24:24 PM
Sample period (hh:mm:ss): 00:00:00
Controller detected errors: 5
Drive detected errors: 124
Timeout errors: 0
Link down errors: N/A
Total I/O count: 1189180

DRIVE CHANNEL 2

Port: 3, 4, Out, Out
Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 4 Gbps
Rate Control: Switch
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 30
Connected to: B, Port 3
Attached drives: 14
Drive enclosure: 4 (14 drives)
Connected to: B, Port 4
Attached drives: 16
Drive enclosure: 6 (16 drives)

CUMULATIVE ERROR COUNTS

Controller A

Baseline time set: 5/22/08 3:24:24 PM
Sample period (hh:mm:ss): 00:00:00
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: N/A
Total I/O count: 0

Controller B

Baseline time set: 5/22/08 3:24:24 PM
Sample period (hh:mm:ss): 00:00:00
Controller detected errors: 25
Drive detected errors: 104
Timeout errors: 1
Link down errors: N/A
Total I/O count: 1039422

ENCLOSURES-----

Drive Enclosure 4 Overall Component Information

Enclosure audible alarm: Disabled
Enclosure path redundancy: OK
Current drive types: Fibre channel
Serial ATA (SATA)
Part number: PN 39M5707
Serial number: SN 1T54614114
Vendor: VN IBM
Date of manufacture: January 1, 2006

2 ESM Canisters Detected

ESM card status: Optimal
Firmware version: 98B5
Configuration settings version: Not Available
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Location: A (left)
Card communication: OK

Product ID: EXP810
Part number: PN 13677-03
Serial number: SN 1T53234028
Vendor: IBM
Date of manufacture: October 1, 2005

ESM card status: Optimal
Firmware version: 98B5
Configuration settings version: Not Available
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Location: B (right)
Card communication: OK
Product ID: EXP810
Part number: PN 21749-00
Serial number: SN 1T54136508
Vendor: IBM
Date of manufacture: December 1, 2005

2 SFPs Detected

SFP status: Optimal
Attached to: ESM Canister A (left)
Location: Port 1B, Out
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 01 9c
Revision:
Part number: JSH-42S3AB3
Serial number: F53896750207
Vendor: JDS UNIPHASE
Date of manufacture: September 20, 2005

SFP status: Optimal
Attached to: ESM Canister B (right)
Location: Port 1B, Out
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 04 85
Revision: 1
Part number: PLRXPLVCSG324N
Serial number: C547RQZC7
Vendor: PICOLIGHT
Date of manufacture: November 19, 2005

2 Power-Fan Canisters Detected

Power-fan canister (right) status: Optimal
Part number: PN 39M5906
Serial number: SN YPT060111438
Vendor: VN IBM
Date of manufacture: January 1, 2006

Power-fan canister (left) status: Optimal
Part number: PN 39M5906
Serial number: SN YPT060111431
Vendor: VN IBM
Date of manufacture: January 1, 2006

2 Power Supplies Detected

Power supply status: Optimal
Location: Power supply canister (right)
Part number: PN 39M5906
Serial number: SN YPT060111438
Vendor: VN IBM
Date of manufacture: January 1, 2006

Power supply status: Optimal
Location: Power supply canister (left)
Part number: PN 39M5906
Serial number: SN YPT060111431
Vendor: VN IBM
Date of manufacture: January 1, 2006

2 Fans Detected

Fan Status: Optimal
Location: Power-fan canister (left)

Fan Status: Optimal
Location: Power-fan canister (right)

4 Temperature Sensors Detected

Temperature sensor status: Optimal
Location: ESM Canister A (left)

Temperature sensor status: Optimal
Location: ESM Canister B (right)

Temperature sensor status: Optimal
Location: Power-fan canister (left)

Temperature sensor status: Optimal
Location: Power-fan canister (right)

Drive Enclosure 6 Overall Component Information

Enclosure audible alarm: Disabled
Enclosure path redundancy: OK
Current drive types: Fibre channel
Serial ATA (SATA)
Part number: PN 20321-00
Serial number: SN 1T54402542
Vendor: VN ENGENIO

Date of manufacture: January 1, 2006

2 ESM Canisters Detected

ESM card status: Optimal
Firmware version: 98B5
Configuration settings version: Not Available
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Location: A (left)
Card communication: OK
Product ID: EXP810
Part number: PN 19695-00
Serial number: SN 1T54136017
Vendor: IBM
Date of manufacture: October 1, 2005

ESM card status: Optimal
Firmware version: 98B5
Configuration settings version: Not Available
Maximum data rate: 4 Gbps
Current data rate: 4 Gbps
Location: B (right)
Card communication: OK
Product ID: EXP810
Part number: PN 21750-00
Serial number: SN 1T54507828
Vendor: IBM
Date of manufacture: November 1, 2005

2 SFPs Detected

SFP status: Optimal
Attached to: ESM Canister A (left)
Location: Port 1B, Out
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 17 6a
Revision:
Part number: AFBR-57R5AEZ
Serial number: A8063603JP
Vendor: AVAGO
Date of manufacture: September 10, 2006

SFP status: Optimal
Attached to: ESM Canister B (right)
Location: Port 1B, Out
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 17 6a
Revision:
Part number: AFBR-57R5AEZ
Serial number: A8063603JS
Vendor: AVAGO
Date of manufacture: September 10, 2006

2 Power-Fan Canisters Detected

Power-fan canister (right) status: Optimal
Part number: PN 15240-07
Serial number: SN YPT060521983
Vendor: VN ENGENIO
Date of manufacture: February 1, 2006

Power-fan canister (left) status: Optimal
Part number: PN 15240-07
Serial number: SN YPT060522023
Vendor: VN ENGENIO
Date of manufacture: February 1, 2006

2 Power Supplies Detected

Power supply status: Optimal
Location: Power supply canister (right)
Part number: PN 15240-07
Serial number: SN YPT060521983
Vendor: VN ENGENIO
Date of manufacture: February 1, 2006

Power supply status: Optimal
Location: Power supply canister (left)
Part number: PN 15240-07
Serial number: SN YPT060522023
Vendor: VN ENGENIO
Date of manufacture: February 1, 2006

2 Fans Detected

Fan Status: Optimal
Location: Power-fan canister (left)

Fan Status: Optimal
Location: Power-fan canister (right)

4 Temperature Sensors Detected

Temperature sensor status: Optimal
Location: ESM Canister A (left)

Temperature sensor status: Optimal
Location: ESM Canister B (right)

Temperature sensor status: Optimal
Location: Power-fan canister (left)

Temperature sensor status: Optimal
Location: Power-fan canister (right)

Controller/Drive Enclosure Overall Component Information

Enclosure audible alarm: Disabled
Enclosure path redundancy: OK
Current drive types: Serial ATA (SATA)
Part number: PN 42D3314
Serial number: SN SG70313544
Vendor: VN IBM
Date of manufacture: June 1, 2007

2 Batteries Detected

Battery status: Optimal
Location: Controller A
Age: 902 days
Days until replacement: 2,788 days
Part number: PN 41Y0679
Serial number: SN 100757541
Vendor: VN IBM
Date of manufacture: March 1, 2007

Battery status: Optimal
Location: Controller B
Age: 902 days
Days until replacement: 2,788 days
Part number: PN 41Y0679
Serial number: SN 100757542
Vendor: VN IBM
Date of manufacture: March 1, 2007

8 SFPs Detected

SFP status: Optimal
Attached to: Drive-side of controller A
Location: Channel 1, Port 1
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 04 85
Revision: 1
Part number: PLRXPLVESG464N
Serial number: C707SVZQD
Vendor: PICOLIGHT
Date of manufacture: May 1, 2007

SFP status: Optimal
Attached to: Drive-side of controller A
Location: Channel 1, Port 0
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 04 85
Revision: 1
Part number: PLRXPLVESG462N
Serial number: C753SP1DH

```

Vendor:                PICOLIGHT
Date of manufacture:   January 3, 2008

SFP status:           Optimal
Attached to:           Host-side of controller A
Location:              Channel 2
Supported data rate(s): 4 Gbps
Link length:          Intermediate
Connector:            LC
Transmitter type:     Shortwave Laser w/o OFC
Transmission media:   TM Multi-mode 62.5m(M6)
IEEE company ID:      00 04 85
Revision:             1
Part number:          PLRXPLVESG464N
Serial number:        C652SVZSV
Vendor:               PICOLIGHT
Date of manufacture:   January 1, 2007

SFP status:           Optimal
Attached to:           Host-side of controller A
Location:              Channel 1
Supported data rate(s): 4 Gbps
Link length:          Intermediate
Connector:            LC
Transmitter type:     Shortwave Laser w/o OFC
Transmission media:   TM Multi-mode 62.5m(M6)
IEEE company ID:      00 04 85
Revision:             1
Part number:          PLRXPLVESG464N
Serial number:        C652SVZSW
Vendor:               PICOLIGHT
Date of manufacture:   January 1, 2007

SFP status:           Optimal
Attached to:           Host-side of controller B
Location:              Channel 1
Supported data rate(s): 4 Gbps
Link length:          Intermediate
Connector:            LC
Transmitter type:     Shortwave Laser w/o OFC
Transmission media:   TM Multi-mode 62.5m(M6)
IEEE company ID:      00 04 85
Revision:             1
Part number:          PLRXPLVESG464N
Serial number:        C652SVZSU
Vendor:               PICOLIGHT
Date of manufacture:   January 1, 2007

SFP status:           Optimal
Attached to:           Host-side of controller B
Location:              Channel 2
Supported data rate(s): 4 Gbps
Link length:          Intermediate
Connector:            LC
Transmitter type:     Shortwave Laser w/o OFC
Transmission media:   TM Multi-mode 62.5m(M6)
IEEE company ID:      00 04 85
Revision:             1
Part number:          PLRXPLVESG464N
Serial number:        C652SVZST
Vendor:               PICOLIGHT
Date of manufacture:   January 1, 2007

```

SFP status: Optimal
Attached to: Drive-side of controller B
Location: Channel 2, Port 0
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 04 85
Revision: 1
Part number: PLRXPLVESG464N
Serial number: C707SVZQF
Vendor: PICOLIGHT
Date of manufacture: May 1, 2007

SFP status: Optimal
Attached to: Drive-side of controller B
Location: Channel 2, Port 1
Supported data rate(s): 4 Gbps
Link length: Intermediate
Connector: LC
Transmitter type: Shortwave Laser w/o OFC
Transmission media: TM Multi-mode 62.5m(M6)
IEEE company ID: 00 04 85
Revision: 1
Part number: PLRXPLVESG462N
Serial number: C752SP2U4
Vendor: PICOLIGHT
Date of manufacture: December 30, 2007

2 Power-Fan Canisters Detected

Power-fan canister (right) status: Optimal
Part number: PN 42D3346
Serial number: SN YPT072272280
Vendor: VN IBM
Date of manufacture: May 1, 2007

Power-fan canister (left) status: Optimal
Part number: PN 42D3346
Serial number: SN YPT072272279
Vendor: VN IBM
Date of manufacture: May 1, 2007

2 Power Supplies Detected

Power supply status: Optimal
Location: Power-fan canister (right)

Power supply status: Optimal
Location: Power-fan canister (left)

2 Fans Detected

Fan Status: Optimal
Location: Power-fan canister (left)

Fan Status: Optimal
Location: Power-fan canister (right)

4 Temperature Sensors Detected

Temperature sensor status: Optimal
Location: Controller A

Temperature sensor status: Optimal
Location: Controller B

Temperature sensor status: Optimal
Location: Power-fan canister (left)

Temperature sensor status: Optimal
Location: Power-fan canister (right)

MAPPINGS (Storage Partitioning - Enabled (1 of 128 used))-----

Logical Drive Name	LUN	Controller	Accessible by	Logical Drive status
Access Logical Drive	31	A,B	Host Group ISR	Optimal
Array_1_R6_A	0	A	Host Group ISR	Optimal
Array_2_R5_A	4	A	Host Group ISR	Optimal
Access Logical Drive	31	A,B	Storage Subsystem	Optimal

TOPOLOGY DEFINITIONS

STORAGE SUBSYSTEM

Default type: Windows 2000/Server 2003/Server 2008 Non-Clustered
Host Port: 21:01:00:e0:8b:a3:1e:cc
Host Port: 21:00:00:e0:8b:95:f8:1f
Host Port: 21:00:00:e0:8b:95:17:1f
Host Port: 21:01:00:e0:8b:b5:0f:20

Default Group

Host Group: MPIO
Host: PineNeedles
Global host type*: Windows 2000/Server 2003/Server 2008 Non-Clustered
Host port identifier: 21:00:00:e0:8b:95:bb:1e
Alias: PineNeedles_1
Global host type*: Windows 2000/Server 2003/Server 2008 Non-Clustered
Host port identifier: 21:01:00:e0:8b:b5:bb:1e
Alias: PineNeedles_2

Host Group: RDAC
Host: Pinehurst
Global host type*: Windows 2000/Server 2003/Server 2008 Non-Clustered
Host port identifier: 10:00:00:00:c9:5f:f8:21
Alias: Pinehurst_1
Global host type*: Windows 2000/Server 2003/Server 2008 Non-Clustered
Host port identifier: 10:00:00:00:c9:5f:f8:22
Alias: Pinehurst_2

Host Group: ISR

```

Host:                ISR-B
Host type:           Windows 2000/Server 2003/Server 2008 Non-Clustered
Host port identifier: 21:00:00:c0:dd:0d:a9:b4
Alias:              ISR-B

```

* Because you are not using storage partitions, all hosts attached to this storage subsystem must have the same host type (operating system). The host type shown is the global setting for all hosts. If this is not the correct host type, you can change it using the Change Default Host Type option. You only need to change it once for all hosts.

NVSRAM HOST TYPE DEFINITIONS

HOST TYPE	ADT STATUS	ASSOCIATED INDEX
AIX	Disabled	6
AIX-ADT/AVT	Enabled	4
DEFAULT	Disabled	0
HP-UX	Enabled	7
IBM TS SAN VCE	Enabled	12
Irix	Disabled	10
LNXCLVMWARE	Disabled	13
Linux	Enabled	5
Netware Failover	Enabled	11
Solaris (with Veritas DMP)	Enabled	14
Solaris (with or without MPXIO)	Disabled	8
Unused1	Disabled	1
Windows 2000/Server 2003/Server 2008 Clustered	Disabled	3
Windows 2000/Server 2003/Server 2008 Clustered (supports DMP)	Enabled	15
Windows 2000/Server 2003/Server 2008 Non-Clustered	Disabled	2 (Default)
Windows 2000/Server 2003/Server 2008 Non-Clustered (supports DMP)	Enabled	9

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 module. You can produce a similar table for a controller in slot B, or you can produce a table for both controllers.

Attention: Incorrectly setting the NVSRAM can severely affect the performance and the availability of the subsystems. You should only modify the NVSRAM settings according to the latest information documented for your configuration in DS4000 publications, readmes, or by your IBM support representative.

```

Controller "a":
Region 226:
  0000: 4942 4d20 2020 2020 3138 3135 2020 2020      IBM.....1815....
  0010: 2020 4641 5374 5420 3039 3134 0000 0000      ..FAST.0914....
  0020: 0000 0000 0000 0000 0000 0000 0000 0000      .....
  0030: 0000 0008 5600 0000 0000 0000 0000 0000      ....V.....

Region 232:
  0000: 0000 0000 0000 0000 0000 0000 0000 0000      .....
  0010: 5500 0000 0000 0000 0000 0000 0000 0000      U.....

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 0300 0000 .....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

```

Region 238:
0000: 0000 d032 0000 0000 0050 0600 0000 0000 ...2....P.....
0010: 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
0020: 0000 0000 0000 0000 cc8c 008a 0029 fe00 .....???.?....
0030: 80be 9f45 7300 0000 0f00 1400 0000 0000 ???Es.....

```

```

Region 240:
0000: fb01 80fa 0101 0104 ffff 0000 ab04 bebe ..?.....?..?
0010: 00f2 ac04 bebe 00ee 0c0d 656c 7669 7261 ..?..?.elvira
0020: 5f34 3530 305f 62a0 0665 6c76 6972 61a1 .4500.b?.elvira?
0030: 0961 6e6f 6e79 6d6f 7573 a204 626c 6168 .anonymous?.blah
0040: ff00 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 .....
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 d032 0000 0000 0050 0600 0000 0000 ...2....P.....
0010: 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
0020: 0000 0000 0000 0000 cc8c 008a 0029 fe00 .....???.?....
0030: 80be 9f45 7300 0000 0f00 1400 0000 0000 ???Es.....
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: 4445 4641 554c 5400 0000 0000 0000 0000 DEFAULT.....
0010: 0020 0000 0100 0001 0000 0000 0000 0000 .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

```

Region 242:
0000: 556e 7573 6564 3100 0000 0000 0000 0000  Unused1.....
0010: 0000 0000 0000 0000 0000 0000 0000 0000  .....
0020: 0000 0000 0000 0008 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 0001 0000 0000 0000 0000  .....
0020: 0001 0100 0000 0008 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 0001 0001 0001 0000 0000  .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000  .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000  .....

```

```

Region 242:
0000: 4149 582d 4144 542f 4156 5400 0000 0000  AIX.ADT.AVT....
0010: 0100 0000 0100 0001 0100 0000 0001 0000  .....
0020: 0000 0100 0100 0008 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: 017f 0000 0100 0001 0001 0000 0000 0000  .....
0020: 0000 0100 0100 0008 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: 0100 0000 0100 0001 0100 0000 0001 0000  .....
0020: 0000 0100 0000 0008 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: 017f 0000 0100 0001 0101 0001 0101 0001  .....
0020: 0100 0000 0100 001c 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 0001 0001 0000 0000 0000  .....
0020: 0000 0000 0000 000a 0000 0000 0000 0000  .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000  .....

```

```

Region 242:
0000: 5732 4b4e 4554 4e43 4c44 4d50 0000 0000  W2KNETNCLDMP...
0010: 0020 0000 0100 0001 0000 0100 0000 0000  .....
0020: 0000 0100 0180 0008 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 2001 0000 0000 0000 0000  .....
0020: 0000 0100 0000 0008 0000 0000 0000 0000  .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000  .....

```

```

Region 242:
0000: 4e57 5246 4f00 0000 0000 0000 0000 0000  NWRFO.....
0010: 007f 0000 0100 0001 0000 0000 0000 0000  .....
0020: 0100 0100 0100 0008 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 0001 0000 0000 0000 0000 .....
0020: 0001 0100 0000 0008 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 0001 0001 0001 0000 0000 .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4149 582d 4144 542f 4156 5400 0000 0000 AIX.ADT.AVT....
0010: 0100 0000 0100 0001 0100 0000 0001 0000 .....
0020: 0000 0100 0100 0008 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: 017f 0000 0100 0001 0001 0000 0000 0000 .....
0020: 0000 0100 0100 0008 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: 0100 0000 0100 0001 0100 0000 0001 0000 .....
0020: 0000 0100 0000 0008 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: 017f 0000 0100 0001 0101 0001 0101 0001 .....
0020: 0100 0000 0100 001c 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 0001 0001 0000 0000 0000 .....
0020: 0000 0000 0000 000a 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 5732 4b4e 4554 4e43 4c44 4d50 0000 0000 W2KNETNCLDMP....
0010: 0020 0000 0100 0001 0000 0100 0000 0000 .....
0020: 0000 0100 0180 0008 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 2001 0000 0000 0000 0000 .....
0020: 0000 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4e57 5246 4f00 0000 0000 0000 0000 0000 NWRFO.....
0010: 007f 0000 0100 0001 0000 0000 0000 0000 .....
0020: 0100 0100 0100 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4942 4d20 5453 2053 414e 2056 4345 0000 IBM.TS.SAN.VCE..
0010: 0000 0000 0100 0000 0000 0001 0000 0000 .....
0020: 0000 0100 0100 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4c4e 5843 4c56 4d57 4152 4500 0000 0000 LNXCLVMWARE....

```

```

0010: 017f 0000 0100 0001 0001 0001 0000 0000 .....
0020: 0000 0100 0000 0008 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 0001 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: 5732 4b4e 4554 434c 444d 5000 0000 0000 W2KNETCLDMP.....
0010: 0020 0000 0100 0001 0001 0101 0000 0000 .....
0020: 0000 0101 0180 0008 0000 0000 0000 0000 .....?.....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

```

Controller "b":
Region 226:
0000: 4942 4d20 2020 2020 3138 3135 2020 2020 IBM.....1815....
0010: 2020 4641 5374 5420 3039 3134 0000 0000 ..FASTT.0914....
0020: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0030: 0000 0008 5600 0000 0000 0000 0000 0000 ....V.....

```

```

Region 232:
0000: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0010: 5500 0000 0000 0000 0000 0000 0000 0000 U.....

```

```

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 0700 0000 .....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

```

Region 238:
0000: 0000 d032 0000 0000 0050 0600 0000 0000 ...2.....P.....
0010: 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
0020: 0000 0000 0000 0000 cc8c 008a 0029 fe00 .....???.?....
0030: 80be 9f45 7300 0000 0f00 1400 0000 0000 ???Es.....

```

```

Region 240:
0000: fb01 80fa 0101 0104 ffff 0000 ab04 bebe ..?...?..??

```

```

0010: 00f3 ac04 bebe 00ee 0c0d 656c 7669 7261 ..?.??....elvira
0020: 5f34 3530 305f 61a0 0665 6c76 6972 61a1 .4500.a?.elvira?
0030: 0961 6e6f 6e79 6d6f 7573 a204 626c 6168 .anonymous?.blah
0040: ff00 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 .....
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 d032 0000 0000 0050 0600 0000 0000 ...2....P.....
0010: 0000 0000 0000 0000 f001 0000 8480 0000 .....??..
0020: 0000 0000 0000 0000 cc8c 008a 0029 fe00 .....???.?....
0030: 80be 9f45 7300 0000 0f00 1400 0000 0000 ???Es.....
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: 4445 4641 554c 5400 0000 0000 0000 0000 DEFAULT.....
0010: 0020 0000 0100 0001 0000 0000 0000 0000 .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 556e 7573 6564 3100 0000 0000 0000 0000 Unused1.....
0010: 0000 0000 0000 0000 0000 0000 0000 0000 .....
0020: 0000 0000 0000 0008 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 0001 0000 0000 0000 0000 .....
0020: 0001 0100 0000 0008 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 0001 0001 0001 0000 0000 .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4149 582d 4144 542f 4156 5400 0000 0000 AIX.ADT.AVT.....
0010: 0100 0000 0100 0001 0100 0000 0001 0000 .....
0020: 0000 0100 0100 0008 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: 017f 0000 0100 0001 0001 0000 0000 0000 .....
0020: 0000 0100 0100 0008 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: 0100 0000 0100 0001 0100 0000 0001 0000 .....

```

```

0020: 0000 0100 0000 0008 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: 017f 0000 0100 0001 0101 0001 0101 0001 .....
0020: 0100 0000 0100 001c 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 0001 0001 0000 0000 0000 .....
0020: 0000 0000 0000 000a 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 0001 0000 0000 0000 0000 .....
0020: 0001 0100 0000 0008 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 0001 0001 0001 0000 0000 .....
0020: 0001 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 5732 4b4e 4554 4e43 4c44 4d50 0000 0000 W2KNETCLDMP....
0010: 0020 0000 0100 0001 0000 0100 0000 0000 .....
0020: 0000 0100 0180 0008 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 2001 0000 0000 0000 0000 .....
0020: 0000 0100 0000 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4e57 5246 4f00 0000 0000 0000 0000 0000 NWRFO.....
0010: 007f 0000 0100 0001 0000 0000 0000 0000 .....
0020: 0100 0100 0100 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4942 4d20 5453 2053 414e 2056 4345 0000 IBM.TS.SAN.VCE..
0010: 0000 0000 0100 0000 0000 0000 0001 0000 .....
0020: 0000 0100 0100 0008 0000 0000 0000 0000 .....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

Region 242:
0000: 4c4e 5843 4c56 4d57 4152 4500 0000 0000 LNXCLVMWARE.....
0010: 017f 0000 0100 0001 0001 0001 0000 0000 .....
0020: 0000 0100 0000 0008 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 0001 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: 5732 4b4e 4554 434c 444d 5000 0000 0000 W2KNETCLDMP....

```

```

0010: 0020 0000 0100 0001 0001 0101 0000 0000 .....
0020: 0000 0101 0180 0008 0000 0000 0000 0000 .....?.....
0030: 0000 0000 0000 0000 0000 0000 0000 0000 .....

```

Show Drive

The show drive command returns information about the disk drives in a storage subsystem.

DRIVES-----

SUMMARY

```

Number of drives: 32
Current drive types: Fibre (32)

```

BASIC:

TRAY, SLOT	STATUS	CAPACITY	CURRENT DATA RATE	PRODUCT ID	FIRMWARE VERSION
1, 1	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 2	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 3	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 4	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 5	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 6	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 7	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
1, 8	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 1	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 2	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 3	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 4	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 5	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 6	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 7	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972
2, 8	Optimal	279.397 GB	4 Gbps	ST3300655FC	F B972

DRIVE CHANNELS:

TRAY, SLOT	PREFERRED CHANNEL	REDUNDANT CHANNEL
1, 1	2	3
1, 2	3	2
1, 3	2	3
1, 4	3	2
1, 5	2	3
1, 6	3	2
1, 7	2	3
1, 8	3	2
2, 1	2	3
2, 2	3	2
2, 3	2	3
2, 4	3	2
2, 5	2	3
2, 6	3	2
2, 7	2	3
2, 8	3	2

HOT SPARE COVERAGE:

```

The following arrays are not protected: 5, 4, 1, 2, 3
Total hot spare drives: 0
Standby: 0
In use: 0

```

DETAILS

Drive at Enclosure 1, Slot 1

```
Status: Optimal
```

Mode: Assigned
Raw capacity: 279.397 GB
Usable capacity: 278.897 GB
World-wide name: 20:00:00:18:62:1c:ba:4e
Associated array: 1

PORT	CHANNEL	ID
0	2	0/0xEF
1	3	0/0xEF

Drive path redundancy: OK
Drive type: Fibre Channel

Speed: 15015 RPM
Current data rate: 4 Gbps
Product ID: ST3300655FC F
Firmware version: B972
Serial number: 3LM083JW00009714C38H
Vendor: IBM-SSG
Date of manufacture: Not Available

Drive at Enclosure 1, Slot 2

Status: Optimal

Mode: Assigned
Raw capacity: 279.397 GB
Usable capacity: 278.897 GB
World-wide name: 20:00:00:18:62:1c:ba:5b
Associated array: 1

PORT	CHANNEL	ID
0	3	1/0xE8
1	2	1/0xE8

Drive path redundancy: OK
Drive type: Fibre Channel

Speed: 15015 RPM
Current data rate: 4 Gbps
Product ID: ST3300655FC F
Firmware version: B972
Serial number: 3LM083KK00008708DEQ0
Vendor: IBM-SSG
Date of manufacture: Not Available

Drive at Enclosure 1, Slot 3

Status: Optimal

Mode: Assigned
Raw capacity: 279.397 GB
Usable capacity: 278.897 GB
World-wide name: 20:00:00:18:62:1c:ba:4b
Associated array: 1

PORT	CHANNEL	ID
0	2	2/0xE4
1	3	2/0xE4

Drive path redundancy: OK
Drive type: Fibre Channel

Speed: 15015 RPM
Current data rate: 4 Gbps
Product ID: ST3300655FC F

Firmware version: B972
Serial number: 3LM083JV00009714C38E
Vendor: IBM-SSG
Date of manufacture: Not Available

Show Drive Channel Status

The show drive channel stat command returns information about the drive channels in a storage subsystem. Use this information to determine how well the channels are running and errors that might be occurring on the channels.

DRIVE CHANNELS-----

SUMMARY

CHANNEL	STATUS	CTRL A LINK	CTRL B LINK
1	Optimal	Up	Up
2	Optimal	Up	Up
3	Optimal	Up	Up
4	Optimal	Up	Up

DETAILS

DRIVE CHANNEL 1

Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 4 Gbps
Rate Control: Auto
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 0

CUMULATIVE ERROR COUNTS

Controller A
Baseline time set: 3/18/08 3:13:48 PM
Sample period (hh:mm:ss): 23:23:50
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 1
Link down errors: 4
Total I/O count: 33607764

Controller B
Baseline time set: 3/18/08 3:15:16 PM
Sample period (hh:mm:ss): 23:22:22
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 0
Total I/O count: 33826277

DRIVE CHANNEL 2

Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 4 Gbps
Rate Control: Auto
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 32
Connected to: Controller A, Port 1
Attached drives: 32
Drive enclosure: 2 (16 drives)
Drive enclosure: 1 (16 drives)

CUMULATIVE ERROR COUNTS

Controller A
Baseline time set: 3/18/08 3:13:50 PM
Sample period (hh:mm:ss): 23:23:48
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 1
Link down errors: 4
Total I/O count: 43656478

Controller B
Baseline time set: 3/18/08 3:15:18 PM
Sample period (hh:mm:ss): 23:22:20
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 0
Total I/O count: 41525248

DRIVE CHANNEL 3

Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 4 Gbps
Rate Control: Auto
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 32
Connected to: Controller B, Port 1
Attached drives: 32
Drive enclosure: 2 (16 drives)
Drive enclosure: 1 (16 drives)

CUMULATIVE ERROR COUNTS

Controller A
Baseline time set: 3/18/08 3:13:48 PM
Sample period (hh:mm:ss): 23:23:50
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 1
Link down errors: 5
Total I/O count: 41116049

Controller B
Baseline time set: 3/18/08 3:15:18 PM
Sample period (hh:mm:ss): 23:22:20
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 0
Total I/O count: 43091729

DRIVE CHANNEL 4

Status: Optimal
Max. Rate: 4 Gbps
Current Rate: 2 Gbps
Rate Control: Auto
Controller A link status: Up
Controller B link status: Up

DRIVE COUNTS

Total # of attached drives: 0

CUMULATIVE ERROR COUNTS

Controller A
Baseline time set: 3/18/08 3:13:49 PM
Sample period (hh:mm:ss): 23:23:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 6
Total I/O count: 33607757

Controller B
Baseline time set: 3/18/08 3:15:16 PM
Sample period (hh:mm:ss): 23:22:22
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 0
Total I/O count: 33826278

Script execution complete.

Show Logical Drives

The show logicalDrives command returns information about the logical drives in a storage subsystem.

STANDARD LOGICAL DRIVES-----

SUMMARY

Number of standard logical drives: 6

See other Logical Drives sub-tabs for premium feature information.

NAME	STATUS	CAPACITY	RAID LEVEL	ARRAY
E_Drive	Optimal	1,115.586 GB	5	1
K_Drive	Optimal	1,115.586 GB	5	2
Q_Drive	Optimal	1,115.586 GB	5	3
Test1	Optimal	2,035.998 GB	0	5
Test2	Optimal	1,589.656 GB	0	5
Z_Drive	Optimal	836.689 GB	5	4

DETAILS

LOGICAL DRIVE NAME: E_Drive

Logical Drive status: Optimal

Capacity: 1,115.586 GB (1,197,851,279,360 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:e6:00:00:01:d8:47:df:c6:e8
Subsystem ID (SSID): 0

Associated array: 1
RAID level: 5

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot A
Current owner: Controller in slot A

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

LOGICAL DRIVE NAME: K_Drive

Logical Drive status: Optimal

Capacity: 1,115.586 GB (1,197,851,279,360 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:ea:00:00:10:b9:47:df:c5:57
Subsystem ID (SSID): 1
Associated array: 2
RAID level: 5

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot B
Current owner: Controller in slot B

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

LOGICAL DRIVE NAME: Q_Drive

Logical Drive status: Optimal

Capacity: 1,115.586 GB (1,197,851,279,360 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:e6:00:00:01:e2:47:df:c7:38
Subsystem ID (SSID): 2
Associated array: 3
RAID level: 5

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot A
Current owner: Controller in slot A

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

LOGICAL DRIVE NAME: Test1

Logical Drive status: Optimal

Capacity: 2,035.998 GB (2,186,136,256,512 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:e6:00:00:01:ec:47:df:cb:6e
Subsystem ID (SSID): 4
Associated array: 5
RAID level: 0

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot A
Current owner: Controller in slot A

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

LOGICAL DRIVE NAME: Test2

Logical Drive status: Optimal

Capacity: 1,589.656 GB (1,706,880,401,408 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:ea:00:00:10:cf:47:df:c9:cb
Subsystem ID (SSID): 5
Associated array: 5
RAID level: 0

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot B
Current owner: Controller in slot B

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

LOGICAL DRIVE NAME: Z_Drive

Logical Drive status: Optimal

Capacity: 836.689 GB (898,388,459,520 Bytes)
Logical Drive ID: 60:0a:0b:80:00:11:59:ea:00:00:10:c4:47:df:c5:b7
Subsystem ID (SSID): 3
Associated array: 4
RAID level: 5

Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot B
Current owner: Controller in slot B

Segment size: 128 KB
Modification priority: High

Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Enabled
Flush write cache after (in seconds): 10.00
Dynamic cache read prefetch: Enabled

Enable background media scan: Enabled
Media scan with redundancy check: Disabled

MISSING LOGICAL DRIVES-----

Number of missing logical drives: 0

Chapter 10. Example Script Files

This appendix provides example scripts for configuring a storage subsystem. 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 subsystem.

You can create a script file in two ways:

- Using the `save storageSubsystem` configuration command
- Writing a script

By using the `save storageSubsystem` configuration command, you can create a file that you can use to copy an existing configuration from one storage subsystem to other storage subsystems. 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 that are described in Chapter 2, “About the Script Commands,” on page 11 and the command formatting rules that are listed in “Formatting Rules for Script Commands” on page 20. 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:\...\...\client>smcli 123.45.67.89 -f scriptfile.scr;
```

Configuration Script Example 1

This example creates a new logical drive by using the `create logicalDrive` command in the free space of an array.

```
Show "Create RAID 5 Logical Drive 7 on existing Array 1";
//Create logical drive on array created by the create logical drives command
//Note: For arrays that use all available capacity, the last logical drive
on the group is created using all remaining capacity by omitting
the capacity=logical drive creation parameter
create logicalDrive array=1 RAIDLevel=5 userLabel="7" owner=A segmentSize=16
cacheReadPrefetch=TRUE capacity=2GB;
show "Setting additional attributes for logical drive 7"; //Configuration settings
that cannot be set during logical drivecreation
set logicalDrive ["7"] cacheFlushModifier=10;
set logicalDrive ["7"] cacheWithoutBatteryEnabled=false;
set logicalDrive ["7"] mirrorEnabled=true;
set logicalDrive ["7"] readCacheEnabled=true;
set logicalDrive ["7"] writeCacheEnabled=true;
set logicalDrive ["7"] mediaScanEnabled=false;
set logicalDrive ["7"] redundancyCheckEnabled=false;
set logicalDrive ["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 Logical Drive 7 on existing Array 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 logical drive by using the create logicalDrive command on an existing array.

The line beginning **//Note:** is a comment in the script file that explains that the size of the last logical drive created that uses all of the available capacity because the **capacity** parameter is not used.

The command in this example creates a new logical drive in array 1. The logical drive has a RAID level of 5. The logical drive 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 logical drive is assigned to the controller in slot A in the controller module. The segment size is set to 16. The logical drive has a read ahead multiplier value of 256. The capacity of the logical drive is 2 GB.

The command takes this form:

```
create logicalDrive array=arrayNumber userLabel=logicalDriveName
[freeCapacityArea=freeCapacityIndexNumber] [capacity=logicalDriveCapacity
| owner=(a | b) | cacheReadPrefetch=(TRUE | FALSE) | segmentSize=segmentSizeValue]
[enclosureLossProtect=(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 logical drive 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 logicalDrive command ran successfully and that properties that could not be set by the create logicalDrive command are now set.

The set logicalDrive 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 logicalDrive command by leaving a space between the parameters, as in this example:

```
set logicalDrive["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 logical drive by using the create logicalDrive command with user-defined disk drives in the storage subsystem.

```
Show "Create RAID3 Logical Drive 2 on existing Array 2";
//This command creates the array and the initial logical drive on that array.
//Note: For arrays that use all available capacity, the last logical drive on the array is created
using all remaining capacity by omitting the capacity=logical drive creation parameter
create logicalDrive RAIDLevel=3 userLabel="2" drives=[0,1 0,6 1,7 1,3 2,3 2,6]
owner=B segmentSize=16 capacity=2GB;
show "Setting additional attributes for logical drive 7"//Configuration settings that
cannot be set during logical drive creation
set logicalDrive ["7"] cacheFlushModifier=10;
set logicalDrive ["7"] cacheWithoutBatteryEnabled=false;
```

```
set logicalDrive ["7"] mirrorEnabled=true;
set logicalDrive ["7"] readCacheEnabled=true;
set logicalDrive ["7"] writeCacheEnabled=true;
set logicalDrive ["7"] mediaScanEnabled=false;
set logicalDrive ["7"] redundantCheckEnabled=false;
set logicalDrive ["7"] modificationPriority=high;
```

The command in this example, like the create logicalDrive command in the previous example, creates a new logical drive. The significant difference between these two examples is that this example shows how you can define specific disk drives to include in the logical drive. Use the show storageSubsystem profile command to find out what disk drives are available in a storage subsystem.

The create logicalDrive command takes this form:

```
create logicalDrive raidLevel=(0 | 1 | 3 | 5 | 6) userLabel=logicalDriveName
drives=(enclosureID1,slotID1..enclosureIDn,slotIDn) [capacity=logicalDriveCapacity | owner=(a | b) |
cacheReadPrefetch=(TRUE | FALSE) | segmentSize=segmentSizeValue] [enclosureLossProtect=(TRUE | FALSE)]
```


Chapter 11. Deprecated Commands and Parameters

This appendix lists the commands, command formats, and parameters that are no longer supported by this level of software. The information is presented in two tables. “Deprecated Commands” lists commands no longer supported in this level of software and the new commands that replaced them. “Deprecated Parameters” on page 261 lists the parameters 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
accept storageSubsystem pendingTopology (allHosts host [<i>user-label</i>] hosts [<i>user-label</i>])	Removed.
create logicalDrive (drive drives) [<i>enclosureID1,slotID1...enclosureIDn,slotIDn</i>]	create logicalDrive drives=(<i>enclosureID1,slotID1...enclosureIDn,slotIDn</i>) The new syntax for specifying disk drives requires an equal sign (=) after the drives parameter.
create logicalDrive driveCount [<i>numberOfDrives</i>]	create logicalDrive driveCount= <i>numberOfDrives</i> The new syntax for specifying the number of disk drives requires an equal sign (=) after the driveCount parameter.
create logicalDrive logicalDriveGroup [<i>numberOfDrives</i>]	create logicalDrive logicalDriveGroup= <i>subsystemNumber</i> The new syntax for specifying the Subsystem number requires an equal sign (=) after the subsystem parameter.
download storageSubsystem (firmwareFile NVSRAMFile)= <i>filename</i>	download drive [<i>enclosureID,slotID</i>] firmware file=" <i>filename</i> " download storageSubsystem firmware [, NVSRAM] file=" <i>filename</i> " [, " <i>NVSRAM-filename</i> "] [downgrade=(TRUE FALSE)] [activateNow=(TRUE FALSE)] The new version of the storage management software provides unique commands to perform these functions.
set controller [(a b)] NVSRAMByte [<i>nvsram-offset</i>]=(<i>nvsramByteSetting</i> <i>nvsramBitSetting</i>)	set controller [(a b)] globalNVSRAMByte [<i>nvsramOffset</i>]=(<i>nvsramByteSetting</i> <i>nvsramBitSetting</i>) This new command provides additional parameters for setting the NVSRAM values.
show storageSubsystem pendingTopology	Removed.

Table 26. Deprecated Commands (continued)

Deprecated Command	New Command
show storageSubsystem preferredLogicalDriveOwners	show storageSubsystem profile This command, with the profile parameter, returns information about the preferred logical drive owner.
show logicalDrives logicalDrive [userLabel]	show storageSubsystem profile This command, with the profile parameter, returns information about the logical drive.
download enclosure [0]	download allenclosures firmware file="filename" When you download ESM firmware to all of the drive expansion enclosures, in the previous command "all modules" was defined by entering [0]. The new command uses the allenclosures parameter.
capacity=logicalDriveCapacity raidLevel=(0 1 3 5) userLabel= logicalDriveName owner=(a b) usageHint=usageHintSpec segmentSize= segmentSizeValue readAheadMultiplier= multiplierValue	capacity=logicalDriveCapacity owner=(a b) cacheReadPrefetch=(TRUE FALSE) segmentSize=segmentSizeValue In the create RAID logicalDrive commands, the new commands use fewer optional attribute=value pairs.
create mapping logicalDrive=userLabel logicalGroupNumber=logicalGroupNumber [host hostGroup]=hostName hostGroupName	Use the set logicalDrive command to define logical drive-to-LUN mapping. See "Set Logical Drive" on page 93.
delete mapping logicalDrive=userLabel [host hostGroup]=hostName hostGroupName	Use the remove logicalDrive LUNMapping command to remove logical drive-to-LUN mapping. See "Remove Logical Drive LUN Mapping" on page 68.
download storageSubsystem file=filename content=firmware [downgrade=(TRUE FALSE)]	Use the download storageSubsystem firmware command to download the firmware. See "Download Storage Subsystem Firmware/NVSRAM" on page 61.
download storageSubsystem file=filename content=NVSRAM	Use the download storageSubsystem NVSRAM command to download the NVSRAM values. See "Download Storage Subsystem NVSRAM" on page 62.
download drive [enclosureID,slotID] file=filenamecontent=(firmware modePage)	Use the download storageSubsystem driveFirmware command to download the firmware images to all of the disk drives in the storage subsystem. See "Download Storage Subsystem Drive Firmware" on page 61.
download (allenclosures enclosure [enclosureID]) file=filename content=firmware	Use the download (environmental card) firmware command to download the enclosure firmware. See "Download Environmental Card Firmware" on page 60.
download storageSubsystem file=filename content=featureKey	Use the enable storageSubsystem feature command to enable a premium feature. See "Enable Storage Subsystem Feature Key" on page 63.

Table 26. *Deprecated Commands (continued)*

Deprecated Command	New Command
remove copyEntry target [<i>targetName</i>] [source [<i>sourceName</i>]]	Use the remove volumeCopy command to remove VolumeCopy entries. “Remove Volume Copy” on page 69.
remove logicalDriveReservations (allLogicalDrives logicalDrive [<i>logicalDriveName</i>])	Use the clear logicalDrive command to clear persistent logicalDrive reservations. See “Clear Logical Drive Reservations” on page 37.
set controller [(a b)] batteryInstallDate=(TRUE FALSE)	Use the reset storageSubsystem batteryInstallDate command to reset the battery date. See “Reset Storage Subsystem Battery Install Date” on page 71.
set controller [(a b)] serviceMode=(TRUE FALSE)	Use the set controller availability=serviceMode command to place the storage subsystem in Service mode. See “Set Controller Service Action Allowed Indicator” on page 84.
set drive [<i>enclosureID,slotID</i>] operationalState=(optimal failed)	Use the set drive operationalState=failed command to place a disk drive in the storage subsystem in Failed mode. See “Set Drive State” on page 86. To return a disk drive to the Optimal state, use the revive drive command. See “Revive Drive” on page 74.
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. See “Set Session” on page 99.
set storageSubsystem batteryInstallDate=(TRUE FALSE)	Use the reset storageSubsystem batteryInstallDate command to reset the battery date. See “Reset Storage Subsystem Battery Install Date” on page 71.
set storageSubsystem clearEventLog=(TRUE FALSE)	Use the clear storageSubsystem eventLog command to clear the Event Log for the storage subsystem. See “Clear Storage Subsystem Event Log” on page 38.
set storageSubsystem resetConfiguration=(TRUE FALSE)	Use the clear storageSubsystem configuration command to clear the entire configuration from the controllers in a storage subsystem. See “Clear Storage Subsystem Configuration” on page 38.
set storageSubsystem RLSBaseline=currentTime	Use the reset storageSubsystem RLSBaseline command to reset the Read Link Status (RLS) baseline for all of the devices. See “Reset Storage Subsystem RLS Baseline” on page 73.
set storageSubsystem dayOfTime=(TRUE FALSE)	Use the set storageSubsystem time command to set the clocks on both of the controllers in a storage subsystem to the clock of the host. See “Set Storage Subsystem Time” on page 106.
set logicalDrive [<i>logicalDriveName</i>] mirrorEnabled=(TRUE FALSE)	Use the set logicalDrive command with mirror cache enabled. See “Set Logical Drive” on page 93.
set volumeCopy target [<i>targetName</i>] [source [<i>sourceName</i>]] priority=(lower low medium high highest)	Use the set volumeCopy command to define the VolumeCopy pair. See “Set VolumeCopy” on page 106.
set logicalDriveLabel ID [<i>hexValue</i>] userLabel= <i>logicalDriveName</i>	Use the set logicalDrive command to define a user name for a logical drive. See “Set Logical Drive” on page 93.

Table 26. *Deprecated Commands (continued)*

Deprecated Command	New Command
show hostTopology	Use the show storageSubsystem 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 subsystem. See "Show Storage Subsystem Host Topology" on page 117.
show logicalDriveReservations (allLogicalDrives logicalDrive <i>logicalDriveName</i>)	Use the show logicalDrive reservations command to return information about the logical drives that have reservations. See "Show Logical Drive Reservations" on page 115.
disableFlashCopy logicalDrive	Use the stop FlashCopy command to stop a copy-on-write operation. See "Stop FlashCopy" on page 127.
recreateFlashCopy logicalDrive	Use the recreate FlashCopy command to start a fresh copy-on-write operation by using an existing FlashCopy logical drive. See "Re-create FlashCopy" on page 67.
start increaseVolCapacity logicalDrive= <i>logicalDriveName</i> incrementalCapacity= <i>capacityValue</i> drives=(<i>enclosureID1</i> , <i>slotID1</i> ... <i>enclosureIDn</i> , <i>slotIDn</i>)	Use the set logicalDrive command to define values for increasing the capacity of a logical drive. See "Set Logical Drive" on page 93.
start volumeCopy source=" <i>sourceName</i> " target=" <i>targetName</i> " copyPriority=(lowest low medium high highest)	Use the create volumeCopy command to create a VolumeCopy pair and to start the VolumeCopy. See "Create Volume Copy" on page 54.
on error (stop continue)	Use the set session command to define how the CLI session responds to errors. See "Set Session" on page 99.
use password " <i>storageSubsystemPassword</i> "	Use the set session command to define a password for the storage subsystem. See "Set Session" on page 99.
upload storageSubsystem file= <i>filename</i> content=configuration	Use the save configuration command to save a storage subsystem configuration. See "Save Storage Subsystem Configuration" on page 75.
upload storageSubsystem file= <i>filename</i> content=(allEvents criticalEvents)	Use the save storageSubsystem (allEvents criticalEvents) command to save events to a file. See "Save Storage Subsystem Events" on page 77.
upload storageSubsystem file= <i>filename</i> content=performanceStats	Use the save storageSubsystem performanceStats command to save the performance statistics to a file. See "Save Storage Subsystem Performance Statistics" on page 78.
upload storageSubsystem file= <i>filename</i> content=RLSCounts	Use the save storageSubsystem RLSCounts command to save the RLS counters to a file. See "Save Storage Subsystem RLS Counts" on page 78.
upload storageSubsystem file= <i>filename</i> content=stateDump	Use the save storageSubsystem stateCapture command to save state dumps to a file. See "Save Storage Subsystem State Capture" on page 79.

Deprecated Parameters

Table 27. *Deprecated Parameters*

Old Syntax	New Syntax
<code>clearEventLog</code>	<code>clear storageSubsystem eventLog</code>
<code>copyEntry</code>	<code>volumeCopy</code>
<code>database</code>	Removed
<code>disableFlashcopy</code>	<code>stop flashcopy</code>
<code>enforceSoftLimit</code>	Removed
<code>featureKey</code>	<code>feature</code>
<code>filesystem</code>	Removed
<code>gatewayIPAddress</code>	<code>IPv4GatewayIP</code>
<code>hostType</code>	Removed from the create host port command and the set host port command.
<code>id[]</code>	<code>logicalDrive<></code>
<code>increaseVolCapacity</code>	<code>set logicalDrive addCapacity</code>
<code>incrementalCapacity</code>	<code>addCapacity</code>
<code>ipAddress</code>	<code>IPv4Address</code> or <code>IPv6Address</code>
<code>mapping</code>	<code>lunMapping</code>
<code>modePage</code>	Removed
<code>multimedia</code>	Removed
<code>on error</code>	<code>set session errorAction</code>
<code>performanceMonitor interval</code>	<code>performanceMonitorInterval</code>
<code>performanceMonitor iterations</code>	<code>performanceMonitorIterations</code>
<code>priority</code>	<code>copyPriority</code>
<code>readAheadMultiplier</code>	<code>cacheReadPrefetch</code>
<code>recreateFlashcopy</code>	<code>recreate FlashCopy</code>
<code>resetConfiguration</code>	<code>reset storageSubsystem configuration</code>
<code>stateDump</code>	<code>stateCapture</code>
<code>subnetMask</code>	<code>IPv4SubnetMask</code>
<code>timeOfDay</code>	<code>time</code>
<code>upload</code>	<code>save</code>
<code>use password</code>	<code>set session password</code>
<code>logicalDriveLabel</code>	Removed
<code>logicalDriveReservations</code>	<code>show logicalDrive reservations</code> or <code>reservations</code>

Appendix. Additional Documentation

Finding DS3000, DS4000, and DS5000 documents

The following tables present an overview of the IBM System Storage Storage Manager, Storage Subsystem, and Storage Expansion Enclosure product libraries, as well as other related documents. Each table lists documents that are included in the libraries and what common tasks they address.

You can access the documents listed in these tables at both of the following Web sites:

www.ibm.com/servers/storage/support/disk/

www.ibm.com/shop/publications/order/

DS3000 Storage Subsystem documents

The following table associates each document in the DS3000 Storage Subsystem library with its related common user tasks.

Table 28. DS3000 Storage Subsystem document titles by user tasks

Title	User Tasks					
	Planning	Hardware Installation	Software Installation	Configuration	Operation and Administration	Diagnosis and Maintenance
<i>IBM System Storage DS3000 Storage Subsystem Installation, User's and Maintenance Guides</i>	✓	✓		✓	✓	✓
<i>IBM System Storage DS3000 Quick Start Guides</i>		✓				

DS4000 Storage Subsystem documents

The following table associates each document in the DS4000 Storage Subsystem library with its related common user tasks.

Table 29. DS4000 Storage Subsystem document titles by user tasks

Title	User Tasks					
	Planning	Hardware Installation	Software Installation	Configuration	Operation and Administration	Diagnosis and Maintenance
<i>IBM System Storage DS4000 Storage Subsystem Installation, User's and Maintenance Guides</i>	✓	✓		✓	✓	✓
<i>IBM System Storage DS4000 Quick Start Guides</i>		✓				

DS3000, DS4000. and DS5000 Storage Expansion Enclosure documents

The following table associates each of the following documents with its related common user tasks.

Table 30. Storage Expansion Enclosure document titles by user tasks

Title	User Tasks					
	Planning	Hardware Installation	Software Installation	Configuration	Operation and Administration	Diagnosis and Maintenance
<i>IBM System Storage EXPxxx Storage Expansion Enclosure Installation, User's, and Maintenance Guide</i>	✓	✓		✓	✓	✓
<i>IBM System Storage Quick Start Guide, Quick Reference for the DS4700 and DS4200, Sections 2, 3, and 4 also for installing the EXP810 and EXP420</i>		✓	✓	✓		
<i>IBM System Storage Hard Drive and Storage Expansion Enclosures Installation and Migration Guide</i>	✓	✓				
<i>IBM System Storage Quick Start Guide, Quick Reference for the DS5100 and DS5300, and the EXP5000</i>		✓	✓	✓		

Other DS3000 and DS4000-related documents

The following table associates each of the following documents with its related common user tasks.

Table 31. DS3000 and DS4000-related document titles by user tasks

Title	User Tasks					
	Planning	Hardware Installation	Software Installation	Configuration	Operation and Administration	Diagnosis and Maintenance
<i>IBM Safety Information</i>					✓	
<i>IBM TotalStorage® Hardware Maintenance Manual 1</i>						✓
<i>IBM System Storage Problem Determination Guide</i>						✓

Table 31. DS3000 and DS4000–related document titles by user tasks (continued)

Title	User Tasks					
	Planning	Hardware Installation	Software Installation	Configuration	Operation and Administration	Diagnosis and Maintenance
<i>IBM Fibre Channel Planning and Integration: User's Guide and Service Information</i>	✓	✓			✓	✓
<i>IBM TotalStorage FC2-133 Host Bus Adapter Installation and User's Guide</i>		✓			✓	
<i>IBM TotalStorage FC2-133 Dual Port Host Bus Adapter Installation and User's Guide</i>		✓			✓	
<i>IBM Netfinity® Fibre Channel Cabling Instructions</i>		✓				
<i>IBM Fibre Channel SAN Configuration Setup Guide</i>	✓		✓	✓	✓	

Note:

1. The *IBM TotalStorage DS4000 Hardware Maintenance Manual* does not contain maintenance information for the IBM System Storage DS4100, DS4200, DS4300, DS4500, DS4700, or DS4800 storage subsystems. You can find maintenance information for these products in the *IBM System Storage DSx000 Storage Subsystem Installation, User's, and Maintenance Guide* for the particular subsystem.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.*

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan*

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATIONS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publications. IBM may make improvements or changes (or both) in the product(s) or program(s) (or both), described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Some software may differ from its retail version (if available), and may not include user manuals or all program functionality.

Glossary

This glossary provides definitions for the terminology and abbreviations used in IBM System Storage publications.

If you do not find the term you are looking for, see the *IBM Glossary of Computing Terms* located at the following Web site:

<http://www.ibm.com/ibm/terminology>

This glossary also includes terms and definitions from:

- *Information Technology Vocabulary* by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.
- *IBM Glossary of Computing Terms*. New York: McGraw-Hill, 1994.

The following cross-reference conventions are used in this glossary:

See Refers you to (a) a term that is the expanded form of an abbreviation or acronym, or (b) a synonym or more preferred term.

See also
Refers you to a related term.

Abstract Windowing Toolkit (AWT)
A Java graphical user interface (GUI).

accelerated graphics port (AGP)
A bus specification that gives low-cost 3D graphics cards faster access to main memory on personal computers than the usual peripheral component interconnect (PCI) bus. AGP reduces the overall cost of creating high-end graphics subsystems by using existing system memory.

access volume
A special logical drive that allows the

host-agent to communicate with the controllers in the storage subsystem.

adapter
A printed circuit assembly that transmits user data input/output (I/O) between the internal bus of the host system and the external fibre-channel (FC) link and vice versa. Also called an I/O adapter, host adapter, or FC adapter.

advanced technology (AT[®]) bus architecture
A bus standard for IBM compatibles. It extends the XT[™] bus architecture to 16 bits and also allows for bus mastering, although only the first 16 MB of main memory are available for direct access.

agent A server program that receives virtual connections from the network manager (the client program) in a Simple Network Management Protocol-Transmission Control Protocol/Internet Protocol (SNMP-TCP/IP) network-managing environment.

AGP See *accelerated graphics port*.

AL_PA
See *arbitrated loop physical address*.

arbitrated loop
One of three existing fibre-channel topologies, in which 2 - 126 ports are interconnected serially in a single loop circuit. Access to the Fibre Channel Arbitrated Loop (FC-AL) is controlled by an arbitration scheme. The FC-AL topology supports all classes of service and guarantees in-order delivery of FC frames when the originator and responder are on the same FC-AL. The default topology for the disk array is arbitrated loop. An arbitrated loop is sometimes referred to as a Stealth Mode.

arbitrated loop physical address (AL_PA)
An 8-bit value that is used to uniquely identify an individual port within a loop. A loop can have one or more AL_PAs.

array A collection of fibre-channel or SATA hard drives that are logically grouped together. All the drives in the array are assigned the same RAID level. An array is

sometimes referred to as a "RAID set."
See also *redundant array of independent disks (RAID)*, *RAID level*.

asynchronous write mode

In remote mirroring, an option that allows the primary controller to return a write I/O request completion to the host server before data has been successfully written by the secondary controller. See also *synchronous write mode*, *remote mirroring*, *Global Copy*, *Global Mirroring*.

AT See *advanced technology (AT) bus architecture*.

ATA See *AT-attached*.

AT-attached

Peripheral devices that are compatible with the original IBM AT computer standard in which signals on a 40-pin AT-attached (ATA) ribbon cable followed the timings and constraints of the Industry Standard Architecture (ISA) system bus on the IBM PC AT computer. Equivalent to integrated drive electronics (IDE).

auto-volume transfer/auto-disk transfer (AVT/ADT)

A function that provides automatic failover in case of controller failure on a storage subsystem.

AVT/ADT

See *auto-volume transfer/auto-disk transfer*.

AWT See *Abstract Windowing Toolkit*.

basic input/output system (BIOS)

The personal computer code that controls basic hardware operations, such as interactions with diskette drives, hard disk drives, and the keyboard.

BIOS See *basic input/output system*.

BOOTP

See *bootstrap protocol*.

bootstrap protocol (BOOTP)

In Transmission Control Protocol/Internet Protocol (TCP/IP) networking, an alternative protocol by which a diskless machine can obtain its Internet Protocol (IP) address and such configuration information as IP addresses of various servers from a BOOTP server.

bridge A storage area network (SAN) device that provides physical and transport

conversion, such as fibre channel to small computer system interface (SCSI) bridge.

bridge group

A bridge and the collection of devices connected to it.

broadcast

The simultaneous transmission of data to more than one destination.

cathode ray tube (CRT)

A display device in which controlled electron beams are used to display alphanumeric or graphical data on an electroluminescent screen.

client A computer system or process that requests a service of another computer system or process that is typically referred to as a server. Multiple clients can share access to a common server.

command

A statement used to initiate an action or start a service. A command consists of the command name abbreviation, and its parameters and flags if applicable. A command can be issued by typing it on a command line or selecting it from a menu.

community string

The name of a community contained in each Simple Network Management Protocol (SNMP) message.

concurrent download

A method of downloading and installing firmware that does not require the user to stop I/O to the controllers during the process.

CRC See *cyclic redundancy check*.

CRT See *cathode ray tube*.

CRU See *customer replaceable unit*.

customer replaceable unit (CRU)

An assembly or part that a customer can replace in its entirety when any of its components fail. Contrast with *field replaceable unit (FRU)*.

cyclic redundancy check (CRC)

(1) A redundancy check in which the check key is generated by a cyclic algorithm. (2) An error detection technique performed at both the sending and receiving stations.

dac See *disk array controller*.

dar See *disk array router*.

DASD
See *direct access storage device*.

data striping
See *striping*.

default host group
A logical collection of discovered host ports, defined host computers, and defined host groups in the storage-partition topology that fulfill the following requirements:

- Are not involved in specific logical drive-to-LUN mappings
- Share access to logical drives with default logical drive-to-LUN mappings

device type
Identifier used to place devices in the physical map, such as the switch, hub, or storage.

DHCP See *Dynamic Host Configuration Protocol*.

direct access storage device (DASD)
A device in which access time is effectively independent of the location of the data. Information is entered and retrieved without reference to previously accessed data. (For example, a disk drive is a DASD, in contrast with a tape drive, which stores data as a linear sequence.) DASDs include both fixed and removable storage devices.

direct memory access (DMA)
The transfer of data between memory and an input/output (I/O) device without processor intervention.

disk array controller (dac)
A disk array controller device that represents the two controllers of an array. See also *disk array router*.

disk array router (dar)
A disk array router that represents an entire array, including current and deferred paths to all logical unit numbers (LUNs) (hdisks on AIX). See also *disk array controller*.

DMA See *direct memory access*.

domain
The most significant byte in the node port (N_port) identifier for the fibre-channel

(FC) device. It is not used in the fibre channel-small computer system interface (FC-SCSI) hardware path ID. It is required to be the same for all SCSI targets logically connected to an FC adapter.

drive channels
The DS4200, DS4700, and DS4800 subsystems use dual-port drive channels that, from the physical point of view, are connected in the same way as two drive loops. However, from the point of view of the number of drives and enclosures, they are treated as a single drive loop instead of two different drive loops. A group of storage expansion enclosures are connected to the DS4000 storage subsystems using a drive channel from each controller. This pair of drive channels is referred to as a redundant drive channel pair.

drive loops
A drive loop consists of one channel from each controller combined to form one pair of redundant drive channels or a redundant drive loop. Each drive loop is associated with two ports. (There are two drive channels and four associated ports per controller.) For the DS4800, drive loops are more commonly referred to as drive channels. See *drive channels*.

DRAM
See *dynamic random access memory*.

Dynamic Host Configuration Protocol (DHCP)
A protocol defined by the Internet Engineering Task Force that is used for dynamically assigning Internet Protocol (IP) addresses to computers in a network.

dynamic random access memory (DRAM)
A storage in which the cells require repetitive application of control signals to retain stored data.

ECC See *error correction coding*.

EEPROM
See *electrically erasable programmable read-only memory*.

EISA See *Extended Industry Standard Architecture*.

electrically erasable programmable read-only memory (EEPROM)

A type of memory chip which can retain its contents without consistent electrical power. Unlike the PROM which can be

programmed only once, the EEPROM can be erased electrically. Because it can only be reprogrammed a limited number of times before it wears out, it is appropriate for storing small amounts of data that are changed infrequently.

electrostatic discharge (ESD)

The flow of current that results when objects that have a static charge come into close enough proximity to discharge.

environmental service module (ESM) canister

A component in a storage expansion enclosure that monitors the environmental condition of the components in that enclosure. Not all storage subsystems have ESM canisters.

E_port See *expansion port*.

error correction coding (ECC)

A method for encoding data so that transmission errors can be detected and corrected by examining the data on the receiving end. Most ECCs are characterized by the maximum number of errors they can detect and correct.

ESD See *electrostatic discharge*.

ESM canister

See *environmental service module canister*.

automatic ESM firmware synchronization

When you install a new ESM into an existing storage expansion enclosure in a DS4000 storage subsystem that supports automatic ESM firmware synchronization, the firmware in the new ESM is automatically synchronized with the firmware in the existing ESM.

EXP See *storage expansion enclosure*.

expansion port (E_port)

A port that connects the switches for two fabrics.

Extended Industry Standard Architecture (EISA)

A bus standard for IBM compatibles that extends the Industry Standard Architecture (ISA) bus architecture to 32 bits and allows more than one central processing unit (CPU) to share the bus. See also *Industry Standard Architecture*.

fabric A fibre channel entity which interconnects and facilitates logins of N_ports attached to it. The fabric is responsible for routing frames between source and destination

N_ports using address information in the frame header. A fabric can be as simple as a point-to-point channel between two N_ports, or as complex as a frame-routing switch that provides multiple and redundant internal pathways within the fabric between F_ports.

fabric port (F_port)

In a fabric, an access point for connecting a user's N_port. An F_port facilitates N_port logins to the fabric from nodes connected to the fabric. An F_port is addressable by the N_port connected to it. See also *fabric*.

FC See *Fibre Channel*.

FC-AL See *arbitrated loop*.

feature enable identifier

A unique identifier for the storage subsystem, which is used in the process of generating a premium feature key. See also *premium feature key*.

Fibre Channel (FC)

A set of standards for a serial input/output (I/O) bus capable of transferring data between two ports at up to 100 Mbps, with standards proposals to go to higher speeds. FC supports point-to-point, arbitrated loop, and switched topologies.

Fibre Channel Arbitrated Loop (FC-AL)

See *arbitrated loop*.

Fibre Channel Protocol (FCP) for small computer system interface (SCSI)

A high-level fibre-channel mapping layer (FC-4) that uses lower-level fibre-channel (FC-PH) services to transmit SCSI commands, data, and status information between a SCSI initiator and a SCSI target across the FC link by using FC frame and sequence formats.

field replaceable unit (FRU)

An assembly that is replaced in its entirety when any one of its components fails. In some cases, a field replaceable unit might contain other field replaceable units. Contrast with *customer replaceable unit (CRU)*.

FlashCopy

A premium feature for that can make an instantaneous copy of the data in a volume.

F_port See *fabric port*.

FRU See *field replaceable unit*.

GBIC See *gigabit interface converter*

gigabit interface converter (GBIC)

A transceiver that performs serial, optical-to-electrical, and electrical-to-optical signal conversions for high-speed networking. A GBIC can be hot swapped. See also *small form-factor pluggable*.

Global Copy

Refers to a remote logical drive mirror pair that is set up using asynchronous write mode without the write consistency group option. This is also referred to as "Asynchronous Mirroring without Consistency Group." Global Copy does not ensure that write requests to multiple primary logical drives are carried out in the same order on the secondary logical drives as they are on the primary logical drives. If it is critical that writes to the primary logical drives are carried out in the same order in the appropriate secondary logical drives, Global Mirroring should be used instead of Global Copy. See also *asynchronous write mode*, *Global Mirroring*, *remote mirroring*, *Metro Mirroring*.

Global Mirroring

Refers to a remote logical drive mirror pair that is set up using asynchronous write mode with the write consistency group option. This is also referred to as "Asynchronous Mirroring with Consistency Group." Global Mirroring ensures that write requests to multiple primary logical drives are carried out in the same order on the secondary logical drives as they are on the primary logical drives, preventing data on the secondary logical drives from becoming inconsistent with the data on the primary logical drives. See also *asynchronous write mode*, *Global Copy*, *remote mirroring*, *Metro Mirroring*.

graphical user interface (GUI)

A type of computer interface that presents a visual metaphor of a real-world scene, often of a desktop, by combining high-resolution graphics, pointing devices,

menu bars and other menus, overlapping windows, icons, and the object-action relationship.

GUI See *graphical user interface*.

HBA See *host bus adapter*.

hdisk An AIX term representing a logical unit number (LUN) on an array.

heterogeneous host environment

A host system in which multiple host servers, which use different operating systems with their own unique disk storage subsystem settings, connect to the same storage subsystem at the same time. See also *host*.

host A system that is directly attached to the storage subsystem through a fibre-channel input/output (I/O) path. This system is used to serve data (typically in the form of files) from the storage subsystem. A system can be both a storage management station and a host simultaneously.

host bus adapter (HBA)

An interface between the fibre-channel network and a workstation or server.

host computer

See *host*.

host group

An entity in the storage partition topology that defines a logical collection of host computers that require shared access to one or more logical drives.

host port

Ports that physically reside on the host adapters and are automatically discovered by the Storage Manager software. To give a host computer access to a partition, its associated host ports must be defined.

hot swap

To replace a hardware component without turning off the system.

hub In a network, a point at which circuits are either connected or switched. For example, in a star network, the hub is the central node; in a star/ring network, it is the location of wiring concentrators.

IBMSAN driver

The device driver that is used in a Novell NetWare environment to provide multipath input/output (I/O) support to the storage controller.

IC See *integrated circuit*.

IDE See *integrated drive electronics*.

in-band

Transmission of management protocol over the fibre-channel transport.

Industry Standard Architecture (ISA)

Unofficial name for the bus architecture of the IBM PC/XT™ personal computer. This bus design included expansion slots for plugging in various adapter boards. Early versions had an 8-bit data path, later expanded to 16 bits. The "Extended Industry Standard Architecture" (EISA) further expanded the data path to 32 bits. See also *Extended Industry Standard Architecture*.

initial program load (IPL)

The initialization procedure that causes an operating system to commence operation. Also referred to as a system restart, system startup, and boot.

integrated circuit (IC)

A microelectronic semiconductor device that consists of many interconnected transistors and other components. ICs are constructed on a small rectangle cut from a silicon crystal or other semiconductor material. The small size of these circuits allows high speed, low power dissipation, and reduced manufacturing cost compared with board-level integration. Also known as a *chip*.

integrated drive electronics (IDE)

A disk drive interface based on the 16-bit IBM personal computer Industry Standard Architecture (ISA) in which the controller electronics reside on the drive itself, eliminating the need for a separate adapter card. Also known as an Advanced Technology Attachment Interface (ATA).

Internet Protocol (IP)

A protocol that routes data through a network or interconnected networks. IP acts as an intermediary between the higher protocol layers and the physical network.

Internet Protocol (IP) address

The unique 32-bit address that specifies the location of each device or workstation on the Internet. For example, 9.67.97.103 is an IP address.

interrupt request (IRQ)

A type of input found on many processors that causes the processor to suspend normal processing temporarily and start running an interrupt handler routine. Some processors have several interrupt request inputs that allow different priority interrupts.

IP See *Internet Protocol*.

IPL See *initial program load*.

IRQ See *interrupt request*.

ISA See *Industry Standard Architecture*.

Java Runtime Environment (JRE)

A subset of the Java Development Kit (JDK) for end users and developers who want to redistribute the Java Runtime Environment (JRE). The JRE consists of the Java virtual machine, the Java Core Classes, and supporting files.

JRE See *Java Runtime Environment*.

label A discovered or user entered property value that is displayed underneath each device in the Physical and Data Path maps.

LAN See *local area network*.

LBA See *logical block address*.

local area network (LAN)

A computer network located on a user's premises within a limited geographic area.

logical block address (LBA)

The address of a logical block. Logical block addresses are typically used in hosts' I/O commands. The SCSI disk command protocol, for example, uses logical block addresses.

logical partition (LPAR)

A subset of a single system that contains resources (processors, memory, and input/output devices). A logical partition operates as an independent system. If hardware requirements are met, multiple logical partitions can exist within a system.

A fixed-size portion of a logical volume. A logical partition is the same size as the physical partitions in its volume group. Unless the logical volume of which it is a part is mirrored, each logical partition

corresponds to, and its contents are stored on, a single physical partition.

One to three physical partitions (copies). The number of logical partitions within a logical volume is variable.

logical unit number (LUN)

An identifier used on a small computer system interface (SCSI) bus to distinguish among up to eight devices (logical units) with the same SCSI ID.

loop address

The unique ID of a node in fibre-channel loop topology sometimes referred to as a loop ID.

loop group

A collection of storage area network (SAN) devices that are interconnected serially in a single loop circuit.

loop port

A node port (N_port) or fabric port (F_port) that supports arbitrated loop functions associated with an arbitrated loop topology.

LPAR See *logical partition*.

LUN See *logical unit number*.

MAC See *medium access control*.

management information base (MIB)

The information that is on an agent. It is an abstraction of configuration and status information.

man pages

In UNIX-based operating systems, online documentation for operating system commands, subroutines, system calls, file formats, special files, stand-alone utilities, and miscellaneous facilities. Invoked by the **man** command.

MCA See *micro channel architecture*.

media scan

A media scan is a background process that runs on all logical drives in the storage subsystem for which it has been enabled, providing error detection on the drive media. The media scan process scans all logical drive data to verify that it can be accessed, and optionally scans the logical drive redundancy information.

medium access control (MAC)

In local area networks (LANs), the

sublayer of the data link control layer that supports medium-dependent functions and uses the services of the physical layer to provide services to the logical link control sublayer. The MAC sublayer includes the method of determining when a device has access to the transmission medium.

Metro Mirroring

This term is used to refer to a remote logical drive mirror pair which is set up with synchronous write mode. See also *remote mirroring*, *Global Mirroring*.

MIB See *management information base*.

micro channel architecture (MCA)

Hardware that is used for PS/2 Model 50 computers and above to provide better growth potential and performance characteristics when compared with the original personal computer design.

Microsoft Cluster Server (MSCS)

MSCS, a feature of Windows NT Server (Enterprise Edition), supports the connection of two servers into a cluster for higher availability and easier manageability. MSCS can automatically detect and recover from server or application failures. It can also be used to balance server workload and provide for planned maintenance.

mini hub

An interface card or port device that receives short-wave fiber channel GBICs or SFPs. These devices enable redundant fibre channel connections from the host computers, either directly or through a fibre channel switch or managed hub, over optical fiber cables to the DS4000 Storage Server controllers. Each DS4000 controller is responsible for two mini hubs. Each mini hub has two ports. Four host ports (two on each controller) provide a cluster solution without use of a switch. Two host-side mini hubs are shipped as standard. See also *host port*, *gigabit interface converter (GBIC)*, *small form-factor pluggable (SFP)*.

mirroring

A fault-tolerance technique in which information on a hard disk is duplicated on additional hard disks. See also *remote mirroring*.

model The model identification that is assigned to a device by its manufacturer.

MSCS See *Microsoft Cluster Server*.

network management station (NMS)

In the Simple Network Management Protocol (SNMP), a station that runs management application programs that monitor and control network elements.

NMI See *non-maskable interrupt*.

NMS See *network management station*.

non-maskable interrupt (NMI)

A hardware interrupt that another service request cannot overrule (mask). An NMI bypasses and takes priority over interrupt requests generated by software, the keyboard, and other such devices and is issued to the microprocessor only in disastrous circumstances, such as severe memory errors or impending power failures.

node A physical device that allows for the transmission of data within a network.

node port (N_port)

A fibre-channel defined hardware entity that performs data communications over the fibre-channel link. It is identifiable by a unique worldwide name. It can act as an originator or a responder.

nonvolatile storage (NVS)

A storage device whose contents are not lost when power is cut off.

N_port

See *node port*.

NVS See *nonvolatile storage*.

NVSRAM

Nonvolatile storage random access memory. See *nonvolatile storage*.

Object Data Manager (ODM)

An AIX proprietary storage mechanism for ASCII stanza files that are edited as part of configuring a drive into the kernel.

ODM See *Object Data Manager*.

out-of-band

Transmission of management protocols outside of the fibre-channel network, typically over Ethernet.

partitioning

See *storage partition*.

parity check

A test to determine whether the number of ones (or zeros) in an array of binary digits is odd or even.

A mathematical operation on the numerical representation of the information communicated between two pieces. For example, if parity is odd, any character represented by an even number has a bit added to it, making it odd, and an information receiver checks that each unit of information has an odd value.

PCI local bus

See *peripheral component interconnect local bus*.

PDF See *portable document format*.

performance events

Events related to thresholds set on storage area network (SAN) performance.

peripheral component interconnect local bus (PCI local bus)

A local bus for PCs, from Intel[®], that provides a high-speed data path between the CPU and up to 10 peripherals (video, disk, network, and so on). The PCI bus coexists in the PC with the Industry Standard Architecture (ISA) or Extended Industry Standard Architecture (EISA) bus. ISA and EISA boards plug into an IA or EISA slot, while high-speed PCI controllers plug into a PCI slot. See also *Industry Standard Architecture*, *Extended Industry Standard Architecture*.

polling delay

The time in seconds between successive discovery processes during which discovery is inactive.

port

A part of the system unit or remote controller to which cables for external devices (such as display stations, terminals, printers, switches, or external storage units) are attached. The port is an access point for data entry or exit. A device can contain one or more ports.

portable document format (PDF)

A standard specified by Adobe Systems, Incorporated, for the electronic distribution of documents. PDF files are compact; can be distributed globally by

e-mail, the Web, intranets, or CD-ROM; and can be viewed with the Acrobat Reader, which is software from Adobe Systems that can be downloaded at no cost from the Adobe Systems home page.

premium feature key

A file that the storage subsystem controller uses to enable an authorized premium feature. The file contains the feature enable identifier of the storage subsystem for which the premium feature is authorized, and data about the premium feature. See also *feature enable identifier*.

private loop

A freestanding arbitrated loop with no fabric attachment. See also *arbitrated loop*.

program temporary fix (PTF)

A temporary solution or bypass of a problem diagnosed by IBM in a current unaltered release of the program.

PTF See *program temporary fix*.

RAID See *redundant array of independent disks (RAID)*.

RAID level

An array's RAID level is a number that refers to the method used to achieve redundancy and fault tolerance in the array. See also *array, redundant array of independent disks (RAID)*.

RAID set

See *array*.

RAM See *random-access memory*.

random-access memory (RAM)

A temporary storage location in which the central processing unit (CPU) stores and executes its processes. Contrast with *DASD*.

RDAC

See *redundant disk array controller*.

read-only memory (ROM)

Memory in which stored data cannot be changed by the user except under special conditions.

recoverable virtual shared disk (RVSD)

A virtual shared disk on a server node configured to provide continuous access to data and file systems in a cluster.

redundant array of independent disks (RAID)

A collection of disk drives (*array*) that appears as a single volume to the server, which is fault tolerant through an assigned method of data striping, mirroring, or parity checking. Each array is assigned a RAID level, which is a specific number that refers to the method used to achieve redundancy and fault tolerance. See also *array, parity check, mirroring, RAID level, striping*.

redundant disk array controller (RDAC)

(1) In hardware, a redundant set of controllers (either active/passive or active/active). (2) In software, a layer that manages the input/output (I/O) through the active controller during normal operation and transparently reroutes I/Os to the other controller in the redundant set if a controller or I/O path fails.

remote mirroring

Online, real-time replication of data between storage subsystems that are maintained on separate media. The Enhanced Remote Mirror Option is a premium feature that provides support for remote mirroring. See also *Global Mirroring, Metro Mirroring*.

ROM See *read-only memory*.

router A computer that determines the path of network traffic flow. The path selection is made from several paths based on information obtained from specific protocols, algorithms that attempt to identify the shortest or best path, and other criteria such as metrics or protocol-specific destination addresses.

RVSD See *recoverable virtual shared disk*.

SAI See *Storage Array Identifier*.

SA Identifier

See *Storage Array Identifier*.

SAN See *storage area network*.

SATA See *serial ATA*.

scope Defines a group of controllers by their Internet Protocol (IP) addresses. A scope must be created and defined so that dynamic IP addresses can be assigned to controllers on the network.

SCSI See *small computer system interface*.

segmented loop port (SL_port)

A port that allows division of a fibre-channel private loop into multiple segments. Each segment can pass frames around as an independent loop and can connect through the fabric to other segments of the same loop.

sense data

(1) Data sent with a negative response, indicating the reason for the response. (2) Data describing an I/O error. Sense data is presented to a host system in response to a sense request command.

serial ATA

The standard for a high-speed alternative to small computer system interface (SCSI) hard drives. The SATA-1 standard is equivalent in performance to a 10 000 RPM SCSI drive.

serial storage architecture (SSA)

An interface specification from IBM in which devices are arranged in a ring topology. SSA, which is compatible with small computer system interface (SCSI) devices, allows full-duplex packet multiplexed serial data transfers at rates of 20 Mbps in each direction.

server A functional hardware and software unit that delivers shared resources to workstation client units on a computer network.

server/device events

Events that occur on the server or a designated device that meet criteria that the user sets.

SFP See *small form-factor pluggable*.

Simple Network Management Protocol (SNMP)

In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application layer protocol. Information on devices managed is defined and stored in the application's Management Information Base (MIB).

SL_port

See *segmented loop port*.

SMagent

The Storage Manager optional Java-based host-agent software, which can be used on Microsoft Windows, Novell NetWare, AIX, HP-UX, Solaris, and Linux on

POWER host systems to manage storage subsystems through the host fibre-channel connection.

SMclient

The Storage Manager client software, which is a Java-based graphical user interface (GUI) that is used to configure, manage, and troubleshoot storage servers and storage expansion enclosures in a storage subsystem. SMclient can be used on a host system or on a storage management station.

SMruntime

A Java compiler for the SMclient.

SMutil

The Storage Manager utility software that is used on Microsoft Windows, AIX, HP-UX, Solaris, and Linux on POWER host systems to register and map new logical drives to the operating system. In Microsoft Windows, it also contains a utility to flush the cached data of the operating system for a particular drive before creating a FlashCopy.

small computer system interface (SCSI)

A standard hardware interface that enables a variety of peripheral devices to communicate with one another.

small form-factor pluggable (SFP)

An optical transceiver that is used to convert signals between optical fiber cables and switches. An SFP is smaller than a gigabit interface converter (GBIC). See also *gigabit interface converter*.

SNMP

See *Simple Network Management Protocol* and *SNMPv1*.

SNMP trap event

An event notification sent by the SNMP agent that identifies conditions, such as thresholds, that exceed a predetermined value. See also *Simple Network Management Protocol*.

SNMPv1

The original standard for SNMP is now referred to as SNMPv1, as opposed to SNMPv2, a revision of SNMP. See also *Simple Network Management Protocol*.

SRAM

See *static random access memory*.

SSA See *serial storage architecture*.

static random access memory (SRAM)

Random access memory based on the logic circuit known as flip-flop. It is called static because it retains a value as long as power is supplied, unlike dynamic random access memory (DRAM), which must be regularly refreshed. It is however, still volatile, meaning that it can lose its contents when the power is turned off.

storage area network (SAN)

A dedicated storage network tailored to a specific environment, combining servers, storage products, networking products, software, and services. See also *fabric*.

Storage Array Identifier (SAI or SA Identifier)

The Storage Array Identifier is the identification value used by the Storage Manager host software (SMClient) to uniquely identify each managed storage server. The Storage Manager SMClient program maintains Storage Array Identifier records of previously-discovered storage servers in the host resident file, which allows it to retain discovery information in a persistent fashion.

storage expansion enclosure (EXP)

A feature that can be connected to a system unit to provide additional storage and processing capacity.

storage management station

A system that is used to manage the storage subsystem. A storage management station does not need to be attached to the storage subsystem through the fibre-channel input/output (I/O) path.

storage partition

Storage subsystem logical drives that are visible to a host computer or are shared among host computers that are part of a host group.

storage partition topology

In the Storage Manager client, the Topology view of the Mappings window displays the default host group, the defined host group, the host computer, and host-port nodes. The host port, host computer, and host group topological elements must be defined to grant access to host computers and host groups using logical drive-to-LUN mappings.

striping

Splitting data to be written into equal blocks and writing blocks simultaneously to separate disk drives. Striping maximizes performance to the disks. Reading the data back is also scheduled in parallel, with a block being read concurrently from each disk then reassembled at the host.

subnet

An interconnected but independent segment of a network that is identified by its Internet Protocol (IP) address.

sweep method

A method of sending Simple Network Management Protocol (SNMP) requests for information to all the devices on a subnet by sending the request to every device in the network.

switch A fibre-channel device that provides full bandwidth per port and high-speed routing of data by using link-level addressing.

switch group

A switch and the collection of devices connected to it that are not in other groups.

switch zoning

See *zoning*.

synchronous write mode

In remote mirroring, an option that requires the primary controller to wait for the acknowledgment of a write operation from the secondary controller before returning a write I/O request completion to the host. See also *asynchronous write mode*, *remote mirroring*, *Metro Mirroring*.

system name

Device name assigned by the vendor's third-party software.

TCP See *Transmission Control Protocol*.

TCP/IP

See *Transmission Control Protocol/Internet Protocol*.

terminate and stay resident program (TSR program)

A program that installs part of itself as an extension of DOS when it is executed.

topology

The physical or logical arrangement of

devices on a network. The three fibre-channel topologies are fabric, arbitrated loop, and point-to-point. The default topology for the disk array is arbitrated loop.

TL_port

See *translated loop port*.

transceiver

A device that is used to transmit and receive data. Transceiver is an abbreviation of transmitter-receiver.

translated loop port (TL_port)

A port that connects to a private loop and allows connectivity between the private loop devices and off loop devices (devices not connected to that particular TL_port).

Transmission Control Protocol (TCP)

A communication protocol used in the Internet and in any network that follows the Internet Engineering Task Force (IETF) standards for internetwork protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communication networks and in interconnected systems of such networks. It uses the Internet Protocol (IP) as the underlying protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP)

A set of communication protocols that provide peer-to-peer connectivity functions for both local and wide-area networks.

trap In the Simple Network Management Protocol (SNMP), a message sent by a managed node (agent function) to a management station to report an exception condition.

trap recipient

Receiver of a forwarded Simple Network Management Protocol (SNMP) trap. Specifically, a trap receiver is defined by an Internet Protocol (IP) address and port to which traps are sent. Presumably, the actual recipient is a software application running at the IP address and listening to the port.

TSR program

See *terminate and stay resident program*.

uninterruptible power supply

A source of power from a battery that is

installed between a computer system and its power source. The uninterruptible power supply keeps the system running if a commercial power failure occurs, until an orderly shutdown of the system can be performed.

user action events

Actions that the user takes, such as changes in the storage area network (SAN), changed settings, and so on.

worldwide port name (WWPN)

A unique identifier for a switch on local and global networks.

worldwide name (WWN)

A globally unique 64-bit identifier assigned to each fibre channel port.

WORM

See *write-once read-many*.

write-once read many (WORM)

Any type of storage medium to which data can be written only a single time, but can be read from any number of times. After the data is recorded, it cannot be altered.

WWN See *worldwide name*.

zoning

In fibre channel environments, the grouping of multiple ports to form a virtual, private, storage network. Ports that are members of a zone can communicate with each other, but are isolated from ports in other zones.

A function that allows segmentation of nodes by address, name, or physical port and is provided by fabric switches or hubs.

Index

A

address
of the IBM director of licensing 267
address, IBM xiv
AIX host
support xiii

C

comments about this document, how to
send xiv

D

device drivers
downloading latest versions xii
documentation
Web sites xiii
documentation, Storage Subsystem,
Storage Subsystem library 263
drivers xii
DS3000 software/firmware support xi
DS4000 software/firmware support xi
DS5000 software/firmware support xi

F

fire suppression xiv

G

glossary 269

H

hardware service and support xiv
how to send your comments xiv

I

IBM
director of licensing address 267
IBM address xiv
Intel and AMD-based host
support xiii

L

Linux host
support xiii

N

notices
general 267

P

products, developed 267

R

resources
Web sites xiii

S

sending your comments to IBM xiv
services offered in the U.S.A. 267
storage area network (SAN)
technical support Web site xiii
Storage Manager software
where to obtain xii
switch
technical support Web site xiii
System p host
support xiii
System Storage Interoperation Center
(SSIC) xiii
System x host
support xiii

T

trademarks xvii

W

Web sites
AIX xiii
Fix central xiv
IBM publications center xiv
IBM System Storage product
information xiv
interoperability matrix xiii
list xiii
premium feature activation xiii
SAN support xiii
SSIC xiii
switch support xiii
System p xiii
System x xiii

Readers' Comments — We'd Like to Hear from You

IBM System Storage DS3000, DS4000, and DS5000
Command Line Interface and Script Commands Programming Guide

Publication No. GC52-1275-01

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name

Address

Company or Organization

Phone No.

E-mail address



Fold and Tape

Please do not staple

Fold and Tape



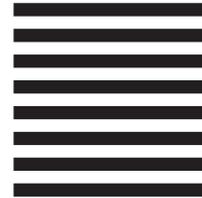
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

DS3000/DS4000/DS5000
GZW
900 South Rita Road
Tucson, AZ
USA 85744-0001



Fold and Tape

Please do not staple

Fold and Tape