RPC12 Series
CLI Reference Manual

P/N 83-00004288-11
Revision A
May 2009

Contents

Preface

This manual describes how to use the command-line interface (CLI) to configure and manage Phoenix™ storage systems, and applies to the following enclosures:

- 69501 FC Controller Enclosure
- 69503 iSCSI Controller Enclosure
- SAS Expansion Enclosure

This book is written for system administrators who are familiar with Fibre Channel (FC), Internet SCSI (iSCSI), and Serial Attached SCSI (SAS) configurations, network administration, and RAID technology.

Before You Read This Book

Before you begin to follow the procedures in this book, you must have already installed the system and learned of any late-breaking information related to system operations as described in the *Getting Started* guide and *Release Notes*.

Typographic Conventions

Typeface[1]	Meaning	Examples
AaBbCc123	Book title, new term, or emphasized word.	See the *Release Notes*. A virtual disk (*vdisk*) can .... You *must* be an advanced user to ....
`AaBbCc123`	Directory or file name, value, command, or on-screen output.	The default file name is `store.logs`. The default IP address is `10.0.0.1`. Type `exit`.
`AaBbCc123`	Text you type, contrasted with on-screen output.	`# set password` `Enter new password:`
`AaBbCc123`	Variable text you replace with an actual value.	Use the format `http://`*`ip-address`*.
`AaBbCc123`	Command-line variable that must be replaced with a real value.	To delete a file, type `rm` *`filename`*.
`[ ]`	Brackets indicate that a parameter is optional.	`show disks [`*`disks`*`]`
`[...]`	Brackets containing an ellipsis indicates that the preceding parameter can be entered again, with a different value.	`set awt` *`trigger`* `enable|disable [...]`
`|`	A separator indicates that only one of the parameters or values separated by this character can be specified.	`base 2|10`

1 The fonts used in your viewer might differ.

Related Documentation

Application	Title	Part Number
Site planning information	*Phoenix Storage System Site Planning Guide*	83-00004283
Late-breaking information not included in the documentation set	*Phoenix 69501 Release Notes* *Phoenix 69503 Release Notes*	83-00004282 83-00005032
Installing and configuring hardware	*Phoenix 69501 Getting Started Guide* *Phoenix 69503 Getting Started Guide*	83-00004284 83-00005034
Configuring and managing storage	*Phoenix RPC12 Administrator's Guide*	83-00004289
Troubleshooting	*Phoenix RPC12 Troubleshooting Guide*	83-00004287
Recommendations for maximizing reliability, accessibility, and serviceability	*Phoenix RPC12 Best Practices Guide*	83-00004286

CHAPTER 1

Using the CLI

This chapter introduces the Phoenix storage system command-line interface (CLI) and includes the following topics:

Accessing the CLI

The CLI software embedded in controller modules enables you to configure, monitor, and manage storage systems out of band.

You can access the CLI in two ways:

- By using a terminal emulator on a management host that is directly connected to a controller module's CLI port. See the *Getting Started Guide* for information on setting up the terminal emulator.
- By using `telnet`, an SSH application, or a terminal emulator on a management host that is remotely connected through a LAN to a controller module's Ethernet port.

Operation Modes

By default the CLI is an interactive application. When you are logged into the CLI, the CLI waits for a command to be entered and then responds to it. This single-operation mode is known as *interactive mode*.

The following example shows interactively starting a `telnet` session, logging into the CLI, executing a command to show free (available) disks, and exiting the CLI:

```
$: telnet 172.22.5.55
Login: monitor
Password: ********

RPC-69501-001
System Name: Test
System Location: Lab
Version: W420B22

# show disks free
ID Serial#               Vendor   Rev. State      Type Size(GB) Rate(Gb/s) SP
------------------------------------------------------------------------------
3  3KN0Z0BZ00007626NM4F  SEAGATE  0003 AVAIL      SAS  146      3.0
6  VDK41BT4CVLPNE        ATA      A9BA AVAIL      SATA 250      3.0
7  3KN0Z6PD00007626NLV5  SEAGATE  0003 AVAIL      SAS  146      1.5
10 3KN0Z09500007626NM28  SEAGATE  0003 AVAIL      SAS  146      3.0
------------------------------------------------------------------------------
# exit
```

CLI commands can also be scripted using a `telnet` client like Expect or a Perl library. Scripts can use interactive mode to execute single commands to emulate a *single-command mode*.

The following example shows the Perl Expect script `showfree.exp` that starts a `telnet` session, logs into the CLI, executes a command to show free disks, and exits the CLI:

```
#!/usr/bin/expect
set login [lindex $argv 1]
set password [lindex $argv 2]
set host [lindex $argv 3]
set command [lindex $argv 4]
spawn telnet $host

expect "Login:"
send "$login\r"
expect "Password:"
send "$password\r"
send "$command\r"
send "exit"
expect eof
```

The following shows a possible result of executing this script:

```
$:./showfree.exp monitor !monitor 172.22.4.245 "show disks free"

Login: monitor
Password: ********

RPC-69501-001
System Name: Test
System Location: Lab
Version: W420B22

# show disks free
ID Serial#               Vendor   Rev. State      Type Size(GB) Rate(Gb/s) SP
--------------------------------------------------------------------------------
3  3KN0Z0BZ00007626NM4F  SEAGATE  0003 AVAIL      SAS  146      3.0
6  VDK41BT4CVLPNE        ATA      A9BA AVAIL      SATA 250      3.0
7  3KN0Z6PD00007626NLV5  SEAGATE  0003 AVAIL      SAS  146      1.5
10 3KN0Z09500007626NM28  SEAGATE  0003 AVAIL      SAS  146      3.0
--------------------------------------------------------------------------------
# exit
```

The example below and continuing on the next page shows how to construct a script using a Perl library for `telnet` communication.

```
use Net::Telnet;
$mVer = "v.072006";
$mLine = "============================================================";
$mStr = "Management Controller System Cloning Utility";
$nLine = "\n";
$cliDumpFile = "get_config_dump.txt";
$space = ' ';
$username = "";
$password = "";

sub cLogin {
    $telnet->open($_[0]);
    $telnet->waitfor(/(login|username)[: ]*$/i);
    $telnet->print("$_[1]");
    $telnet->waitfor(/password[: ]*$/i);
    $telnet->print("$_[2]");

    # either got a login or a prompt
    @ok = $telnet->waitfor(/(#|login:*) /i);

    if ($debug_comamnds == 1) { print "-"; print @ok; print "-\n";  }

    if ($ok[1]  =~ m/login/gi)
    {
        return 0;
    }
    else
    {
        return 1;
    }
}
$ipAddr = $ARGV[0];
$username = $ARGV[1];
$password = $ARGV[2];

$telnet = new Net::Telnet ( Timeout=>10,
                            Errmode=>'die',
                            Prompt => '/\# $/i');
```

```
if ( !cLogin($ipAddr, $username, $password) == 1 )
{
    print("Error: $username user failed to log in. Exiting.\n");
    $telnet->close;
    exit(0);
}
```

The example shows a Perl script for logging in. `cLogin` is called at the start of the script to log a user into the CLI. The script uses the command-line parameters specified as the IP address, username, and password. Once the user has been logged in, other commands can be sent to the CLI.

The command below provides better scripting support. The first argument sets the output format to XML, which allows easier parsing. The second argument disables the paging mode that pauses for each full screen of command output.

```
$telnet->cmd("set cli-parameters api pager disabled");
```

The following code segment shows how to get the entire configuration information from the CLI and print the output. The output can easily be redirected to a file for archiving.

```
@sV = $telnet->cmd("show configuration");

for ($i=0; $i<scalar(@sV); $i++)
{
       print ("@sV[ $i ]");
}
```

Using the XML API

You can set the CLI output to either `console` or `api` using the `set cli-parameters` command. When set to `api`, the output is returned in XML. You can use an XML parser such `XML::Parser` in Perl to process the XML output and store this information as objects.

An updated Document Type Definition (DTD) is available with each update of the firmware. The DTD is located on the Customer Resource Center web site. Go to: `http://crc.dothill.com` > R/Evolution Products > Software Downloads search for the DTD in the article list.

The XML parser should use the DTD version that corresponds to the firmware level to ensure that the DTD is validated. By obtaining the latest DTD for validation, the parser will be forward compatible.

DTD

The following is Revision 1 of the DTD. Elements and attributes are described in the table on the following page.

```
<!ATTLIST OBJECT
    oid ID #REQUIRED
    name CDATA #IMPLIED
    basetype CDATA #IMPLIED
  >

<!--- Property definition -->
<!ELEMENT PROPERTY (#PCDATA)>
<!ATTLIST PROPERTY
    display-name CDATA #IMPLIED
    draw (true|false) #IMPLIED
    size CDATA #IMPLIED
    type (string|uint8|uint16|uint32|uint64|int8|int16|int32|int64|bool|enum)
#IMPLIED
    key (true|false) #IMPLIED
    name CDATA #REQUIRED
  >

<!--- Composition, P is the part component, G is the grouping component -->
<!ELEMENT COMP EMPTY>
<!ATTLIST COMP
    P IDREF #REQUIRED
    G IDREF #REQUIRED
  >

  <!--- Simple Association, A and B are the oids of the Objects -->
<!ELEMENT ASC EMPTY>
<!ATTLIST ASC
    A IDREF #REQUIRED
    B IDREF #REQUIRED
  >
```

Elements and attributes of the DTD are described in the following table.

Element	Description and Attributes
RESPONSE	The RESPONSE is the top level XML object. The request attributes contain the details of the request that was sent. The RESPONSE is a container for objects that are returned as part of the request. All commands have a status object as part of the RESPONSE which provides a message and return code. A return code of 0 indicates that the command was successful. Any other return code is an error code.
OBJECT	The OBJECT element is a container for properties. Each OBJECT also has attributes which describe the OBJECT characteristics. • oid (Object Identifier) - A unique number per request which identifies the OBJECT within a response. This number is used as a reference in associations that the response may contain. • name - The class name of the object. • basetype - The generic class name of the object. This is used to distinguish between differing types of objects within a response. A response can contain a number of volumes all of which may be of different classes (standard volumes, snap-pools, snapshots, etc.). The response also contains a status object. The basetype indicates which object is a volume and which object is a status object.

Element	Description and Attributes
PROPERTY	The PROPERTY element provides the detailed information for each OBJECT. The element value is the actual data content to be used. The attributes provide further details on how to interpret the data. • display-name - The suggested label to use in a user interface for the data value. • raw - Indicates whether or not a value should be displayed to a user. If this value is false, the value might apply only to a programmatic interface which can use the value for calculations. • size - The number of columns to use to display the value. • type - The type that the data should be interpreted as. • key - Indicates whether or not this values can be used as a key to other commands. • name - The name of the property. The name attribute is used for unique identification within an object.
COMP	The composition element gives the association between a grouping object and a part of object. This provides a description of the hierarchy of the objects that are part of the response. • P - The part component oid. • G - The grouping component oid.
ASC	The association element provides a simple association description between two objects in the response. • A • B

Example

The following example shows the output of the `show schedule-details` command with the output first set to `console` and second set to `api`.

```
# set cli-parameters console
Success: CLI parameter changed successfully

# show schedule-details Sched1
Schedule Details
----------------
Schedule Name: Sched1
Schedule Specification: Start 5/01/2007 00:01:00, Only 1st Weekday Of Month
Schedule Status: Ready
Next Time: 5/01/2007 00:01:00
Task To Run: task1
Error Message: none

   Task Details
   ------------
   Task Name: task1
   Task Type: TakeSnapshot
   Task Status: Ready
   Task State: Init
   Master Volume Name: mv1
   Master Volume Serial: 00c0ff0a43010048f9ca2d4601000000
   Snapshot Prefix: ss
   Retention Count: 3
   Last Snapshot Created: none
   Error Message: none

      Snapshot Name        Snapshot Serial
      ------------------------------------------------------
      ss_S0003           00c0ff0a43010048f9ca2d4601000000
```

```
# set cli-parameters api
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<RESPONSE request="cli-parameters">
<OBJECT name="status" oid="1">
    <PROPERTY name="success" type="string" size="77" draw="true" display-name=
"Success">CL
I parameter changed successfully</PROPERTY>
    <PROPERTY name="return-code" type="int32" size="5" draw="false" display-
name="Return C
ode">0</PROPERTY>
</OBJECT>
</RESPONSE>

#show schedule-details Sched1

<?xml version="1.0"?>
<!DOCTYPE RESPONSE SYSTEM "DHXML.dtd">
<RESPONSE>
    <OBJECT basetype="schedule" name="schedule" oid="1">
      <PROPERTY name="name" key="true" type="string" size="32" draw="true"
display-name="Schedule Name">Sched1</PROPERTY>
     <PROPERTY name="schedule-specification" type="string" size="170" draw=
"true" display-name="Schedule Specification">Start 5/01/2007 00:01:00, Only 1st
Weekday Of Month</PROPERTY>
      <PROPERTY name="status" type="string" size="20" draw="true" display-
name="Schedule Status">Ready</PROPERTY>
      <PROPERTY name="next-time" type="string" size="32" draw="true" display-
name="Next Time">5/01/2007 00:01:00</PROPERTY>
      <PROPERTY name="task-to-run" type="string" size="32" draw="true"
display-name="Task To Run">task1</PROPERTY>
      <PROPERTY name="error-message" type="string" size="100" draw="true"
display-name="Error Message">none</PROPERTY>
</OBJECT>
    <COMP G="1" P="2" />
    <OBJECT basetype="task" name="task" oid="2">
      <PROPERTY name="name" key="true" type="string" size="32" draw="true"
display-name="Task Name">task1</PROPERTY>
      <PROPERTY name="type" type="string" size="32" draw="true" display-name=
"Task Type">TakeSnapshot</PROPERTY>
      <PROPERTY name="status" type="string" size="10" draw="true" display-
name="Task Status">Ready</PROPERTY>
     <PROPERTY name="state" type="string" size="15" draw="true" display-name=
"Task State">Init</PROPERTY>
      <PROPERTY name="master-volume-name" type="string" size="32" draw="true"
display-name="Master Volume Name">mv1</PROPERTY>
```

```
     <PROPERTY name="master-volume-serial" type="string" size="32" draw="true"
display-name="Master Volume
Serial">00c0ff0a43010048f9ca2d4601000000</PROPERTY>
    <PROPERTY name="snapshot-prefix" type="string" size="14" draw="true"
display-name="Snapshot Prefix">ss</PROPERTY>
    <PROPERTY name="retention-count" type="string" size="5" draw="true"
display-name="Retention Count">3</PROPERTY>
    <PROPERTY name="last-created" type="string" size="32" draw="true" display-
name="Last Snapshot Created">none</PROPERTY>
    <PROPERTY name="error-message" type="string" size="100" draw="true"
display-name="Error Message">none</PROPERTY>
    </OBJECT>
</RESPONSE>
```

Command Syntax

This section describes syntax rules for CLI commands.

Keywords and Parameters

Command keywords must be entered in lowercase. Parameter values can be entered in uppercase and lowercase.

Parameter values that contain non-alphanumeric characters, such as spaces, must be enclosed in quotation marks ("), which the CLI parses and removes.

Disk Drive Syntax

You can specify disk drives by using either:

- A disk drive ID. For example, `4`.
- A hyphenated range of disk drive IDs from *a* to *z*. For example, `5-7`.
- A list of disk drive IDs, ranges, or both, separated by commas; do not include spaces before or after commas. For example, `4,6-9`.

Virtual Disk Syntax

You can specify virtual disks by using either:

- **Virtual disk serial number.** A unique 32-digit number that is automatically assigned when a virtual disk is created, and does not change for the life of the virtual disk.
- **Virtual disk name.** A user-defined, case-sensitive name that can include 20 characters. Allowed characters include letters, numbers, hyphens, underscores, and spaces. A name that includes a space must be enclosed in quotation marks.

Some commands accept a comma-separated list of virtual disk serial numbers and names. Do not include spaces before or after commas. The following virtual disk list specifies a serial number and two names:

```
00c0ff0a43180048e6dd1c4500000000,Sales/Mktg,”Vdisk #1”
```

Volume Syntax

You can specify volumes by using either:

- **Volume serial number.** A unique 32-digit number that is automatically assigned when a volume is created, and does not change for the life of the volume.
- **Volume name.** A user-defined name that can include a maximum of 17 printable ASCII characters. A name cannot include a comma, backslash (\), or quotation mark ("); however, a name that includes a space must be enclosed in quotation marks. Names are case sensitive.

Note – Volumes on different virtual disks can have the same name.

Some commands accept a comma-separated list of volume serial numbers and names. Do not include spaces before or after commas. The following volume list specifies a serial number and two names:

```
AA43BF501234560987654321FEDCBA,Image-Data,”Vol #1”
```

Host Nickname Syntax

You can specify a nickname for a data host's bus adapter or switch port. A nickname is a user-defined string that can include a maximum of 16 printable ASCII characters. For example, `MyHBA`. A name cannot include a comma, backslash (\), or quotation mark ("); however, a name that includes a space must be enclosed in quotation marks. Names are case sensitive.

Volume Mapping Syntax

You specify the mapping of a host to a volume by using the syntax *`channels`*.*`LUN`*, where:

- *`channels`* is a single host channel number or a list of host channel numbers, ranges, or both. For example, `0,1,3-5`.
- *`LUN`* is a logical unit number (LUN) from 0–127 to assign to the mapping. For example, `8`.

The following example maps channels 0 and 1 using LUN 8:

```
0-1.8
```

Command Completion, Editing, and History

The CLI supports command completion, command editing, and command history.

When using command completion, if you enter too few letters to uniquely identify a keyword, the CLI lists keywords that match the entered string and redisplays the string so you can complete it.

The history contains all commands entered in the active CLI session. You can recall a command from the history, edit it, and run it.

To	Press
Complete a partially entered keyword	Tab
Get previous command from history	Up Arrow or Ctrl-P
Get next command from history	Down Arrow or Ctrl-N
Move cursor left	Left Arrow or Ctrl-B
Move cursor right	Right Arrow or Ctrl-F
Move back one word	Esc-B
Move forward one word	Esc-F
Move cursor to start of line	Ctrl-A
Move cursor to end of line	Ctrl-E
Transpose current and previous character	Ctrl-T
Delete current character	Delete or Ctrl-D
Delete previous character	Backspace
Delete word up to cursor	Ctrl-W
Delete rest of word	Esc-D
Delete text up to cursor	Ctrl-U
Delete rest of line	Ctrl-K
Convert rest of word to uppercase	Esc-C
Convert rest of word to lowercase	Esc-L
Enter command and redisplay prompt	Ctrl-Z
Refresh input line	Ctrl-L

Viewing Command Help

To view brief descriptions of all commands that are available to the user level you logged in as, type:

```
# help
```

To view help for a specific command, type either:

```
# help command
# command ?
```

To view information about the syntax to use for specifying disk drives, virtual disks, volumes, and volume mapping, type:

```
# help syntax
```

Size of Devices and Logical Units

The size of disk drives and logical units such as virtual disks and volumes are presented in base 2 (binary) format, not base 10 (decimal) format. Table 1-1 shows how sizes are presented in each format.

Table 1-1 Sizes in Binary and Decimal

Unit	Size in Binary	Size in Decimal
Kbyte	1024 bytes	1000 bytes
Mbyte	1024 Kbyte 1,048,576 bytes	1000 Kbyte 1 million bytes
Gbyte	1024 Mbyte 1,073,741,824 bytes	1000 Mbyte 1 billion bytes
Tbyte	1024 Gbyte 1,099,511,627,776 bytes	1000 Gbyte 1 trillion bytes

To check which format is being used, use “show cli-parameters” on page 146. To change the output format, use “set cli-parameters” on page 106.

Event Log

A controller enclosure's event log records all events that have occurred in or been detected by the controller modules and encompasses all field-replaceable units (FRUs) in the enclosure.

Each event has one of the following levels, in decreasing severity:

- **Critical.** Events that might affect data integrity or system stability.
- **Warning.** Events that do not affect data integrity.
- **Informational.** Events that show the change of state or configuration changes.

For information about viewing events, see "show events" on page 162.

CHAPTER 2

Command Categories

Commands in this manual are organized alphabetically to make it easy to find a command by name. This chapter helps you find a command within a logical grouping, based on the command's function. A command might be listed in more than one category.

CLI and Users

- create user
- delete user
- exit
- help; see “Viewing Command Help” on page 28
- set cli-parameters
- set password
- set prompt
- set user
- show cli-parameters
- show users

Virtual Disks and Disk Drives

- abort create
- abort scrub
- abort verify
- clear disk-metadata
- create vdisk
- delete global-spare
- delete vdisk
- delete vdisk-spare
- expand vdisk
- scrub vdisk
- set expander-fault-isolation
- set led
- set vdisk
- set vdisk-spare
- show disks
- show vdisks
- trust
- verify vdisk

Volumes and Mapping

- abort volumecopy
- create host-wwn-name
- create iscsi-host
- create volume
- delete host-wwn-name
- delete iscsi-host
- delete volume
- expand volume
- map volume
- set cache-parameters
- set host-wwn-name
- set iscsi-host
- set volume
- show cache-parameters
- show host-maps
- show host-wwn-names
- show iscsi-hosts
- show port-wwn
- show volumes
- show volumecopy-status
- show volume-maps
- unmap volume
- volumecopy

Snapshots

- convert master-to-std
- convert std-to-master
- create master-volume
- create snap-pool
- create snapshots
- delete all-master-volumes
- delete all-snapshots
- delete master-volume
- delete snap-pool
- delete snapshot
- delete snapshot-write-data
- expand snap-pool
- reset snapshot
- rollback master-volume
- set snap-pool-policy

- set snap-pool-threshold
- show master-volumes
- show snap-pools
- show snapshots

Task Scheduling

- create schedule
- create task
- delete schedule
- delete task
- show schedule
- show schedule-details
- show tasks
- show task-detail

Event Notification

- set snmp-parameters
- show events
- show snmp-parameters

Configuration and Utilities

- alias
- clear cache
- ping
- reset host-channel-link
- restart
- set auto-write-through-trigger
- set controller-date
- set drive-parameters
- set expander-fault-isolation
- set expander-phys
- set host-parameters
- set host-port-interconnects
- set job-parameters
- set led
- set network-parameters
- set password
- set protocols

- set system
- show auto-write-through-trigger
- show channels
- show configuration
- show controller-date
- show drive-parameters
- show enclosure-status
- show expander-status
- show frus
- show host-parameters
- show host-port-interconnects
- show job-parameters
- show license
- show network-parameters
- show protocols
- show redundancy-mode
- show shutdown-status
- show sensor-status
- show system
- shutdown
- stty
- versions

Service Utilities

- clear expander-status
- restore defaults
- set debug-log-parameters
- show debug-log
- show debug-log-parameters

CHAPTER 3

Command Descriptions

This chapter describes the commands in alphabetical order. Each command topic includes one or more of the following sections:

- **Description.** The command's purpose and notes about its usage.
- **Input.** The command's syntax and descriptions of its parameters.
- **Output.** A description of information that is displayed by the command.
- **Examples.** One or more examples of the command's usage, if the command has parameters or detailed output.
- **Error Messages.** Descriptions of error messages you might encounter while using the command.
- **Related Commands.** Cross-references to commands that are used with or similar to the command.

abort create

Description

Stops the `create vdisk` operation for a virtual disk being initialized online or offline. This command does not revert the system to the state it was in before starting to create the virtual disk; instead, the virtual disk is left in an offline state (status `OFFL`).

Input

```
abort create vdisk vdisk
```

Parameter	Description
vdisk	Specifies the virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Abort creating virtual disk VD1.

```
# abort create vdisk VD1
```

Related Commands

- "create vdisk" on page 61
- "set vdisk" on page 137

abort scrub

Description

Aborts the `scrub vdisk` operation for specified virtual disks.

Input

```
abort scrub vdisk vdisks
```

Parameter	Description
`vdisk` *`vdisks`*	Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Abort scrubbing virtual disk VD1.

```
# abort scrub vdisk VD1
```

Related Commands

- "scrub vdisk" on page 100

abort verify

Description

Aborts the `verify vdisk` operation for specified virtual disks.

Input

```
abort verify vdisk vdisks
```

Parameter	Description
`vdisk` *vdisks*	Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Abort verifying virtual disk VD1.

```
# abort verify vdisk VD1
```

Related Commands

- "verify vdisk" on page 224

abort volumecopy

Description

Aborts copying a volume. When the abort is complete, the destination volume is deleted

Input

```
abort volumecopy volume
```

Parameter	Description
volume	Specifies the name or serial number of either the source or the destination volume. For the syntax to use, see “Virtual Disk Syntax” on page 25.

Example

Abort creating destination volume Copy.

```
# abort volumecopy Copy
```

Related Commands

- “show volumecopy-status” on page 211
- “show volumes” on page 215
- “volumecopy” on page 226

alias

Description

Shows, creates, or deletes aliases for the current CLI session. An alias is usually a short string that is substituted for a longer string. If both parameters are omitted, existing aliases are shown.

Input

```
alias [alias] [command-string]
```

Parameter	Description
alias	Optional. Specifies the string to substitute for the command string. An alias that includes a space must be enclosed in quotation marks ("). Aliases are not case sensitive; for example, new alias `SS` will replace existing alias `ss`. If the alias exists and the command string parameter is omitted, the alias is deleted.
command-string	Optional. Specifies the command string that the alias substitutes for. A string that includes a space must be enclosed in quotation marks (").

Output

Shows, in the order aliases were created, each alias and its associated command string.

Example

Create alias `se5` for a command that shows the last five events.

```
# alias se5 "show events last 5"
```

Show existing aliases.

```
# alias

se5   show events last 5
sd    show disks
```

clear cache

Description

Clears any unwritable cache in the RAID controllers for a specified volume, or any orphaned data for volumes that no longer online or no longer exist. Orphaned data can occur if there is a power loss during I/O, a drive is removed from a vdisk while the power is off, or two or more drives in the same vdisk go down causing the vdisk to go down. Unwritable data can only exist if write-back cache is enabled. If writeback cache is disabled, then unwritable cache will never appear.

Input

```
clear cache [volume volume]
```

Parameter	Description
volume *volume*	Optional. Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25. If this parameter is omitted, the command clears any unneeded orphaned data for volumes that are no longer online or no longer exist.

Example

Clear the cache in both controllers for volume V1.

```
# clear cache volume Volume1
Success: Orphan cache data has been cleared
```

clear disk-metadata

Description

Clears metadata from “leftover” disk drives. Each disk drive contains metadata that the system uses to identify the drive’s owning virtual disk, if any. If the system cannot locate the virtual disk, as when the drive has been moved to a different system, the owning virtual disk is shown as Leftover. You must clear the metadata before you can use the drive in a different virtual disk or as a spare.

Input

```
clear disk-metadata disks
```

Parameter	Description
disks	Specifies the disk drives by a range or list of IDs. For the syntax to use, see “Disk Drive Syntax” on page 24.

Example

Clear metadata for disk drives having ID 0 and 4.

```
# clear disk-metadata 0,4
```

clear expander-status

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Clears the counters and status for SAS expander controller lanes. Counters and status can be reset to a good state for all enclosures, or for a specific enclosure whose status is ERROR as shown by the `show expander-status` command.

Input

```
clear expander-status [enclosure id]
```

Parameter	Description
`enclosure` *id*	Optional. Specifies the enclosure number, as shown by the `show expander-status` command.

Example

Clear the expander status for the controller enclosure.

```
# clear expander-status enclosure 0
```

Related Commands

- "show expander-status" on page 165
- "set expander-fault-isolation" on page 113
- "set expander-phy" on page 114

convert master-to-std

Description

Converts a specified master volume into a standard volume; that is, it disables the volume from accepting snapshots.

Note – You must delete all snapshots that exist for the master volume before converting it to a standard volume.

Input

```
convert master-to-std volume
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25.

Example

Convert master volume `MV1` to a standard volume.

```
# convert master-to-std MV1
```

Related Commands

- “delete all-snapshots” on page 67
- “show master-volumes” on page 180

convert std-to-master

Description

Converts a standard volume to a master volume; that is, it enables the volume for snapshots and associates it with an existing snap pool. The standard volume and the snap pool must be owned by the same controller, though they can be on different virtual disks.

Input

```
convert std-to-master volume snap-pool volume
```

Parameter	Description
volume	Specifies the master volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
snap-pool *volume*	Specifies the snap-pool volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Example

Convert standard volume V1 to a master volume and associate it with snap pool SP1.

```
# convert std-to-master V1 snap-pool SP1
```

Related Commands

- "show master-volumes" on page 180
- "show snap-pools" on page 194

create host-wwn-name

Description

Creates an HBA entry for a host world wide port name (WWPN) and associates a nickname with the entry. This enables the nickname to be used instead of the numeric WWPN when mapping volumes to hosts.

Input

```
create host-wwn-name wwn wwpn name
```

Parameter	Description
wwn *wwpn*	Specifies a 16-hex-digit WWPN that corresponds to an HBA.
name	Specifies a nickname for the HBA. For the syntax to use, see "Host Nickname Syntax" on page 26.

Example

Create the nickname HBA1 for the HBA having WWPN 210000e08b095562.

```
# create host-wwn-name wwn 210000e08b095562 HBA1
```

Related Commands

- "delete host-wwn-name" on page 69
- "set host-wwn-name" on page 120
- "show host-wwn-names" on page 175

create iscsi-host

Description

Associates a nickname with an iSCSI host initiator's IP address. You can then use the nickname when mapping volumes to hosts.

Input

```
create iscsi-host ip IP nickname
```

Parameter	Description
ip *IP*	Specifies an iSCSI host initiator's IP address in IPv4 format.
nickname	Specifies a nickname for the iSCSI host initiator. For the syntax to use, see "Host Nickname Syntax" on page 26.

Example

Create the nickname iSCSI-1 for the iSCSI host initiator at IP address 192.168.0.254.

```
# create iscsi-host ip 192.168.0.254 iSCSI-1
```

Related Commands

- "delete iscsi-host" on page 70
- "set iscsi-host" on page 121
- "show iscsi-hosts" on page 176

create master-volume

Description

Creates a volume on a specified virtual disk, enables the volume for snapshots, and associates it with an existing snap pool. The volume and snap pool must be owned by the same controller.

Input

```
create master-volume vdisk vdisk size sizeMB|GB|TB
snap-pool volume name [lun LUN]
```

Parameter	Description
vdisk *vdisk*	Specifies the virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
size *size*MB|GB|TB	Specifies the volume size in MB (Mbyte), GB (Gbyte), or TB (Tbyte). The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, type show cli-parameters.
snap-pool *volume*	Specifies the snap pool volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
name	Specifies a name for the new master volume, using a maximum of 20 characters and no apostrophe ('), quotation mark ("), or backslash (\) characters.
lun *LUN*	Optional. Specifies a logical unit number to assign to the new master volume. If this parameter is omitted, no LUN is assigned.

Example

Create the 1-Gbyte master volume MV1 on virtual disk VD1, and associates it with snap pool SP1.

```
# create master-volume vdisk VD1 size 1GB snap-pool SP1 MV1
```

Related Commands

- “delete master-volume” on page 71
- “show master-volumes” on page 180
- “show snap-pools” on page 194
- “show vdisks” on page 208

create schedule

Description

Schedules a task to run automatically.

Input

```
create schedule sname schedule-specification "specification"
task-name tname
```

Parameter	Description
sname	Specifies a name for the new schedule, using a maximum of 32 characters and no apostrophe ('), quotation mark ("), or backslash (\) characters. Names are case sensitive.
schedule-specification "*specification*"	Specifies when the task will run the first time, optional conditions defining when the task will recur and expire. You can use a comma between conditions. • start *mm*/*dd*/*yyyy hh*:*mm* [AM|PM] If AM|PM is not specified, a 24 hour clock is used. If you also use the between condition, below, the start time must be in the between range. Optional specifications: • every # minutes|hours|days|months|years • between *hh*:*mm* [AM|PM] and *hh*:*mm* [AM|PM] • only any|first|second|third|fourth|fifth last|#st|#nd|#rd|#th| weekday|weekend day |Sunday|Monday|Tuesday| Wednesday|Thursday|Friday|Saturday of year|month|January|February|March|April| May|June|July|August|September|October| November|December • count # • expires *mm*/*dd*/*yyyy hh*:*mm* [AM|PM]
task-name *tname*	Specifies the name of the task to run. The name is case sensitive.

Example

Create schedule Sched1 that runs Task1 for the first time on March 1, 2007, runs daily between midnight and 1 AM, and runs for the last time in the morning of January 1, 2008.

```
# create schedule Sched1 schedule-specification "start 03/01/2007
00:01 AM, every 1 days, between 12:00 AM and 1:00 AM, expires
1/1/2008 1:00 AM" task-name Task1

Success: Schedule created
```

Create schedule Sched2 that runs Task2 for the first time on March 1, 2007, and runs only on the first weekday of each month, with no expiration.

```
# create schedule Sched2 schedule-specification "start 03/01/2007
00:01 only first weekday of month" task-name Task2

Success: Schedule created
```

Related Commands

- “create task” on page 56
- “delete schedule” on page 72
- “show schedule-details” on page 188
- “show schedules” on page 190
- “show task-details” on page 201
- “show tasks” on page 205

create snap-pool

Description

Creates a snap pool volume to use for snapshot data. A snap pool is an internal volume only and cannot be assigned a LUN.

Input

```
create snap-pool vdisk vdisk size sizeMB|GB|TB name
```

Parameter	Description
vdisk *vdisk*	Specifies the virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
size *size*MB|GB|TB	Specifies the volume size in MB (Mbyte), GB (Gbyte), or TB (Tbyte). The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, type show cli-parameters. A snap pool should be no smaller than 10 GB (10,000 MB).
name	Specifies a name for the new snap pool volume, using a maximum of 20 characters and no apostrophe ('), quotation mark ("), or backslash (\) characters.

Example

Create the 10-Gbyte snap pool SP1 on virtual disk VD1.

```
# create snap-pool vdisk VD1 size 10GB SP1
```

Related Commands

- "delete snap-pool" on page 73
- "show snap-pools" on page 194
- "show vdisks" on page 208

create snapshots

Description

Creates a named snapshot of each specified master volume.

Input

```
create snapshots master-volumes volumes snap-names
```

Parameter	Description
`master-volumes` *`volumes`*	Specifies a comma-separated list of volumes by their names or serial numbers. For the syntax to use, see "Volume Syntax" on page 25.
`snap-names`	Specifies a comma-separated list of names for the resulting snapshots. For the syntax to use, see "Volume Syntax" on page 25.

Example

Create snapshot SS1 of master volume MV1, and snapshot SS2 of master volume `MV2`.

```
# create snapshots master-volumes MV1,MV2 SS1,SS2
```

Related Commands

- "delete snapshot" on page 74
- "show master-volumes" on page 180
- "show snapshots" on page 196

create task

Description

Creates a task that can be scheduled. You can create a task to take a snapshot of a master volume, to copy a snapshot or a master volume to a new standard volume, or to reset a snapshot.

Caution – Before scheduling a reset snapshot task, consider that if the snapshot is mounted to a host operating system, the snapshot must be unmounted before the reset is performed; leaving it mounted can cause data corruption. You should create a scheduled job on the host to unmount the snapshot prior to resetting the snapshot.

Input

```
create task name type TakeSnapshot master-volume volume snapshot-
prefix prefix retention-count #
```

```
create task name type ResetSnapshot snapshot-volume volume
```

```
create task name type VolumeCopy source-volume volume dest-vdisk
vdisk dest-prefix prefix [modified-snapshot yes|no]
```

Parameter	Description
name	Specifies a name for the new task, using a maximum of 32 characters and no apostrophe ('), quotation mark ("), or backslash (\) characters. Names are case sensitive.

Parameter	Description
`type TakeSnapshot|ResetSnapshot|VolumeCopy`	Specifies the task. • `TakeSnapshot`: Takes a snapshot of a specified master volume. Up to 16 snapshots per controller pair can exist, or more if your license permits it. • `ResetSnapshot`: Deletes the data in the snapshot and resets it to the current data in the associated master volume. The snapshot's name and other volume characteristics are not changed. • `VolumeCopy`: Copies a snapshot or a master volume to a new standard volume. The command creates the destination volume you specify, which must be in a vdisk owned by the same controller as the source volume.
`master-volume` *volume*	Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`snapshot-prefix` *prefix*	Label to identify snapshots created by this task. Snapshot names have the format *pref*`_S#`, where *#* increments from 0001 to 9999 before rolling over.
`retention-count` *#*	Number of snapshots with this prefix to retain. When a new snapshot exceeds this limit, the oldest snapshot with the same prefix is deleted.
`snapshot-volume` *volume*	Specifies the snapshot volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`source-volume` *volume*	Specifies the master volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`dest-vdisk` *vdisk*	Specifies the destination virtual disk by name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`dest-prefix` *prefix*	Label to identify the volume copy created by this task. Copy names have the format *prefix*`_V#`, where *#* increments from 0001 to 9999 before rolling over.

Parameter	Description
`modified-snapshot yes|no`	Optional. Specifies whether to include or exclude modified write data from the snapshot in the copy. This parameter applies only when the source volume is a snapshot; it is ignored if the source volume is a master volume. • `yes`: Include modified snapshot data. • `no`: Exclude modified snapshot data. If this parameter is omitted for a snapshot, modified snapshot data is excluded.

Example

Create task Task1 that takes a snapshot of master volume VD1_V1 and retains only the latest four snapshots with the prefix VD1_V1, e.g., VD1_V1_S0001.

```
# create task Task1 type TakeSnapshot master-volume VD1_V1
snapshot-prefix VD1_V1 retention-count 4

Success: Task created
```

Create task Task2 that resets snapshot VD1_S0001.

```
# create task Task2 type ResetSnapshot snapshot-volume VD1_S0001

Success: Task created
```

Create task Task3 that copies volume VD1_V1 to virtual disk VD2 with the name C_V0001.

```
# create task Task3 type VolumeCopy source-volume VD1_V1 dest-
vdisk VD2 dest-prefix C modified-snapshot yes

Success: Task created
```

Related Commands

- “create schedule” on page 52
- “delete task” on page 76
- “show schedule-details” on page 188
- “show task-details” on page 201
- “show tasks” on page 205

create user

Description

Creates a new user profile. When you enter the command, the system prompts you to enter a case-sensitive password, which can include a maximum of 19 printable ASCII characters except backslash (\), quotation mark ("), apostrophe ('), or spaces. The system supports 13 user profiles.

Input

```
create user name [level monitor|manage] [type
standard|advanced|diagnostic] [interfaces interfaces]
```

Parameter	Description
name	Unique name that can include a maximum of 19 alphanumeric characters. Names are case sensitive.
`level monitor|manage`	Optional. Specifies whether the user has `monitor` (view-only) or `manage` (modify) access for user interfaces. The default is `monitor`.
`type standard|advanced |diagnostic`	Optional. Specifies the user's level of technical expertise, to control access to functionality in the WBI. • `standard`: Enables access to standard administrative functions. This is the default for monitor users. • `advanced`: Enables access to standard and advanced functions. This is the default for manage users. • `diagnostic`: Enables access to standard, advanced, and troubleshooting functions. This is the default for users of the CLI.
`interfaces` *interfaces*	Optional. Specifies the interfaces that the user can access. Multiple values must be separated by a comma with no spaces. If this parameter is omitted, the defaults are `cli` and `wbi`. • `cli`: Command-line interface. • `wbi`: Web-browser interface. • `ftp`: File transfer protocol interface. • `none`: No interfaces.

Example

Create an advanced manage user who can use the web and FTP interfaces.

```
# create user jsmith level manage interfaces wbi,ftp

Enter Password for new user jsmith:*****
Re-enter Password:*****
Info: User Type not specified, defaulting to Advanced.
Success: New user created
```

Related Commands

- “delete user” on page 77
- “set user” on page 135
- “show users” on page 206

create vdisk

Description

Creates a virtual disk using the specified RAID level, disk drives, and optional vdisk spares. All disks used in a virtual disk and its spares must be either SAS or SATA; mixing disk types is not supported.

Table 3-1 specifies the number of disks supported for each virtual disk type, as determined by its RAID level.

Table 3-1 Number of Disk Drives Required for Each RAID Level

RAID Level	Minimum–Maximum Number of Disk Drives
NRAID (non-RAID)	1
0	2–16
1	2
3	3–16
5	3–16
6	4-16
10	4–16
50	6–32

Input

```
create vdisk level nraid|raid0|r0|raid1|r1|raid3|r3|raid5|r5
|raid6|r6|raid10|r10|raid50|r50 disks disks
[assigned-to a|b|auto][spare disks] [chunk-size 16k|32k|64k] [mode
online|offline] name
```

Parameter	Description
`level nraid|raid0|r0|raid1|r1|raid3|r3|raid5|r5|raid6|r6|raid10|r10|raid50|r50`	Specifies non-RAID (`nraid`) or a RAID level.

Parameter	Description
`disks` *disks*	Specifies the disk drives by a range or list of IDs; or for RAID 10, a colon-separated list of RAID-1 subvirtual disks; or for RAID 50, a colon-separated list of RAID-5 subvirtual disks. For the syntax to use, see "Disk Drive Syntax" on page 24.
`assigned-to a|b|auto`	Optional. Specifies the controller to own the virtual disk. To have the system automatically load-balance virtual disks between controllers, use `auto` or omit this parameter.
`spare` *disks*	Optional. Specifies up to four vdisk spares to assign to a RAID 1, 3, 5, 6, 10, or 50 virtual disk. For the syntax to use, see "Disk Drive Syntax" on page 24.
`chunk-size 16k|32k|64k`	Optional. Specifies the chunk size in Kbyte. The default is `64k`.
`mode online|offline`	Optional. Specifies whether the virtual disk is initialized online or offline. The default is `online`.
name	Specifies a name for the new virtual disk. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Create the RAID-1 virtual disk VD1 using two disk drives.

```
# create vdisk level raid1 disks 1,3 VD1
```

Create the RAID-50 virtual disk VD2 having three RAID-5 subvirtual disks, each having three disk drives.

```
# create vdisk level r50 disks 1-3:4-6:7-9 VD2
```

Related Commands

- “abort create” on page 38
- “delete vdisk” on page 78
- “set vdisk” on page 137
- “show disks” on page 154
- “show vdisks” on page 208

create volume

Description

Creates a volume on a virtual disk. This command enables you to specify a size and name for the volume, and map it to hosts.

Input

```
create volume vdisk vdisk size sizeMB|GB|TB [mapping mapping] name
```

Parameter	Description
vdisk *vdisk*	Specifies the virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
size *size*MB|GB|TB	Specifies the volume size in MB (Mbyte), GB (Gbyte), or TB (Tbyte). The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, type show cli-parameters.
mapping *mapping*	Optional. Specifies the host-to-volume mapping. For the syntax to use, see "Volume Mapping Syntax" on page 26. If this argument is omitted, the volume is unmapped and its LUN is set to None.
name	Specifies a name for the volume. For the syntax to use, see "Volume Syntax" on page 25.

Example

Create the 10-Gbyte volume V1 on virtual disk VD1, and map it to LUN 12 on channel 1.

```
# create volume vdisk VD1 size 10GB mapping 1.12 V1
```

Related Commands

- “delete volume” on page 81
- “map volume” on page 89
- “set volume” on page 139
- “show vdisks” on page 208
- “show volumes” on page 215
- “unmap volume” on page 222

delete all-master-volumes

Description

Deletes all master volumes associated with a snap pool.

Input

```
delete all-master-volumes snap-pool volume
```

Parameter	Description
`snap-pool` *`volume`*	Specifies the snap pool volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Example

Delete all master volumes associated with snap pool SP1.

```
# delete all-master-volumes snap-pool SP1
```

Related Commands

- "show master-volumes" on page 180
- "show snap-pools" on page 194

delete all-snapshots

Description

Deletes all snapshots associated with a specified volume, which can be a snap pool or master volume. All data associated with the snapshots is deleted and associated space in the snap pool is freed for use.

Input

```
delete all-snapshots volume volume
```

Parameter	Description
`volume` *`volume`*	Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Example

Delete all snapshots associated with master volume MV1.

```
# delete all-snapshots volume MV1
```

Related Commands

- "show snapshots" on page 196
- "show volumes" on page 215

delete global-spare

Description

Removes one or more disk drives from the pool of global spares.

Input

```
delete global-spare disks disks
```

Parameter	Description
`disks` *disks*	Specifies the disk drives to remove. For the syntax to use, see "Disk Drive Syntax" on page 24.

Example

Remove the global spare having ID 5.

```
# delete global-spare disks 5
```

Related Commands

- "show disks" on page 154

delete host-wwn-name

Description

Deletes the nickname associated with a host world wide port name (WWPN).

Input

```
delete host-wwn-name host
```

Parameter	Description
host	Specifies the HBA's nickname or 16-hex-digit WWPN. For the nickname syntax to use, see “Host Nickname Syntax” on page 26.

Example

Delete the nickname HBA1.

```
# delete host-wwn-name HBA1
```

Related Commands

- “create host-wwn-name” on page 48
- “set host-wwn-name” on page 120
- “show host-wwn-names” on page 175

delete iscsi-host

Description

Deletes the nickname associated with an iSCSI host initiator's IP address.

Input

```
delete iscsi-host descriptor
```

Parameter	Description
descriptor	Specifies the iSCSI host initiator's IP address in IPv4 format, or nickname. For the nickname syntax to use, see “Host Nickname Syntax” on page 26.

Example

Delete the iSCSI host initiator nickname iSCSI-1.

```
# delete iscsi-host iSCSI-1
```

Related Commands

- “create iscsi-host” on page 49
- “set host-wwn-name” on page 120
- “show iscsi-hosts” on page 176

delete master-volume

Description

Deletes a master volume. Alias of `delete volume`.

Note – You must delete all snapshots that exist for the master volume before you can delete it.

Input

```
delete master-volume volume
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25.

Example

Delete master volume MV1.

```
# delete master-volume MV1
```

Related Commands

- “delete all-snapshots” on page 67
- “show master-volumes” on page 180

delete schedule

Description

Deletes a specified schedule.

Input

```
delete schedule schedule
```

Parameter	Description
schedule	Specifies the schedule name.

Example

Delete schedule Sched1.

```
# delete schedule Sched1

Success: Schedule Deleted
```

Related Commands

- “create schedule” on page 52
- “show schedule-details” on page 188
- “show schedules” on page 190
- “show task-details” on page 201
- “show tasks” on page 205

delete snap-pool

Description

Deletes a snap pool.

Note – You must disassociate all master volumes from the snap pool before you can delete it.

Input

```
delete snap-pool volume
```

Parameter	Description
volume	Specifies the snap pool volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Example

Delete snap pool SP1.

```
# delete snap-pool SP1
```

Related Commands

- "show master-volumes" on page 180
- "show snap-pools" on page 194

delete snapshot

Description

Deletes a snapshot volume. All data uniquely associated with the snapshot is deleted and associated space in the snap pool is freed for use. Alias of `delete volume`.

Input

```
delete snapshot volume
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, "Volume Syntax" on page 25.

Example

Delete snapshot SS1.

```
# delete snapshot SS1
```

Related Commands

- "delete snapshot-write-data" on page 75
- "show snapshots" on page 196

delete snapshot-write-data

Description

Deletes the modified portion of a snapshot volume. This reverts the snapshot to the state when it was first taken.

Input

```
delete snapshot-write-data volume
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, "Volume Syntax" on page 25.

Example

Delete modified data from snapshot SS1.

```
# delete snapshot-write-data SS1
```

Related Commands

- "delete snapshot" on page 74
- "show snapshots" on page 196

delete task

Description

Deletes a specified task. If the task is scheduled, you must delete the schedule first.

Input

```
delete task task
```

Parameter	Description
task	Specifies the task name.

Example

Delete task Task1.

```
# delete task Task1

Success: Task Deleted
```

Related Commands

- "create task" on page 56
- "delete schedule" on page 72
- "show task-details" on page 201
- "show tasks" on page 205

delete user

Description

Deletes a user profile. By default a confirmation prompt appears, which requires a `yes` or `no` response. You can delete any user including the default users.

Input

```
delete user name [noprompt]
```

Parameter	Description
name	Specifies the user profile to delete. Names are case sensitive.
noprompt	Optional. Suppresses the confirmation prompt.

Example

Delete user `jsmith`.

```
# delete user jsmith

Are you sure? yes
Success: User jsmith deleted.
```

Related Commands

- "create user" on page 59
- "set user" on page 135
- "show users" on page 206

delete vdisk

Description

Deletes one or more virtual disks, disassociates all disk drives that are assigned to the virtual disks, and unmaps all volumes of the virtual disks from all host channels.

Caution – Deleting a virtual disk will delete all data on that virtual disk.

Note – You cannot delete a virtual disk that contains a snap pool associated with a master volume on another virtual disk. You cannot delete a virtual disk that is reconstructing.

Input

```
delete vdisk vdisks [prompt yes|no]
```

Parameter	Description
vdisks	Specifies the virtual disks to delete. For the syntax to use, "Virtual Disk Syntax" on page 25.
`prompt yes|no`	Optional. Specifies an automatic response to the prompt that appears if a utility is running on the vdisk: • `yes`: Stops the utility and enables the deletion to proceed • `no`: Prevents the deletion from proceeding If this parameter is omitted, you must manually reply to the prompt.

Example

Delete two virtual disks.

```
# delete vdisk VD1,VD2
```

Related Commands

- “show master-volumes” on page 180
- “show vdisks” on page 208

delete vdisk-spare

Description

Removes one or more spare disk drives that are assigned to a specified virtual disk.

Input

```
delete vdisk-spare disks disk vdisk
```

Parameter	Description
`disks` *`disk`*	Specifies the vdisk spares to remove. For the syntax to use, see "Disk Drive Syntax" on page 24.
`vdisk`	Specifies the virtual disk to remove the spare from. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Remove the vdisk spare having ID 5 from virtual disk VD1.

```
# delete vdisk-spare disks 5 VD1
```

Related Commands

- "show disks" on page 154
- "show vdisks" on page 208

delete volume

Description

Deletes a volume.

Input

```
delete volume volume
```

Parameter	Description
volume	Specifies the volume to delete. For the syntax to use, see "Volume Syntax" on page 25.

Example

Delete volume V1.

```
# delete volume V1
```

Related Commands

- "show volumes" on page 215

exit

Description

Exits the CLI session.

Syntax

```
exit
```

expand snap-pool

Description

Alias of `expand volume`. See “expand volume” on page 87. Expands a standard volume or snap-pool volume—but not a master volume—by a specified size. Expansion is restricted to the available space on the virtual disk that hosts the volume. If insufficient space is available for expansion on the virtual disk, first expand the virtual disk by using `expand vdisk`.

Note – To expand a master volume:
1. Delete all of its snapshots by using `delete all-snapshots`.
2. Convert it to a standard volume by using `convert master-to-std`.
3. Expand the standard volume by using `expand volume`.
4. Convert the expanded volume to a master volume by using `convert std-to-master`.

Input

To expand a volume to a specific size:

```
expand volume volume size sizeMB|GB|TB
```

To expand a volume to the maximum size:

```
expand volume volume size max
```

Parameter	Description
volume	Specifies the volume to expand. For the syntax to use, “Volume Syntax” on page 25.
`size` *size*`MB|GB|TB`	Specifies the volume size in `MB` (Mbyte), `GB` (Gbyte), or `TB` (Tbyte). The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, type `show cli-parameters`.
`size max`	Expands the volume to fill the available space on the virtual disk.

Example

Expand V1 by 100 Gbyte.

```
# expand volume V1 size 100GB
```

Related Commands

- “expand vdisk” on page 85
- “show snap-pools” on page 194
- “show volumes” on page 215

expand vdisk

Description

Adds disk drives to a virtual disk. Table 3-2 summarizes the expansion capability for each supported RAID level.

Table 3-2 Virtual Disk Expansion by RAID Level

RAID Level	Expansion Capability	Maximum Drives
NRAID (non-RAID)	Cannot expand.	1
0, 3, 5, 6	You can add 1–4 drives at a time.	16
1	Cannot expand.	2
10	You can add 2 or 4 drives at a time.	16
50	You can expand the virtual disk, one RAID 5 virtual subdisk at a time. The added RAID 5 virtual subdisk must contain the same number of drives as each of the existing virtual subdisks.	32

Caution – Virtual disk expansion cannot be stopped and can take days to complete, depending on drive type, RAID level, and other factors.

Input

```
expand vdisk vdisk disks disks
```

Parameter	Description
vdisk	Specifies the virtual disk by name or serial number. For the syntax to use, see “Virtual Disk Syntax” on page 25.
disks *disks*	Specifies the disk drives. For the syntax to use, see “Disk Drive Syntax” on page 24.

Example

Expand virtual disk VD1 to include the disk drives having ID 1, 2, and 3.

```
# expand vdisk VD1 disks 1-3
```

Related Commands

- “show disks” on page 154
- “show vdisks” on page 208

expand volume

Description

Expands a standard volume or snap-pool volume—but not a master volume—by a specified size. Expansion is restricted to the available space on the virtual disk that hosts the volume. If insufficient space is available for expansion on the virtual disk, first expand the virtual disk by using `expand vdisk`.

Note – To expand a master volume:
1. Delete all of its snapshots by using `delete all-snapshots`.
2. Convert it to a standard volume by using `convert master-to-std`.
3. Expand the standard volume by using `expand volume`.
4. Convert the expanded volume to a master volume by using `convert std-to-master`.

Input

To expand a volume to a specific size:

```
expand volume volume size sizeMB|GB|TB
```

To expand a volume to the maximum size:

```
expand volume volume size max
```

Parameter	Description
volume	Specifies the volume to expand. For the syntax to use, "Volume Syntax" on page 25.
`size` *size*`MB|GB|TB`	Specifies the volume size in `MB` (Mbyte), `GB` (Gbyte), or `TB` (Tbyte). The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, type `show cli-parameters`.
`size max`	Expands the volume to fill the available space on the virtual disk.

Example

Expand V1 by 100 Gbyte.

```
# expand volume V1 size 100GB
```

Related Commands

- “expand vdisk” on page 85
- “show snap-pools” on page 194
- “show volumes” on page 215

map volume

Description

Maps a volume to host channels and assigns a specified LUN to the mapping.

Input

```
map volume volume mapping mapping [host host]
[access read-write|rw|read-only|ro]
```

Parameter	Description
volume	Specifies the volume to map. For the syntax to use, see "Volume Syntax" on page 25.
mapping *mapping*	Specifies the host channels to map and LUNs to assign. For the syntax to use, see "Volume Mapping Syntax" on page 26.
host *host*	Optional. If this parameter is omitted, mapped volumes for all hosts are shown. For FC, this specifies the HBA's nickname or 16-hex-digit WWPN. For iSCSI, this specifies the IP address in IPv4 or IPv6 format or a nickname. For the nickname syntax to use, see "Host Nickname Syntax" on page 26.
access read-write|rw|read-only|ro	Optional. Specifies that the host has read-write (rw) or read-only (ro) access to the volume. The default is read-write.

Example

Map volume V1 to channel 1 and LUN 5 on no particular host and permit read-write access.

```
# map volume V1 mapping 1.5
```

Map volume V2 to channels 0-1 and LUN 2 on host HBA1 and permit read-only access.

```
# map volume V2 mapping 0-1.2 host HBA1 access ro
```

Related Commands

- “show host-maps” on page 169
- “show host-wwn-names” on page 175
- “show volumes” on page 215
- “show volume-maps” on page 213
- “unmap volume” on page 222

ping

Description

Tests communication with a remote host. The remote host is specified by IP address. Ping sends ICMP echo response packets and waits for replies.

Input

```
ping host-address [count]
```

Parameter	Description
host-address	Specifies the remote host's IP address in dotted decimal form.
count	Optional. Specifies the number of packets to send. The default is 4 packets. You should use a small count because the command cannot be interrupted.

Example

Send two packets to the remote computer at 10.0.0.1.

```
# ping 10.0.0.1 2

Pinging 10.0.0.1 with 2 packets.
Ping result: remote computer responded with 2 packets.
```

reset host-channel-link

Description

Issues a loop initialization primitive (LIP) from specified controllers on specified channels. This command is for use with an FC system using FC-AL (loop) topology.

Input

```
reset host-channel-link channel channels [controller a|b|both]
```

Parameter	Description
`channel` *channels*	Specifies a host channel number or a comma-separated list of host channel numbers such as 0,1.
`controller a|b|both`	Optional. Specifies whether to issue the LIP from controller A (upper), controller B (lower), or both. If this parameter is omitted, the default is the current controller.

Example

Reset the link on host channel 0 from controller A.

```
# reset host-channel-link channel 0 controller a

Success: Reset Host Channel Link(s) on channel(s) 0 from
controller(s) a.
```

Related Commands

- "show channels" on page 143

reset snapshot

Description

Deletes the data in a snapshot and resets it to the current data in the associated master volume. The snapshot's volume characteristics are not changed. The command prompts you to unmount the snapshot from the host operating system before performing the reset; leaving it mounted can cause data corruption.

Caution – All data represented by the snapshot as it exists prior to issuing this command is lost.

Input

```
reset snapshot volume [prompt yes|no]
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`prompt yes|no`	Optional. Specifies an automatic response to the unmount prompt that either enables the reset to proceed or prevents the reset from proceeding: • `yes`: Enables the reset to proceed • `no`: Prevents the reset from proceeding If this parameter is omitted, you must reply to the prompt.

Example

Reset snapshot SS1.

```
# reset snapshot SS1

Leaving the snapshot mounted during reset on any operating system
can result in data corruption. Is the snapshot un-mounted from all
operating systems? yes
Success: Snapshot Reset Complete.
```

Related Commands

- “show snapshots” on page 196

restart

Description

Restarts the RAID controller or the management controller in either or both controller modules.

If you restart a RAID controller, it attempts to shut down with a proper failover sequence, which includes stopping all I/O operations and flushing the write cache to disk, and then the controller restarts. The management controllers are not restarted so they can provide status information to external interfaces.

If you restart a management controller, communication with it is temporarily lost until it successfully restarts. If the restart fails, the partner management controller remains active with full ownership of operations and configuration information.

Caution – If you restart both controller modules, you and users lose access to the system and its data until the restart is complete.

Input

```
restart sc|mc a|b|both [noprompt}
```

Parameter	Description
`sc|mc`	Specifies the controller to restart: • `sc`: RAID (storage) controller • `mc`: Management controller
`a|b|both`	Specifies whether to restart controller A (upper), controller B (lower), or both.
`noprompt`	Optional. Suppresses the confirmation prompt.

Output

Messages are displayed when the controllers are shut down, when failover is initiated, and when the controllers have been restarted.

Example

Restart the management controller in controller module A.

```
# restart mc a
During the restart process you will briefly lose communication with
the specified management Controller(s).
Continue? y
Info: Restarting Local MC A...
#
```

Related Commands

- "shutdown" on page 218

restore defaults

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Restores the manufacturer's default configuration to the controllers. When the command informs you that the configuration has been restored, you must restart the RAID controllers and management controllers for the changes to take effect. After restarting the controllers, hosts might not be able to access volumes until you re-map them.

Caution – This command changes how the system operates and might require some reconfiguration to restore host access to volumes.

Input

```
restore defaults [noprompt]
```

Parameter	Description
`noprompt`	Optional. Suppresses the confirmation prompt.

Related Commands

- “map volume” on page 89
- “restart” on page 95
- “show host-maps” on page 169

rollback master-volume

Description

Rolls back (reverts) the data on a master volume to the data that exists in a specified snapshot. You can choose whether to whether to include modified write data from the snapshot in the rollback. You must unmount the master volume from the host operating system before using this command. The command will prompt you to ensure the master volume is unmounted before proceeding.

Caution – All data that differs between the master volume and the snapshot is lost. Create a snapshot of the master volume as it currently exists before performing a rollback.

Input

```
rollback master-volume volume snapshot volume [modifiedsnapshot
yes|no] [prompt yes|no]
```

Parameter	Description
volume	Specifies the master volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
snapshot *volume*	Specifies the snapshot volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
modifiedsnapshot yes|no	Optional. Specifies whether to include or exclude modified write data from the snapshot in the rollback. • yes: Include modified snapshot data. • no: Exclude modified snapshot data. If this parameter is omitted, modified snapshot data is excluded.

Parameter	Description
`prompt yes|no`	Optional. Specifies an automatic response to the unmount prompt that either enables the rollback to proceed or prevents the rollback from proceeding. • `yes`: Enable the rollback to proceed • `no`: Prevent the rollback from proceeding If this parameter is omitted, you must reply to the prompt.

Example

Roll back master volume MV1 to snapshot SS1.

```
# rollback master-volume MV1 snapshot SS1
```

Related Commands

- "show master-volumes" on page 180
- "show snapshots" on page 196

scrub vdisk

Description

For specified RAID 3, 5, 6, and 50 virtual disks, the command verifies all parity blocks and corrects any bad parity. For specified RAID 1 and 10 virtual disks, the command compares the primary and secondary drives and if it finds a mismatch it copies the primary to the secondary.

In addition, this command scrubs virtual disk metadata and stripe and NRAID virtual disks. When scrub of a virtual disk is complete, an event is logged specifying the number of errors found.

Input

```
scrub vdisk vdisks
```

Parameter	Description
vdisks	Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Scrub the disk drives in virtual disk `VD1`.

```
# scrub vdisk VD1
```

Related Commands

- "abort scrub" on page 39
- "verify vdisk" on page 224

set auto-write-through-trigger

Description

Sets the trigger conditions that cause the controller to change the cache policy from write-back to write-through. By default, automatic write-back is enabled. Alias: `set awt`.

When the cache mode is changed, an event is logged.

Input

```
set auto-write-through-trigger controller-failure
|super-cap-failure|power-supply-failure
|fan-failure|temperature-exceeded|partner-notify|auto-write-back
enable|disable [...]
```

Parameter	Description
`controller-failure enable|disable`	If the cache policy is set to write-back, specify whether the policy automatically changes to write-through when a controller fails.
`super-cap-failure enable|disable`	If the cache policy is set to write-back, specify whether the policy automatically changes to write-through when cache backup power is not fully charged or fails.
`compact-flash-failure enable|disable`	If the cache policy is set to write-back, specify whether the policy automatically changes to write-through when the compact flash fails.
`power-supply-failure enable|disable`	If the cache policy is set to write-back, specify whether the policy automatically changes to write-through when a power supply fails.

Parameter	Description
`fan-failure enable|disable`	If the cache policy is set to write-back, specify whether the policy automatically changes to write-through when a fan fails.
`temperature-exceeded enable|disable`	Specify whether to force a controller shutdown if a temperature is detected that exceeds system threshold limits.
`partner-notify enable|disable`	In a dual-controller configuration, specify whether to notify the partner controller that the trigger condition is met.
`auto-write-back enable|disable`	Specify whether the cache mode automatically changes to write-back after the trigger condition is cleared.

Example

Enable the `controller-failure` trigger and disable the `partner-notify` trigger.

```
# set auto-write-through-trigger controller-failure enable
partner-notify disable

Success: Auto-Write-Through-Trigger parameters have been changed
```

Related Commands

- "show auto-write-through-trigger" on page 140
- "show events" on page 162

set cache-parameters

Description

Sets the RAID controller cache policy and cache optimization mode for a specified volume.

Note – Only change the read-ahead cache settings if you fully understand how your operating system, application, and HBA move data so that you can adjust the settings accordingly. Be prepared to monitor system performance using the virtual disk statistics and adjust read-ahead size until you find the optimal size for your application.

Cache Policy

The cache policy setting for each volume determines when cached data is written to the disk drives. The ability to hold data in cache while it is being written to disk can increase storage device speed during sequential reads.

- Write-back caching does not wait for data to be completely written to disk before signaling the host that the write is complete. This is the preferred setting for a fault-tolerant environment because it improves the performance of write operations and throughput. Write-back caching is enabled by default.
- Write-through caching significantly impacts performance by waiting for data to be completely written to disk before signaling the host that the write is complete. Use this setting only when operating in an environment with low or no fault tolerance.

You can configure the write policy to automatically change from write-back cache to write-through cache when certain environmental events occur, such as a fan failure. For details, see “set auto-write-through-trigger” on page 101.

Cache Optimization Mode

Before creating or modifying a volume, determine the appropriate cache optimization mode. The controller supports `super-sequential` optimization mode for sequential I/O and `standard` optimization mode for random I/O.

The cache optimization mode setting for each volume optimizes the cache block size used by the controller:

- For sequential optimization, the cache block size is 128 Kbyte.
- For random optimization, the cache block size is 32 Kbyte.

An appropriate cache block size improves performance when a particular application uses either large or small stripe sizes:

- Video playback, multimedia post-production audio and video editing, and similar applications read and write large files in sequential order.
- Transaction-based and database update applications read and write small files in random order.

Since the cache block size works in conjunction with the default stripe size set by the cache optimization mode for each volume you create, these default stripe sizes are consistent with the cache block size setting. You can, however, specify a different stripe size for any volume at the time you create it. For more information, see "create volume" on page 64.

Input

```
set cache-parameters [independent enable|disabled] [volume]
[write-policy write-back|write-through] [optimization
standard|super-sequential] [read-ahead-size size]
```

Parameter	Description
`independent enable|disable`	Optional. Sets independent-cache mode. Write-back cache is not mirrored to the other controller when independent mode is enabled.
volume	Optional. Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`write-policy write-back|write-through`	Optional. Sets the cache policy; see "Cache Policy" on page 103.
`optimization standard|super-sequential`	Optional. Sets the cache optimization mode; see "Cache Optimization Mode" on page 103.

Parameter	Description
`read-ahead-size` *size*	Optional. Specifies the size of the read-ahead cache. Valid values are: • `Disabled`: Turns off read-ahead cache. • `Default`: Sets one chunk for the first access in a sequential read and one stripe for all subsequent accesses. • `Maximum`: Allows the controller to dynamically calculate the maximum read-ahead cache size for the volume. • `64KB`, `128KB`, `512KB`, `1MB`, `2MB`, `4MB`, `8MB`, `16MB`, `32MB`

Example

Set the cache policy and cache optimization mode for volume V1.

```
# set cache-parameters V1 write-policy write-back optimization
super-sequential
```

Related Commands

- "show cache-parameters" on page 141
- "show volumes" on page 215

set cli-parameters

Description

Sets the CLI timeout, the output characteristics of the show commands, and the way that the system handles some errors.

If you are accessing the CLI through the Ethernet port, settings apply to the current CLI session only. If you are accessing the CLI through the enclosure's CLI port, settings persist across sessions.

Input

```
set cli-parameters [console|api] [base 2|10] [pager enabled|on
|disabled|off] [timeout #]
```

Parameter	Description
`console|api`	Optional. Enables `show` commands to be shown as human-readable console output or as XML output. `console` is the default. api sets the output to XML.
`base 2|10`	Optional. Specifies whether capacities are calculated and shown in base 2 or base 10: • `base 2`: Sizes are represented as powers of two and 1024 is used as a divisor for each magnitude. • `base 10`: Sizes are represented as powers of ten and 1000 is used as a divisor for each magnitude. Disk-drive vendors typically use base 10 for their indication of capacity. This is the default.
`pager enable|disable`	Optional. Specifies whether to halt output after each full screen to wait for keyboard input. This is enabled by default.
`timeout #`	Optional. Sets the timeout value for the login session. Valid values are 10–9999 seconds; the default is 1800 seconds (30 minutes).

Example

Set CLI to use a 30-second timeout, base 2, and no paging.

```
# set cli-parameters timeout 30 base 2 pager disable
```

Related Commands

- “show cli-parameters” on page 146

set controller-date

Description

Sets the date and time of the management controller and then updates the date and time for each RAID controller.

Input

```
set controller-date jan|feb|mar|apr|may|jun|jul|aug|sep|oct
|nov|dec day hh:mm:ss year time-zone
```

Parameter	Description
jan|feb|mar|apr|may|jun|jul|aug|sep|oct|nov|dec	Specifies the month.
day	Specifies the day number (1–31).
hh:*mm*:*ss*	Specifies the hour on a 24-hour clock (0–23), the minute (0-59), and the seconds (0–59).
year	Specifies the year as a four-digit number.
time-zone	Specifies the system's time zone based on Greenwich Mean Time (GMT).

Example

Set the controller date to 1:45 PM on September 22, 2006 in the U.S. Mountain time zone (GMT -07:00).

```
# set controller-date sep 22 13:45:0 2006 -7
```

Related Commands

- "show controller-date" on page 149

set debug-log-parameters

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Sets the types of debug messages to include in the storage controller debug log. If multiple types are specified, use spaces to separate them and enclose the list in quotation marks (").

Input

```
set debug-log-parameters message-type+|- [...] standard
```

Parameter	Description
message-type+|-	Specifies one of the following message types, followed by a plus (+) to enable or a minus (-) to disable inclusion in the log: • `awt`: Auto-write-through feature debug messages • `bkcfg`: Internal configuration debug messages • `cache`: Cache debug messages • `capi`: Internal CAPI debug messages • `capi2`: Internal CAPI tracing debug messages • `disk`: Disk interface debug messages • `dms`: Snapshot feature debug messages • `emp`: Enclosure Management Processor debug messages • `fca`: Fibre Channel driver debug messages (level A) • `fcb`: Fibre Channel driver debug messages (level B) • `fcc`: Fibre Channel driver debug messages (level C) • `fcd`: Fibre Channel driver debug messages (level D) • `fo`: Failover/recovery debug messages • `host`: Host interface debug messages • `mem`: Internal memory debug messages • `misc`: Internal debug messages • `msg`: Inter-controller message debug messages • `mui`: Internal service interface debug messages • `raid`: RAID debug messages • `rcm`: Removable-component manager debug messages • `res2s`: Internal debug messages

Parameter	Description

`standard`	Sets the messages to:
	`host: on`
	`disk: on`
	`mem: off`
	`fo: on`
	`msg: on`
	`fca: on`
	`fcb: off`
	`fcc: off`
	`fcd: off`
	`misc: on`
	`rcm: off`
	`raid: on`
	`cache: on`
	`emp: on`
	`capi: on`
	`mui: on`
	`bkcfg: on`
	`awt: off`
	`res2: off`
	`capi2: off`
	`dms: on`

Example

Include RAID and cache messages, exclude EMP messages, and leave other message types unchanged.

```
# set debug-log-parameters "raid+ cache+ emp-"
```

Related Commands

- "show debug-log-parameters" on page 152
- "show debug-log" on page 150

set drive-parameters

Description

Selects a global disk Self-Monitoring, Analysis and Reporting Technology (SMART) setting. Disk drives equipped with this technology can alert the controller of impending drive failures. Changes to the SMART setting take effect after a rescan or a controller reboot.

Input

```
set drive-parameters smart enable|disable|detect-only
```

Parameter	Description
`smart enable|disable|detect-only`	Enables or disables SMART monitoring. `detect-only` specifies that drives added after this command executes retain their SMART setting. SMART is enabled by default.

Example

Set the SMART setting to enable.

```
# set drive-parameters smart enable
```

Related Commands

- "show drive-parameters" on page 157

set expander-fault-isolation

Description

When fault isolation is enabled, the expander controller will isolate PHYs that fail to meet certain criteria. When fault isolation is disabled, the errors are noted in the logs but the PHYs are not isolated.

Input

```
set expander-fault-isolation wwn enclosure [controller a|b|both]
enable|disable
```

Parameter	Description
`wwn` *`enclosure`*	Specifies the 16-hex-digit logical identifier of the controller enclosure or expansion enclosure containing the PHY to be enabled or disabled. The enclosure logical identifier is found in the WWPN field in the output of the `show enclosure-status` command.
`controller a|b|both`	Optional. Specifies whether the expander is connected to controller A (upper), controller B (lower), or both.
`enable|disable`	Specifies whether to enable or disable the PHY in the SAS expander(s).

Example

Enable checking of the PHYs on both controllers at WWPN `500C0FF0ABCD343C`.

```
# set expander-fault-isolation wwn 500C0FF0ABCD343C controller
both enable
```

Related Commands

- “set expander-phy” on page 114
- “show expander-status” on page 165
- “show enclosure-status” on page 158

set expander-phy

Description

The expander controller will enable or disable (isolate) the specified PHY.

Input

```
set expander-phy wwn enclosure controller a|b|both index phy-
number enable|disable
```

Parameter	Description
`wwn` *`enclosure`*	Specifies the 16-hex-digit logical identifier of the controller enclosure or expansion enclosure containing the PHY to be enabled or disabled. The enclosure logical identifier is found in the WWPN field in the output of the `show enclosure-status` command.
`controller a|b|both`	Specifies whether to apply the settings to controller A (upper), controller B (lower), or both.
`index` *`phy-number`*	Specifies the logical PHY number, 0 to 23.
`enable|disable`	Specifies whether to enable or disable the specified PHY.

Example

Disable PHY 23 on both controllers at WWPN address `500C0FF0ABCD343C`.

```
# set expander-phy wwn 500C0FF0ABCD343C controller both index 23
disable

Success: Disabled Phy #23.
```

Related Commands

- “set expander-fault-isolation” on page 113
- “show expander-status” on page 165
- “show enclosure-status” on page 158

set global-spare

Description

Designates spare disk drives that can be used by any virtual disk. Sets the disk drive status to `global-spare`.

Input

```
set global-spare disks disks
```

Parameter	Description
`disks` *disks*	Specifies the disk drives. For the syntax to use, see "Disk Drive Syntax" on page 24.

Example

Designate the disk drive having ID 5 as a global spare.

```
# set global-spare disks 5
```

Related Commands

- "delete global-spare" on page 68
- "show disks" on page 154

set host-parameters

Description

Sets the link speed and topology for host channels on specified controllers. The new settings take effect without restarting the RAID controllers.

Input

```
set host-parameters [controller a|b|both] [speed speed] [channel
#|all][fibre-connection-mode loop|point-to-point] [IP address]
[netmask address] [gateway address] [noprompt]
```

Parameter	Description
`controller a|b|both`	Optional. Specifies whether to apply the settings to controller A (upper), controller B (lower), or both. Required if either `speed` or `fibre-connection-mode` is specified.
`speed` *speed*	Optional. Sets the link speed in Gbit/sec, which automatically changes for the partner host channel on the other controller. Valid values for FC: 2g, 4g. Valid values for iSCSI: 10mb, 100mb, 1g. If you change a parameter on controller A channel 0, then controller B channel 1 also changes. The same is true for A1 and B0.
`channel #|all`	Optional. Specifies a host channel number. If this parameter is omitted, all host channels on the specified controllers are affected.
`fibre-connection-mode loop|point-to-point`	Optional. FC only. Sets the topology for the specified channels on the specified controllers to either: • `loop`: Fibre Channel-Arbitrated Loop (public or private). • `point-to-point`: Fibre Channel point-to-point. Point-to-point topology is only valid and allowed when port interconnects are disabled (using `set host-port-interconnects`).
`IP` *address*	Optional. iSCSI only. Specifies the IP address for an iSCSI port in IPv4 format.

Parameter	Description
`netmask` *`address`*	Optional. iSCSI only. Specifies the netmask address in IPv4 format.
`gateway` *`address`*	Optional. iSCSI only. Specifies the gateway address in IPv4 format.
`noprompt`	Optional. Specifies an automatic response to the confirmation prompt that enables the operation to proceed. If this parameter is omitted, you must reply to the prompt.

Example

Set the FC link speed to 2 Gbit/sec for channel 1 on controller A and for channel 0 on controller B.

```
# set host-parameters controller a speed 2g channel 1
WARNING:  You have requested a change on host channel parameters.
This change will take effect immediately. If the link speed is
changed, your current link may go down. If the topology is changed
between 'loop' and 'point-to-point', you may not be able to use
dual ID mode for failover and your ability to access LUNs from some
hosts will be limited. Are you sure?
  yes
```

Configure the topology of all host channels on controller A to loop.

```
# set host-parameters channel all fibre-connection-mode loop
controller a
WARNING:  You have requested a change on host channel parameters.
This change will take effect immediately. If the link speed is
changed, your current link may go down. If the topology is changed
between 'loop' and 'point-to-point', you may not be able to use
dual ID mode for failover and your ability to access LUNs from some
hosts will be limited. Are you sure?
  yes
```

Configure the iSCSI port IP address, netmask and gateway for channel 0 on controller A.

```
# set host-parameters channel 0 ip 192.168.0.10 netmask
255.255.255.0 gateway 192.168.0.1 controller a
WARNING:  You have requested a change on host channel parameters.
This change will take effect immediately. If the link speed is
changed, your current link may go down. If the topology is changed
between 'loop' and 'point-to-point', you may not be able to use
dual ID mode for failover and your ability to access LUNs from some
hosts will be limited. Are you sure?
  yes
```

Related Commands

- “show host-parameters” on page 171
- “show host-port-interconnects” on page 174

set host-port-interconnects

Description

Enables or disables the internal connection between FC host port pairs. In a dual-controller system, each host port on the upper controller is connected to a peer host port on the lower controller. When the connection between peer host ports is enabled, it forms an FC-AL loop. This command affects the connections between all host port pairs.

In order to enable the connections, the host port fibre connection mode must be set to loop.

Input

```
set host-port-interconnects enable|disable
```

Example

Enable the connections between all host port pairs.

```
# set host-port-interconnects enable
```

Related Commands

- "show host-parameters" on page 171
- "show host-port-interconnects" on page 174

set host-wwn-name

Description

Changes the nickname of an HBA entry for a host world wide port name (WWPN), or associates a nickname with an unnamed HBA entry.

Input

```
set host-wwn-name host wwn|nickname new-nickname
```

Parameter	Description
host *wwn*|*nickname*	Specifies an existing 16-hex-digit WWPN or an existing nickname that corresponds to an HBA. For the nickname syntax to use, see "Host Nickname Syntax" on page 26.
new-nickname	Specifies a new nickname for the HBA. For the nickname syntax to use, see "Host Nickname Syntax" on page 26.

Example

Create the nickname HBA1 for the HBA having WWPN 100000A0B8040BAC.

```
# set host-wwn-name host 100000A0B8040BAC HBA1
```

Change the nickname HBA1 to Test-HBA.

```
# set host-wwn-name host HBA1 Test-HBA
```

Related Commands

- "create host-wwn-name" on page 48
- "delete host-wwn-name" on page 69
- "show host-wwn-names" on page 175

set iscsi-host

Description

Changes the nickname associated with an iSCSI host initiator's IP address, or associates a nickname with an unnamed iSCSI host initiator’s IP address.

Input

```
set iscsi-host descriptor new-nickname
```

Parameter	Description
descriptor	Specifies an iSCSI host initiator’s IP address in IPv4 format, or nickname. For the nickname syntax to use, see “Host Nickname Syntax” on page 26.
new-nickname	Specifies a new nickname for the iSCSI host initiator. For the nickname syntax to use, see “Host Nickname Syntax” on page 26.

Example

Change the nickname iSCSI-1 to iSCSI HBA-1.

```
# set iscsi-host iSCSI-1 "iSCSI HBA-11"
```

Related Commands

- “create iscsi-host” on page 49
- “delete iscsi-host” on page 70
- “show iscsi-hosts” on page 176

set job-parameters

Description

Sets parameters for background scrub, partner firmware upgrade, and other jobs.

Input

```
set job-parameters [background-scrub enabled|on|disabled|off] |
[partner-firmware-upgrade enabled|on|disabled|off] |
[utility-priority low|medium|high]
```

Parameter	Description
`background-scrub enabled|on|disabled|off`	Optional. Controls whether vdisks are checked for disk drive defects to ensure system health.
`partner-firmware-upgrade enabled|on|disabled|off`	Optional. Controls whether versions of firmware components are monitored and automatically updated on the partner controller.
`utility-priority low|medium|high`	Optional. Sets the priority at which jobs run with respect to I/O operations competing for the system's processors. This affects vdisk verification and reconstruction, but not background scrub.

Example

Disable background scrubbing of vdisks.

```
# set job-parameters background-scrub off
```

Set the job priority to low.

```
# set job-parameters utility-priority low
```

Related Commands

- "show job-parameters" on page 177

set led

Description

Changes the state of drive module or enclosure LEDs to help you locate devices. For a drive module, the Power/Activity/Fault LED will blink yellow. For an enclosure, the Unit Locator LED on the chassis ear and on each controller module will blink white.

Input

```
set led disk|enclosure ID on|off
```

Parameter	Description
`disk` *ID*	Specifies a disk drive number.
`enclosure` *ID*	Specifies the enclosure number, where 0 represents the controller enclosure and the number increments for each attached expansion enclosure.
`on|off`	Specifies to set or unset the LED.

Example

Set the Power/Activity/Fault LED on drive module 5.

```
# set led disk 5 on

Success: Identifying disk 5...
```

Unset the Unit Locator LEDs on the controller enclosure.

```
# set led enclosure 0 off

Success: Identifying enclosure 0...
```

set network-parameters

Description

Sets the enclosure's Ethernet parameters using a dynamic or static method.

Input

To set both controllers' Ethernet parameters using Dynamic Host Configuration Protocol (DHCP):

```
set network-parameters [dhcp]
```

To set a controller's Ethernet parameters statically:

```
set network-parameters [ip address] [netmask netmask]
[gateway gateway] [controller a|b]
```

Parameter	Description
`dhcp`	Optional. Specifies to use DHCP to determine the enclosure's IP address.
`ip` *`address`*	Optional. Specifies the enclosure's IP address (*`xxx.xxx.xxx.xxx`*).
`netmask` *`netmask`*	Optional. Specifies the enclosure's IP netmask (*`xxx.xxx.xxx.xxx`*).
`gateway` *`gateway`*	Optional. Specifies the enclosure's IP gateway (*`xxx.xxx.xxx.xxx`*).
`controller a|b`	Optional. Specifies whether to apply the settings to controller A (upper) or controller B (lower). If this parameter is omitted, settings are applied to the controller being accessed.

Example

Dynamically set enclosure network parameters using DHCP.

```
# set network-parameters dhcp
```

Statically set network parameters for controller A.

```
# set network-parameters ip 192.168.0.10 netmask 255.255.255.0
gateway 192.168.0.1 controller a
```

Related Commands

- “show network-parameters” on page 182

set password

Description

Sets a user's password for system interfaces (such as the CLI). The command prompts you to enter and re-enter the new password. A password can include a maximum of 19 printable ASCII characters except backslash (\), quotation mark ("), apostrophe ('), or spaces. The password is case sensitive.

Input

```
set password [user]
```

Parameter	Description
user	Optional. Specifies the login name of the user to set the password for. If this argument is omitted, this command affects the logged-in user's password.

Example

Change the password of the default user, `monitor`.

```
# set password monitor

Info: Changing password for user: monitor
Enter new password:****
Verify new password:****
Success: Password set
```

set prompt

Description

Sets the prompt for the current CLI session.

Input

```
set prompt prompt
```

Parameter	Description
prompt	Specifies the new prompt string. A string that includes a space must be enclosed in quotation marks (").

Output

Command line with new prompt.

Example

Set the prompt to `> `.

```
# set prompt "> "
>
```

set protocols

Description

Enables or disables one or more service and security protocols.

Input

```
set protocol [capi|debug|ftp|http|https|service|ses|smis|snmp
|ssh|telnet enabled|disabled] [...]
```

Parameter	Description
`capi|debug|ftp|http|https|service|ses|smis|snmp|ssh|telnet`	Optional. Specifies the protocol to enable or disable: • `capi`: In-band CAPI management interface • `debug`: Telnet debug port 4048 • `ftp`: Firmware upgrade interface • `http`: WBI web server • `https`: Secure WBI web server • `service-interface`: Telnet service port 1023 • `ses`: In-band SES management interface • `smis`: Storage Management Initiative Specification • `snmp`: Simple Network Management Protocol • `ssh`: Secure shell CLI • `telnet`: Standard CLI service
`enabled|disabled`	Specifies whether to enable or disable the preceding protocol.

Example

Disable unsecure HTTP connections and enable FTP.

```
# set protocols http disabled ftp enabled
```

Related Commands

- "show protocols" on page 185

set snap-pool-policy

Description

Sets the recovery policy that determines the action taken when a specified snap pool's error and critical threshold levels are reached. The policy for the warning threshold is preset to `notifyonly`. A snap pool's default error policy is `deleteoldestsnapshot` and default critical policy is `deletesnapshots`.

Input

```
set snap-pool-policy volume [error autoexpand
|deleteoldestsnapshot|deletesnapshots|haltwrites|nochange
|notifyonly] [critical deleteoldestsnapshot|deletesnapshots
|haltwrites|nochange] [autoexpansionsize sizeMB|GB|TB]
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`error autoexpand` `|deleteoldestsnapshot` `|deletesnapshots|haltwrites` `|notifyonly|nochange`	Optional. Specifies the recovery policy to invoke when the error threshold level of snap-pool usage is reached. • `autoexpand`: Automatically expand the snap pool using the `autoexpansionsize` value. • `deleteoldestsnapshot`: Delete the oldest snapshot in the snap pool. • `deletesnapshots`: Delete all snapshots in the snap pool. • `haltwrites`: Halt writes to the snap pool. • `nochange`: Take no action. • `notifyonly`: Generates an event to notify the administrator.
`critical deleteoldestsnapshot` `|deletesnapshots|haltwrites` `|nochange`	Optional. Specifies the recovery policy to invoke when the critical threshold level of snap-pool usage is reached.

Parameter	Description
autoexpansionsize *size*MB|GB|TB	Specifies the increment in MB (Mbyte), GB (Gbyte), or TB (Tbyte) by which the snap pool will be automatically expanded when the threshold level is reached. The size uses base 10 (multiples of 1000) or base 2 (multiples of 1024); to see the current base setting, use show cli-parameters.

Example

Set the error policy to autoexpand and the automatic expansion size to 10 MB for snap pool SP1.

```
# set snap-pool-policy SP1 error autoexpand autoexpansionsize 10MB
```

Related Commands

- "set snap-pool-threshold" on page 131
- "show snap-pools" on page 194

set snap-pool-threshold

Description

Sets the percentages of snap-pool space used that trigger the warning and error threshold policies. Three threshold levels are defined:

- **Warning.** This first threshold indicates that snap-pool space is moderately full. When exceeded, an event is generated to warn the administrator.
- **Error.** This second threshold indicates that the snap pool is getting full and unless corrective action is taken, snapshot data loss is probable. When exceeded, an event is generated to warn the administrator and the associated snap-pool policy is triggered.
- **Critical.** This third threshold indicates that the snap pool is nearly full and that data loss is imminent. When exceeded, an event is generate to alert the administrator and the associated snap-pool policy is triggered. This threshold is preset to 99%.

Input

```
set snap-pool-threshold volume [warning #%] [error #%]
```

Parameter	Description
`volume`	Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25.
`warning #%`	Specifies the percent of snap-pool space used that triggers the warning threshold policy. This value must be less than the error threshold value.
`error #%`	Specifies the percent of snap-pool space used that triggers the error threshold policy. This value must be less than 99%.

Example

Set the warning and error thresholds for snap pool SP1.

```
# set snap-pool-threshold SP1 warning 60% error 85%
```

Related Commands

- “set snap-pool-policy” on page 129
- “show snap-pools” on page 194

set snmp-parameters

Description

Configures SNMP notification of events.

Input

```
set snmp-parameters [enable crit|warn|info|none]
[add-trap-host IP] [del-trap-host IP] [read-community string]
[write-community string]
```

Parameter	Description
`enable crit|warn|info|none`	Optional. Sets the level of trap notification: • `crit`: Sends critical events only. • `warn`: Sends all critical events and warnings. • `info`: Sends all events. • `none`: All events are excluded from trap notification and traps are disabled.
`add-trap-host` *IP*	Optional. Adds a destination host to send traps to. Three trap hosts can be set.
`del-trap-host` *IP*	Optional. Deletes a trap destination host.
`read-community` *string*	Optional. Sets an alphanumeric community string for read-only access.
`write-community` *string*	Optional. Sets an alphanumeric community string for write access.

Example

Enable critical events only, specify a trap host, and set the community string for read-only access.

```
# set snmp-parameters enable crit add-trap-host 172.22.4.171
read-community public
```

Related Commands

- "show snmp-parameters" on page 198

set system

Description

Sets the name, contact person, location, and description for a system. Each value can include a maximum of 79 printable ASCII characters except quotation mark (") or backslash (\). If the value contains a space, enclose the value in quotation marks.

Input

```
set system [name value] [contact value] [location value]
[info value]
```

Parameter	Description
`name` *value*	Specifies the system's name.
`contact` *value*	Specifies the system's contact person.
`location` *value*	Specifies the system's location.
`info` *value*	Specifies information about the system.

Example

Set the system name to `Test` and the contact to `J. Doe`.

```
# set system name Test contact "J. Doe"
```

Related Commands

- "show system" on page 200

set user

Description

Changes a user's level, type, and interface access. To change a user's password, use `set password`.

Input

```
set user name [level monitor|manage] [type standard|advanced
|diagnostic] [interfaces interfaces]
```

Parameter	Description
name	Specifies the user profile to change. Names are case sensitive.
`level monitor|manage`	Optional. Specifies whether the user has `monitor` (view-only) or `manage` (modify) access for user interfaces. The default is `monitor`.
`type standard|advanced |diagnostic`	Optional. Specifies the user's level of technical expertise, to control access to functionality in the WBI. • `standard`: Enables access to standard administrative functions. This is the default for monitor users. • `advanced`: Enables access to standard and advanced functions. This is the default for manage users. • `diagnostic`: Enables access to standard, advanced, and troubleshooting functions. This is the default for users of the CLI.
`interfaces` *interfaces*	Optional. Specifies the interfaces that the user can access. Multiple values must be separated by a comma with no spaces. If this parameter is omitted, the defaults are `cli` and `wbi`. • `cli`: Command-line interface. • `wbi`: Web-browser interface. • `ftp`: File transfer protocol interface. • `none`: No interfaces.

Example

Change a user's type and interface access.

```
# set user jsmith type diagnostic interfaces wbi,cli

Success: User-Type set to diagnostic.
Success: CLI-Access enabled.
Success: WBI-Access enabled.
```

Related Commands

- “create user” on page 59
- “delete user” on page 77
- “set password” on page 126
- “show users” on page 206

set vdisk

Description

Sets the name and owning controller for an existing virtual disk.

Input

```
set vdisk vdisk [name new-name] [owner a|b]
```

Parameter	Description
vdisk	Specifies the virtual disk by name or serial number. For the syntax to use, see “Virtual Disk Syntax” on page 25.
name *new-name*	Optional. Specifies a new name for the virtual disk. For the syntax to use, see “Virtual Disk Syntax” on page 25.
owner a|b	Optional. Sets the owner of the virtual disk to controller A (upper) or controller B (lower).

Example

Rename virtual disk VD1 and reassign its ownership.

```
# set vdisk VD1 name VD2 owner a
```

Related Commands

- “show vdisks” on page 208

set vdisk-spare

Description

Designates spare disk drives that can be used by a specific virtual disk, and sets the disk drive status to `VDISK SP`. A vdisk spare can be assigned to RAID 1, 3, 5, 6, 10, and 50 virtual disks, and must have enough capacity to replace the smallest disk drive in the virtual disk.

Input

```
set vdisk-spare disks disks vdisk
```

Parameter	Description
`disks` *disks*	Specifies up to four disk drives. For the syntax to use, see "Disk Drive Syntax" on page 24.
vdisk	Specifies the virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Designate the disk drive having ID 5 as a vdisk spare for virtual disk VD1.

```
# set vdisk-spare disks 5 VD1
```

Related Commands

- "show disks" on page 154
- "show vdisks" on page 208

set volume

Description

Changes the name of a volume.

Input

```
set volume volume name new-name
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25.
name *new-name*	Specifies a new name for the volume. For the syntax to use, see “Volume Syntax” on page 25.

Example

Rename volume V1 to V2.

```
# set volume V1 name V2
```

Related Commands

- “map volume” on page 89
- “show volumes” on page 215
- “show host-maps” on page 169
- “show volume-maps” on page 213

show auto-write-through-trigger

Description

Shows the system's write-through trigger settings. When a trigger condition occurs and the trigger is enabled, the RAID controller cache mode changes from write-back to write-through. Alias: `show awt`.

Input

```
show auto-write-through-trigger
```

Output

Shows whether each trigger is enabled or disabled.

Example

Show the system's trigger settings.

```
# show auto-write-through-trigger

Controller Failure    : Disabled
SuperCap Failure      : Enabled
Compact Flash Failure : Disabled
Power Supply Failure  : Disabled
Fan Failure           : Disabled
Temperature Exceeded  : Disabled
Partner Notify        : Disabled
Auto Write Back       : Enabled
```

Related Commands

- "set auto-write-through-trigger" on page 101

show cache-parameters

Description

Shows a volume's cache policy and cache optimization mode.

Input

```
show cache-parameters volume
```

Parameter	Description
volume	Optional. Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Output

Field	Description
Operation Mode	The volume's operation mode. Possible values are: Unknown, Standalone Single Port, Standalone Dual Port, Active-Active Single Port, Active-Active, Active Pass Dual Port, A-A Dual Port Multi ID, or Independent Cache Performance Mode.
Volume Serial	Volume serial number
Volume Name	Volume name
Write-Mode	The volume's cache policy, either write-back or write-through
Optimization	The volume's cache optimization mode, either standard (random) or super-sequential
Read Ahead Size	The size of the read-ahead cache

Example

Show the cache parameters for volume V1.

```
# show cache-parameters V1
Cache Parameters
----------------
Operation Mode: Active-Active

Volume Serial : 00c0ff0a48880048fd83424601000000
Volume Name   : V1
Write-mode    : write-back
Optimization  : standard
Read Ahead Size: Maximum
```

Related Commands

- "set cache-parameters" on page 103
- "show volumes" on page 215

show channels

Description

Shows information about channels on both controllers.

Input

```
show channels
```

Output

Field	Description
Ctlr	Controller ID
Ch	Channel number
Type	• Host: Channel to host • Drive: Channel to expansion enclosures
Media	• FC(P): Fibre Channel Point-to-Point • FC(L): Fibre Channel-Arbitrated Loop • FC(-): Not applicable • SAS: Serial Attached SCSI • iSCSI: iSCSI Channel
Topo (C)	FC only. Configured topology
Speed (A)	Actual host channel link speed: • FC: 2G, 4G, Auto • SAS: 1.5G, 3G, 6G • iSCSI: 10Mb, 100Mb, 1G • Blank if not applicable
Speed (C)	Configured host channel link speed: • FC: 2G, 4G, Auto • SAS: 1.5G, 3G, 6G • iSCSI: 10Mb, 100Mb, 1G • Blank if not applicable
Status	Whether the channel is operational
PID	FC only. Primary ID, or blank if not applicable
SID	FC only. Secondary ID, or blank if not applicable

Field	Description
IP Ver	iSCSI only.IPv4
IP	iSCSI only. Primary IP address
IP(S) Address	iSCSI only. Secondary IP address. This field is only displayed during failover.
MAC	Unique Media Access Control (MAC) hardware address, also called the physical address.

Example

Show channel information for a dual-controller FC system.

```
# show channels

Ctlr Ch Type  Media    Topo(C) Speed(A) Speed(C) Status PID SID
-------------------------------------------------------------
A    0  Host  FC(P)    PTP     2G       2G       Down
A    1  Host  FC(P)    PTP     2G       2G       Down
A    2  Host  FC(P)    PTP     2G       2G       Down
A    3  Host  FC(P)    PTP     2G       2G       Down
A    0  Drive SAS                             Up
A    1  Drive SAS                             Up
B    0  Host  FC(P)    PTP     2G       2G       Down
B    1  Host  FC(P)    PTP     2G       2G       Down
B    2  Host  FC(P)    PTP     2G       2G       Down
B    3  Host  FC(P)    PTP     2G       2G       Down
B    0  Drive SAS                             Up
B    1  Drive SAS                             Up
-------------------------------------------------------------
```

.Show channel information for a dual-controller iSCSI system.

```
# show channels
Ctlr Ch Type  Media    Speed    Status IP-Ver
  IP                                          MAC
-------------------------------------------------------------
A    0  Host  iSCSI    1G       Up     IPv4
  10.11.10.4                                  00:90:F1:FF:0B:1E
A    1  Host  iSCSI    1G       Up     IPv4
  10.10.10.5                                  00:90:F1:FF:0B:1A
A    0  Drive SAS              Up     N/A
  N/A                                         N/A
B    0  Host  iSCSI    1G       Up     IPv4
  10.11.10.2                                  00:90:F1:FF:4D:0E
B    1  Host  iSCSI    1G       Up     IPv4
  10.10.10.3                                  00:90:F1:FF:4D:0A
B    0  Drive SAS              Up     N/A
  N/A                                         N/A
-------------------------------------------------------------
```

Related Commands

- "set host-parameters" on page 116

show cli-parameters

Description

Shows the current CLI settings. Alias: `show output-format`.

Input

```
show cli-parameters
```

Output

Field	Description
`Timeout`	Timeout value, in seconds, for the login session.
`Output Format`	• `console`: Output is shown in human-readable console format. • `api`: Output is shown in XML format.
`Base`	• `2`: Capacities are shown in base 2, which uses 1024 as a divisor. • `10`: Capacities are shown in base 10, which uses 1000 as a divisor.
`Pager`	• `enabled`: Pager feature is enabled, which halts output after each full screen until keyboard input is detected. • `disable`: Pager feature is disabled.

Example

Show current CLI settings.

```
# show cli-parameters

Timeout: 1800
Output Format: console
Base: 10
Pager: enabled
```

Related Commands

- "set cli-parameters" on page 106

show configuration

Description

Shows system configuration information.

Input

```
show configuration
```

Output

Category and Description	Fields
SYSTEM Shows system information	See "show system" on page 200.
CONTROLLER Shows information about each controller	• Serial Number • Hardware Version • CPLD Version • MAC Address • Node WWN • IP Address • IP Subnet Mask • IP Gateway • Disk Drives Present • Virtual Disks • Cache Memory Size(MB) • Host Channels • Host Bus Type • Drive Channels • Drive Bus Type
VERSIONS Shows information about software and hardware versions for each controller	See "versions" on page 225.
CHANNELS Shows information about each channel	See "show channels" on page 143.
DRIVES Shows information about each disk drive	See "show disks" on page 154.

Category and Description	Fields
ENCLOSURE LIST Shows information about the disk drives in each enclosure	See encl option output for "show disks" on page 154.
VIRTUAL DISKS Shows information about each virtual disk	See "show vdisks" on page 208.
SES and ENCLOSURE COMPONENTS Show general SCSI Enclosure Services (SES) information followed by component-specific information	See "show enclosure-status" on page 158.
MIDPLANE FRU and ENCLOSURE FRU Show information for all field-replaceable units (FRUs) in the controller enclosure and in any attached expansion enclosures	See "show frus" on page 167.

show controller-date

Description

Shows the system's current date and time.

Input

```
show controller-date
```

Output

Field	Description
`Controller Date`	Day, date, time, and year

Example

Show the system date and time.

```
# show controller-date

Controller Date: THU SEP 21 15:56:45 2006
```

Related Commands

- "set controller-date" on page 108

show debug-log

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Shows the debug logs for the RAID storage controller (SC), the management controller (MC), the semaphore trace, task logs, or all of them. If no logs are specified, all logs are shown.

Input

```
show debug-log mc [entries]
show debug-log sc controller a|b|both region boot|error|debug|all
show debug-log sem-trace-log [entries]
show debug-log task-log [entries]
show debug-log all
```

Parameter	Description
`mc`	Specifies to show MC debug logs.
`entries`	Optional. Specifies the number of MC logs to show, which is recommended because debug logs can be very large.
`sc`	Specifies to show SC logs only.
`controller a|b|both`	Specifies whether to show SC debug logs for controller A (upper), controller B (lower), or both.
`region boot|error|debug|all`	Specifies the type of SC logs to show: • `boot`: Shows only the boot log. • `error`: Shows only error logs recorded for previous SC crash or improper shutdown. • `debug`: Shows only debug logs. • `all`: Shows all logs.
`sem-trace-log`	Specifies the semaphore trace log.

Parameter	Description
entries	Optional. Specifies the number of semaphore log entries to show. If this parameter is omitted, all semaphore log entries are shown.
task-log	Specifies to show the MC task status at the time this command is issued.
entries	Optional. Specifies the number of task log entries to show. If this parameter is omitted, all semaphore log entries are shown.
all	Specifies to show all SC and MC logs.

Output

Shows debug log entries.

Example

Show all SC and MC logs.

```
# show debug-log all
```

Show 10 MC debug log entries.

```
# show debug-log mc 10
```

Show SC error logs for both controllers.

```
# show debug-log sc controller both region error
```

Show all task-log entries.

```
# show debug-log task-log
```

Related Commands

- "set debug-log-parameters" on page 109
- "show debug-log-parameters" on page 152

show debug-log-parameters

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Shows which debug message types are enabled (on) or disabled (off) for inclusion in the storage controller debug log.

Input

```
show debug-log-parameters
```

Output

Field	Description
`host`	Host interface debug messages
`disk`	Disk interface debug messages
`mem`	Internal memory debug messages
`fo`	Failover/recovery debug messages
`msg`	Inter-controller message debug messages
`fca, fcb, fcc, fcd`	Four levels of Fibre Channel driver debug messages
`misc`	Internal debug messages
`rcm`	Removable-component manager debug messages
`raid`	RAID debug messages
`cache`	Cache debug messages
`emp`	Enclosure Management Processor debug messages
`capi`	Internal Configuration API debug messages
`mui`	Internal service interface debug messages
`bkcfg`	Internal configuration debug messages
`awt`	Auto-write-through feature debug messages
`res2`	Internal debug messages

Field	Description *(Continued)*
capi2	Internal Configuration API tracing debug messages
dms	Snapshot feature debug messages

Example

Show debug log parameters.

```
# show debug-log-parameters

host: off
disk: on
mem: off
...
```

Related Commands

- "set debug-log-parameters" on page 109
- "show debug-log" on page 150

show disks

Description

Shows information about disk drives in the controller enclosure and any attached expansion enclosures. The `encl` option shows different fields than the other options and shows each slot, whether or not a disk is present. If no parameter is specified, the command shows information for all disk drives.

Input

```
show disks [disks|free|all|encl|vdisk vdisk]
```

Parameter	Description
disks|free|all|encl |vdisk *vdisk*	Optional. Specifies the disk drives to report information about: • *disks*: Specifies the disk drives. For the syntax to use, see "Disk Drive Syntax" on page 24. • free: Shows information for all disk drives that are not assigned (status AVAIL). • all: Shows information for all disk drives. • encl: Show information for all disk drives by enclosure. • vdisk *vdisk*: Shows information for disk drives in a specified virtual disk. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Output (Standard)

Field	Description
ID	SCSI ID
Serial#	Disk drive serial number
Vendor	Disk drive vendor
Rev.	Firmware revision number

Field	Description
State	• AVAIL: Disk is available for use in a vdisk • GLOBAL SP: Global spare • LEFTOVR: Disk contains metadata but is not part of a vdisk • VDISK: Disk is part of a vdisk • VDISK SP: Disk is a spare assigned to a vdisk Any jobs running on the disk or its vdisk follow the state value: • DRSC: The disk is being scrubbed • EXPD: The vdisk is being expanded • INIT: The vdisk is being initialized • LOWF: A low-level format is in progress • RCON: The vdisk is being reconstructed • VRFY: The vdisk is being verified • VRSC: The vdisk is being scrubbed
Type	Disk drive type (SAS or SATA)
Size(GB)	Disk drive capacity in Gbyte
Rate(Gb/s)	Data transfer speed in Gbyte/second
SP	Shows whether the disk drive is connected to single port, and which controller owns the port

Output (encl Option)

Field	Description
Status	Disk drive status: operational or missing
Encl	Enclosure number where the disk drive is located
Slot	Slot number in the enclosure where the disk drive is located
Vendor	Disk drive vendor
Model	Disk drive model
Serial#	Disk drive serial number
Size(GB)	Disk drive size in Gbyte

Example

Show information for disk drives having ID 0.0-2 and 4, where drive 1 is not inserted.

```
# show disks 0.0-2,4

ID  Serial#         Vendor Rev.  State      Type Size(GB) Rate(Gb/s) SP
-------------------------------------------------------------------------
0  KRVN03ZAG4ZPPD  ATA    AD1A  AVAIL      SATA 500      3.0
2  KRVN03ZAGA88PD  ATA    AD1A  AVAIL      SATA 500      3.0
4  KRVN03ZAG4L22D  ATA    AD1A  VDISK VRSC SATA 500      3.0
-------------------------------------------------------------------------
```

Show output for disk drives in a system having one enclosure, where drive 1 is not inserted.

```
# show disks encl

Status Encl Slot Vendor  Model            Serial#         Size(GB)
-----------------------------------------------------------------
Up      0    0    ATA     HDS725050KLA360  KRVN03ZAG4ZPPD  500
Missing 0    1                                             0
Up      0    2    ATA     HDS725050KLA360  KRVN03ZAGA88PD  500
Up      0    3    ATA     HDS725050KLA360  KRVN03ZAGA8A0D  500
Up      0    4    ATA     HDS725050KLA360  KRVN03ZAG4L22D  500
...
-----------------------------------------------------------------
```

Related Commands

- “show vdisks” on page 208

show drive-parameters

Description

Shows the SMART setting.

Input

```
show drive-parameters
```

Output

Field	Description
SMART	Shows whether SMART is enabled or disabled

Example

Show drive parameter settings.

```
# show drive-parameters
Drive Parameters
--------------
SMART: Enabled
```

Related Commands

- "set drive-parameters" on page 112

show enclosure-status

Description

Shows the status of system enclosures and their components. For each attached enclosure, the command shows general SCSI Enclosure Services (SES) information followed by component-specific information.

Input

```
show enclosure-status
```

Output

General SES fields:

Field	Description
`Chassis`	Chassis serial number
`Vendor`	Name of enclosure vendor
`Product ID`	Product model identifier
`Rev`	Product revision number
`CPLD`	Complex Programmable Logic Device revision number
`WWPN`	World wide port name of the SES device reporting the enclosure status
`Status`	Overall status of the enclosure

Enclosure Component Status fields:

Field	Description
`Type`	The component type: • `Fan`: Cooling fan unit • `PSU`: Power supply unit • `Temp`: Temperature sensor • `Voltage`: Voltage sensor • `DiskSlot`: Disk drive module
`#`	Unit ID For the location of each fan and power supply, see Figure 3-1. For the location of each disk slot, see Figure 3-2.
`Status`	Component status: • `Absent`: Component is not present • `Fault`: One or more subcomponents has a fault • `OK`: All subcomponents are operating normally. Temperature status OK indicates that the sensor is working properly, not that the temperature is within an acceptable range. • `N/A`: Status is not available
`FRU P/N`	Part number of the field-replaceable unit (FRU) that contains the component
`FRU S/N`	Serial number of the FRU that contains the component
`Add'l Data`	Additional data such as temperature (Celsius), voltage, or slot address

Figure 3-1 shows the location of each fan and power supply, identified by unit ID. Here, left and right are with respect to the back of the enclosure.

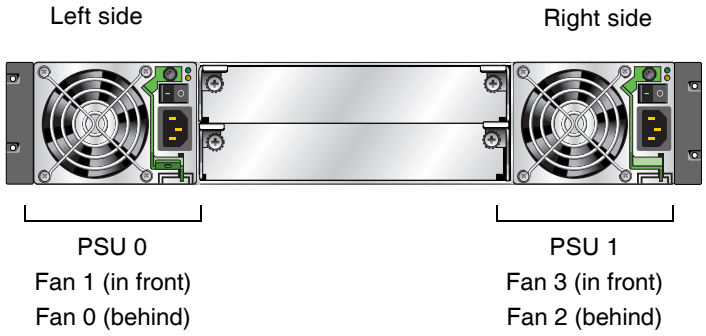

Figure 3-1 Location of Fan and Power Supply Units (PSUs)

Figure 3-2 shows the numbering of disk slots by row from left to right. Here left and right are with respect to the front of the enclosure.

Figure 3-2 Drive Module (Disk Slot) Numbering

Example

Show output for an enclosure.

```
# show enclosure-status
Chassis                          Vendor   Product ID       Rev  CPLD WWPN
  Status
-------------------------------------------------------------------------------
FSCIYHAK000551-242866          DotHill R/Evo 5730-2R 2030 25   500C0FF02428663C
  OK
-------------------------------------------------------------------------------

Type     #  Status   FRU P/N      FRU S/N              Add'l Data
-------------------------------------------------------------------------------
FAN      00 Absent   Unknown      Unknown
FAN      01 OK       FRUHE01      FSCITOR-0607NR012C   --
PSU      00 Absent   Unknown      Unknown
PSU      01 OK       FRUHE01      FSCITOR-0607NR012C   --
Temp     00 OK       FRUHC02      FSCIMIL-06170A4888   temp=40
Temp     01 OK       FRUHC02      FSCIMIL-06170A4893   temp=38
Temp     02 Absent   Unknown      Unknown
Temp     03 OK       FRUHE01      FSCITOR-0607NR012C   temp=40
Voltage  00 Absent   Unknown      Unknown
Voltage  01 Absent   Unknown      Unknown
Voltage  02 Absent   Unknown      Unknown
Voltage  03 OK       FRUHE01      FSCITOR-0607NR012C   voltage=12.370V
Voltage  04 OK       FRUHE01      FSCITOR-0607NR012C   voltage=5.500V
Voltage  05 OK       FRUHE01      FSCITOR-0607NR012C   voltage=2.730V
DiskSlot 00 OK       FRUHA01      FSCIYHAK000551-2428> addr=0
DiskSlot 01 OK       FRUHA01      FSCIYHAK000551-2428> addr=1
DiskSlot 02 OK       FRUHA01      FSCIYHAK000551-2428> addr=2
DiskSlot 03 OK       FRUHA01      FSCIYHAK000551-2428> addr=3
DiskSlot 04 OK       FRUHA01      FSCIYHAK000551-2428> addr=4
DiskSlot 05 OK       FRUHA01      FSCIYHAK000551-2428> addr=5
DiskSlot 06 OK       FRUHA01      FSCIYHAK000551-2428> addr=6
DiskSlot 07 OK       FRUHA01      FSCIYHAK000551-2428> addr=7
DiskSlot 08 OK       FRUHA01      FSCIYHAK000551-2428> addr=8
DiskSlot 09 OK       FRUHA01      FSCIYHAK000551-2428> addr=9
DiskSlot 10 OK       FRUHA01      FSCIYHAK000551-2428> addr=10
DiskSlot 11 Absent   FRUHA01      FSCIYHAK000551-2428> addr=11
-------------------------------------------------------------------------------
```

show events

Description

Shows events for an enclosure, including events from each management controller and each storage controller. A separate set of event numbers is maintained for each controller module. Each event number is prefixed with a letter identifying the controller module that logged the event.

Events are listed from newest to oldest, based on a timestamp with one-second granularity; therefore the event log sequence matches the actual event sequence within about one second.

If SNMP is configured, events can be sent to SNMP traps.

Input

To show a certain number of events:

```
show events [last #] [a|b|both|error]
```

To show events by date:

```
show events [from date] [to date] [a|b|both|error]
```

To show events by ID:

```
show events [from-event ID] [to-event ID] [a|b|both|error]
```

Parameter	Description
`last #`	Optional. Shows the latest specified number of events. If this parameter is omitted, all events are shown.
`from` *`date`*	Optional. Shows events including and after the specified date and time. Use the format *`MMDDYYhhmmss`*, where *`hh`* is the hour on a 24-hour clock. For example, `043006235900` represents Apr 30 2006 at 11:59:00 p.m.

Parameter	Description
`to` *`date`*	Optional. Shows events before and including the specified date and time. Use the format *`MMDDYYhhmmss`*, where *`hh`* is the hour on a 24-hour clock. For example, `043006235900` represents Apr 30 2006 at 11:59:00 p.m. This parameter can be used with either "from" parameter.
`from-event` *`ID`*	Optional. Shows events including and after the specified event ID. If this number is smaller than the ID of the oldest event, events are shown from the oldest available event. This parameter can be used with either "to" parameter.
`to-event` *`ID`*	Optional. Shows events before and including the specified event ID. If this number is larger than the ID of the oldest event, events are shown up to the latest event. This parameter can be used with either "from" parameter.
`a|b|error`	Optional. Specifies to filter the event listing: • `a`: Shows events from controller A (upper) only • `b`: Shows events from controller B (lower) only • `error`: Shows all warning and critical events but not informational events

Output

Shows the following information for each event:

Field	Description
Timestamp	Day, date, time, and year when the event was logged
Event code	Identifies the type of event and might help service technicians diagnose problems; for example, `[181]`
Event ID	Event number prefixed by *A* or *B* to indicate which controller module logged the event; for example, `#A123`
Controller ID	Model, serial number, and ID of the controller module that logged the event; for example, `RPC-69501-001 Array SN#00C0FFD26034 Controller A`

Field	Description
Severity	• CRITICAL: Events that might affect data integrity or system stability. • WARNING: Events that do not affect data integrity. • INFORMATIONAL: Events that show the change of state or configuration changes.
Message	Event-specific message giving details about the event; for example, LAN configuration parameters have been set

Example

Show the last two events.

```
# show events last 2
```

Show the last three warning and critical events.

```
# show events last 3 error
```

Show all events from 11:59:00 p.m. on Apr 30, 2006 through 11:59:00 a.m. on May 2, 2006.

```
# show events from 043006235900 to 050206115900
```

Show a range of events logged by controller A.

```
# show events from-event a100 to-event a123
```

Example

- "set snmp-parameters" on page 133
- "show snmp-parameters" on page 198

show expander-status

Description

Note – This command should only be used by service technicians, or with the advice of a service technician.

Shows diagnostic information relating to SAS expander controller physical channels, known as PHY lanes. Information is shown by controller for each enclosure.

Input

```
show expander-status
```

Output

Parameter	Description
`Id`	Identifier for a specific PHY lane.
`Encl`	Enclosure that contains the SAS expander
`Status`	• `OK`: No errors detected on the PHY lane • `ERROR`: An error has occurred on the PHY lane • DISABLED: The PHY has been disabled
`Type`	• `DRIVE`: Disk drive PHY lane • `INTER-EXP`: Inter-expander PHY lane, communicating between the SAS expanders in a dual-controller system • `INGRESS`: SAS ports on controller enclosures and expansion enclosures • `EGRESS`: SAS ports on expansion enclosures

Example

Show the expander status for all enclosures.

```
# show expander-status
================================================================
                                SAS Status Controller A
================================================================
Id Encl Status     Type
--------------------------------
0  0   OK         DRIVE
1  0   OK         DRIVE
2  0   OK         DRIVE
3  0   OK         DRIVE
4  0   OK         DRIVE
5  0   OK         DRIVE
6  0   OK         DRIVE
7  0   OK         DRIVE
8  0   OK         DRIVE
9  0   OK         DRIVE
10 0   OK         DRIVE
11 0   OK         DRIVE
12 0   OK         INTER-EXP
13 0   OK         INTER-EXP
14 0   OK         INTER-EXP
15 0   OK         INTER-EXP
16 0   OK         INGRESS
17 0   OK         INGRESS
18 0   OK         INGRESS
19 0   OK         INGRESS
20 0   DISABLED   EGRESS
21 0   DISABLED   EGRESS
22 0   DISABLED   EGRESS
23 0   DISABLED   EGRESS
--------------------------------
```

Related Commands

- “clear expander-status” on page 45
- “set expander-fault-isolation” on page 113
- “set expander-phy” on page 114

show frus

Description

Shows information for all field-replaceable units (FRUs) in the controller enclosure and in any attached expansion enclosures. Some information reported is for use by service technicians.

Input

```
show frus
```

Output

Field	Description
Name	FRU name: • CHASSIS_MIDPLANE: 2U chassis and midplane; the metal enclosure and the circuit board to which power and cooling, controller, expansion, and drive modules connect • RAID_IOM: Controller module • BOD_IOM: Expansion module • POWER_SUPPLY: Power and cooling module
Description	FRU description
Part Number	FRU part number
Mid-Plane SN	For the CHASSIS_MIDPLANE FRU, the mid-plane serial number
Serial Number	For the RAID_IOM, BOD_IOM, and POWER_SUPPLY FRUs, the FRU serial number
Revision	FRU revision number
Dash Level	FRU template revision number
FRU Shortname	FRU part number
Mfg Date	Date and time that the FRU was programmed
Mfg Location	Location where the FRU was programmed
Mfg Vendor ID	JEDEC ID of the manufacturer

Field	Description
FRU Location	Location of the FRU in the enclosure, as viewed from the back: • MID-PLANE SLOT: Chassis midplane • UPPER IOM SLOT: Upper controller module or expansion module • LOWER IOM SLOT: Lower controller module or expansion module • LEFT PSU SLOT: Left power and cooling module • RIGHT PSU SLOT: Right power and cooling module
Configuration SN	A customer-specific configuration serial number
FRU Status	Component status: • Absent: Component is not present • Fault: One or more subcomponents has a fault • OK: All subcomponents are operating normally • N/A: Status is not available

Example

This example shows the output for a single FRU.

```
# show frus

Name: POWER_SUPPLY
Description: AC Power Supply
Part Number: FRUKE01
Serial Number: DHSIMIL-0610520155
Revision: 01
Dash Level: 01
FRU Shortname: FRUKE01
Mfg Date: WED JUN 14 21:02:13 2006
Mfg Location: Milpitas California, USA
Mfg Vendor ID: 0x0301
FRU Location: LEFT PSU SLOT
Configuration SN: DHSIMIL-0610138139
FRU Status: OK
```

show host-maps

Description

Shows mapping information for volumes that are mapped to a specified host or to all hosts.

Input

```
show host-maps [host]
```

Parameter	Description
host	Optional. If this parameter is omitted, mapped volumes for all hosts are shown. For FC - Specifies the HBA's nickname or 16-hex-digit WWPN. For iSCSI - Specifies the IP address in IPv4 or IPv6 format or a nickname. For the nickname syntax to use, see "Host Nickname Syntax" on page 26.

Output

Field	Description
WWN	Host WWPN
IP	Host IP address
Name	Host port nickname
Volume Name	Name of the volume seen by the host
Volume SN	Serial number of the volume seen by the host
ID	Host channel ID
LUN	LUN used to access the volume
Access	Host access rights: • rw: read-write • ro: read-only • none: no access
Channels	Channels on which this volume is visible

Example

This example shows volumes' accessibility by FC host HBA1.

```
# show host-maps HBA1

Host [WWN 100000A0B8040BAC, Name (HBA1)] Mapping View:
Volume Name  Volume SN                          ID LUN Access Channels
-------------------------------------------------------------------------
V1           00c0ff0a43180048517e054501000000 0  15  rw     0,1
```

This example shows volumes' accessibility by all iSCSI hosts.

```
# show host-maps
Host [IP 255.255.255.255, Name ()] Mapping View:
Volume Name      Volume SN                          ID LUN Access Channels
-------------------------------------------------------------------------
R0_3_V1          00c0ff0a591300483a21f54501000000 0  23  rw     0,1
R0_1_V1          00c0ff0a427d0048466dc04501000000 0  5   rw     0,1
R0_2_V1          00c0ff0a427d00484f28f54501000000 0  2   ro     0,1
R0_2_V2          00c0ff0a427d00489528f54501000000 0  6   ro     0,1

Host [IP 10.11.10.101, Name (AndiamoP4)] Mapping View:
Volume Name      Volume SN                          ID LUN Access Channels
-------------------------------------------------------------------------
R0_3_V1          00c0ff0a591300483a21f54501000000 0  23  rw     0,1
R0_2_V1          00c0ff0a427d00484f28f54501000000 0  25  rw     0,1
R0_1_V1          00c0ff0a427d0048466dc04501000000 na na  none   none
R0_2_V2          00c0ff0a427d00489528f54501000000 0  6   ro     0,1

Host [IP 10.10.10.102, Name (AndiamoP5)] Mapping View:
Volume Name      Volume SN                          ID LUN Access Channels
-------------------------------------------------------------------------
R0_3_V1          00c0ff0a591300483a21f54501000000 0  23  rw     0,1
R0_2_V1          00c0ff0a427d00484f28f54501000000 0  26  rw     0,1
R0_1_V1          00c0ff0a427d0048466dc04501000000 na na  none   none
R0_2_V2          00c0ff0a427d00489528f54501000000 0  6   ro     0,1
```

Related Commands

- "show host-wwn-names" on page 175
- "show volume-maps" on page 213

show host-parameters

Description

Shows information about host channels on both controllers.

Input

```
show host-parameters
```

Output

Field	Description
Ctlr	Controller ID
Ch	Host channel number
Type	• Host: Channel to host
Media	• FC(P): Fibre Channel Point-to-Point • FC(L): Fibre Channel-Arbitrated Loop • SAS: Serial Attached SCSI • iSCSI: iSCSI Channel
Topo (C)	FC: Configured topology
Speed (A)	Actual host channel link speed: • FC: 2G,4G, Auto • SAS: 1.5G, 3G, 6G, Auto • iSCSI: 10Mb, 100Mb, 1G • Blank if not applicable
Speed (C)	Configured host channel link speed: • FC: 2G, 4G, 10G, Auto • SAS: 1.5G, 3G, 6G, Auto • iSCSI: 10Mb, 100Mb, 1G • Blank if not applicable
Status	Whether the channel is operational
PID	Primary ID, or blank if not applicable
SID	Secondary ID, or blank if not applicable
IP-Ver	iSCSI: IPv4
IP	iSCSI: Primary IP address

Field	Description
Netmask	iSCSI: Netmask IP address
Gateway	iSCSI: Gateway IP address
Service Port	Service Port

Example

Show host parameters for a dual-controller FC system.

```
# show host-parameters

Ctlr Ch Type  Media    Topo(C) Speed(A) Speed(C) Status PID SID
--------------------------------------------------------------
A    0  Host  FC(L)    Loop    2G       2G       Up     0
A    1  Host  FC(L)    Loop    2G       2G       Down
B    0  Host  FC(L)    Loop    2G       2G       Down
B    1  Host  FC(L)    Loop    2G       2G       Up     1
--------------------------------------------------------------
```

Show host parameters for a single controller iSCSI system.

```
# show host-parameters
Host Parameters
---------------
Ctlr: A
Ch: 0
Type: Host
Media: iSCSI
Status: Up
IP-Ver: IPv4
IP: 10.11.10.4
Netmask: 255.255.255.0
Gateway: 0.0.0.0
Service-Port: 3260

Ctlr: A
Ch: 1
Type: Host
Media: iSCSI
Status: Up
IP-Ver: IPv4
IP: 10.10.10.5
Netmask: 255.255.255.0
Gateway: 0.0.0.0
Service-Port: 3260
```

Related Commands

- “set host-parameters” on page 116

show host-port-interconnects

Description

Shows the status of the internal connection between FC host port pairs.

Input

```
show host-port-interconnects
```

Example

Show the status of the host port interconnect.

```
# show host-port-interconnects

Host Port Interconnect: enabled
```

Related Commands

- "set host-port-interconnects" on page 119

show host-wwn-names

Description

Shows the current list of host world wide names (WWNs). FC host channels only.

Input

```
show host-wwn-names
```

Output

Field	Description
Host-ID/WWN	Host port WWN
Name	Nickname assigned to the host WWN

Example

Show host WWNs and nicknames assigned to them.

```
# show host-wwn-names

Host-ID/WWN      Name
---------------------
100000A0B8040BAC HBA1
100000A0B8040BAD HBA2
---------------------
```

Related Commands

- “create host-wwn-name” on page 48
- “delete host-wwn-name” on page 69
- “set host-wwn-name” on page 120

show iscsi-hosts

Description

Shows the iSCSI host initiators.

Input

```
show iscsi-hosts
```

Output

Field	Description
Host-IP	iSCSI host initiator's IP address in IPv4 format
Name	iSCSI host initiator's nickname, if any

Example

Show iSCSI host initiators.

```
# show iSCSI

iSCSI-Host-IP                                    Name
-----------------------------------------------------------------
255.255.255.255
0000:0000:0000:0000:0000:0000:0A0B:0A65       AndiamoP4
0000:0000:0000:0000:0000:0000:0A0A:0A66       AndiamoP5
-----------------------------------------------------------------
```

Related Commands

- "create iscsi-host" on page 49
- "delete iscsi-host" on page 70
- "set iscsi-host" on page 121

show job-parameters

Description

Shows parameters for background scrub, partner firmware upgrade, and other jobs.

Input

```
show job-parameters
```

Output

Field	Description
`Background Scrub`	Shows whether disks are automatically checked for disk drive defects to ensure system health.
`Partner Firmware Upgrade`	Shows whether component firmware versions are monitored and will be automatically upgraded on the partner controller.
`Utility Priority`	Priority at which jobs (such as vdisk verification and reconstruction but not background scrub) run with respect to I/O operations competing for the system's processors: `High`, `Medium`, or `Low`.

Example

Show a system's job parameters.

```
# show job parameters

Job Parameters
--------------
Background Scrub       : Enabled
Parter Firmware Upgrade: Disabled
Utility Priority       : Low
```

Related Commands

- "set job-parameters" on page 122

show license

Description

Shows whether a license key is installed and information about licensed features.

Input

```
show license
```

Output

Field	Description
`License Key`	• The license key, if a license is installed and valid • `not installed`, if a license is invalid or is not installed
`Base Max Snapshots`	Number of snapshots allowed without an installed license
`Max Snapshots`	Number of snapshots allowed by the installed license
`Platform Max Snapshots`	Number of snapshots that the highest-level license allows on this hardware platform
`In-Use Snapshots`	Number of existing snapshots
`Volume Copy`	Shows whether the installed license permits (Enabled) or excludes (Disabled) use of volume copy functions.
`VDS`	Shows whether the Virtual Disk Service (VDS) is enabled or disabled.
`VSS`	Shows whether the Virtual Shadow Copy Service (VSS) is enabled or disabled.

Example

Show information about the installed license.

```
# show license

License Key: 570c8b65899822a37a7acc1fac3c332a
Base Max Snapshots: 16
Max Snapshots: 16
Platform Max Snapshots: 64
In-Use Snapshots: 15
Volume Copy: Enabled
VDS: Disabled
VSS: Disabled
```

show master-volumes

Description

Shows information about master volumes associated with a specified controller or snap pool. If no parameters are specified, information about all master volumes is shown.

Input

```
show master-volumes [controller a|b] [snap-pool volume]
```

Parameter	Description
`controller a|b`	Optional. Only includes master volumes owned by controller A (upper) or controller B (lower).
`snap-pool` *`volume`*	Optional. Only includes master volumes associated with a specified snap pool volume name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Output

Field	Description
`Vdisk`	Virtual disk name
`Serial#`	Master volume serial number
`Name`	Master volume name
`Size`	Total size of the master volume
`Status`	Indicates whether the master volume is valid or offline, or the associated snap pool is offline
`Snap-pool Name`	Name of associated snap pool
`Snapshots`	Number of snapshots that exist for the master volume
`Snap Data`	Amount of snap-pool space occupied by this master volume for its associated snapshots (preserved and write data)
`Rollback`	Either the percent complete if rollback is in progress, or "`---`" if rollback is not in progress

Example

Show information about master volumes associated with snap pool SP1.

```
# show master-volumes snap-pool SP1

Vdisk  Serial#                            Name   Size     Status
  Snap-pool Name  Snapshots  Snap Data  Rollback
-------------------------------------------------------------------
VD1    00c0ff0a43180048acc3134501000000  MV1     10.0GB  Valid
  SP1             1            0.0KB    ---
-------------------------------------------------------------------
```

Related Commands

- “convert master-to-std” on page 46
- “convert std-to-master” on page 47
- “delete all-master-volumes” on page 66
- “delete master-volume” on page 71
- “rollback master-volume” on page 98

show network-parameters

Description

Shows the network settings for each RAID controller.

Input

```
show network-parameters
```

Output

Field	Description
`IP Address`	Controller IP address
`Gateway`	Controller gateway
`Subnet Mask`	Controller subnet mask
`MAC Address`	Controller's unique Media Access Control address
`Addressing Mode`	• `Manual`: Network settings set statically • `DHCP`: DHCP used to set network parameters

Example

Show the network parameters for a dual-controller system.

```
# show network-parameters

Network Parameters Controller A
-------------------------------
IP Address     : 172.22.1.200
Gateway        : 172.22.1.1
Subnet Mask    : 255.255.255.0
MAC Address    : 00:C0:FF:0A:43:18
Addressing Mode: Manual

Network Parameters Controller B
-------------------------------
IP Address     : 172.22.1.201
Gateway        : 172.22.1.1
Subnet Mask    : 255.255.255.0
MAC Address    : 00:C0:FF:0A:43:26
Addressing Mode: Manual
```

Related Commands

- "set network-parameters" on page 124

show port-wwn

Description

Shows the world wide port name (WWPN) for each host channel.

Input

```
show port-wwn
```

Output

Field	Description
CTRL	Controller
CH	Host channel number
WWN	World wide port name (WWPN)

Example

Show port WWNs for both controllers.

```
# show port-wwn

CTRL CH WWPN
-------------------------
A    0  207000C0FF242866
A    1  217000C0FF242866
B    0  207800C0FF242866
B    1  217800C0FF242866
-------------------------
```

show protocols

Description

Shows which service and security protocols are enabled or disabled.

Input

```
show protocols
```

Output

Status of each protocol.

Example

Show the status of service and security protocols.

```
# show protocols

Service and Security Protocols
------------------------------
Web Browser Interface                          (HTTP)    : Enabled
Secure Web Browser Interface                   (HTTPS)   : Enabled
Command Line Interface                         (Telnet)  : Enabled
Secure Command Line Interface                  (SSH)     : Enabled
Storage Management Initiative Specification(SMIS)    : Enabled
File Transfer Protocol                         (FTP)     : Disabled
Simple Network Management Protocol             (SNMP)    : Enabled
Service Interface                              (Service): Disabled
Service Debug                                  (Debug)   : Disabled
Inband SES Management                          (SES)     : Enabled
Inband CAPI Management                         (CAPI)    : Enabled
```

Related Commands

- "set protocols" on page 128

show redundancy-mode

Description

Shows the redundancy status of the system.

Input

```
show redundancy-mode
```

Output

Field	Description
`Redundancy Mode`	Possible values are: Unknown, Standalone Single Port, Standalone Dual Port, Active-Active Single Port, Active-Active, Active Pass Dual Port, A-A Dual Port Multi ID, and Independent Cache Performance Mode.
`Redundancy Status`	• `Redundant Operation`: Both controllers are operating • `Only Operational`: Only the connected controller is operating
`Controller` *`ID`* `Status`	• `Operational`: The controller is operational • `Not Installed`: The controller is not installed or has failed
`Controller` *`ID`* `Serial Number`	• Controller module serial number • `Not Available`

Example

Show the system redundancy status.

```
# show redundancy-mode

System Redundancy
-----------------
Redundancy Mode            : Active-Active
Redundancy Status          : Redundant Operation
Controller A Status        : Operational
Controller A Serial Number: 00C0FF0A4318
Controller B Status        : Operational
Controller B Serial Number: 00C0FF0A4326
```

show schedule-details

Description

Shows information about a specified task schedule.

Inputt

```
show schedule-details schedule
```

Parameter	Description
schedule	Specifies the schedule name.

Output

Field	Description
`Schedule Name`	Schedule name
`Schedule Specification`	Parameters of the schedule
`Schedule Status`	`Ready` or `Active`
`Next Time`	The next time the task will run
`Task to Run`	The name of the task to run
`Error Message`	Any error message associated with this schedule
`Task Details`	Details of the task, as shown by `show task-details`

Example

Show details for task schedule Sched1 which should run task T1. The task will run at 12:59. When the task was scheduled to run the previous time, an error occurred.

```
# show schedule-details Sched1

Schedule Details
----------------
Schedule Name: Sched1
Schedule Specification: Start 2/19/2007 23:47:00, Every 3 Minutes
Schedule Status: Ready
Next Time: 2/23/2007 12:59:00
Task To Run: T1
Error Message: Schedule unable to execute Task, - Task is not Ready
to run

   Task Details
   ------------
   Task Name: T1
   Task Type: TakeSnapshot
   Task Status: Ready
   Task State: Init
   Master Volume Name: VD1_V1
   Master Volume Serial: 00c0ffd2710700481a8fcf4501000000
   Snapshot Prefix: T1
   Retention Count: 1
   Last Snapshot Created: T1_S1530
   Error Message: none

      Snapshot Name        Snapshot Serial
      ------------------------------------------------------
      T1_S1530             00c0ffd2710700482ce3de4501000000
```

Related Commands

- “create schedule” on page 52
- “create task” on page 56
- “show schedules” on page 190

show schedules

Description

Shows configured task schedules.

Input

```
show schedules
```

Output

Field	Description
`Schedule Name`	Schedule name
`Task To Run`	Task name
`Next Time`	The next time the task will run; or `none` if the task will not run again

Example

Show configured task schedules for a system.

```
# show schedules
Schedule Name            Task To Run        Next Time
-------------------------------------------------------------------
S1                       T1                 2/23/2007 12:59:00
S10                      T10                2/23/2007 12:58:00
S2                       T2                 2/23/2007 12:56:00
S3                       T3                 2/23/2007 12:57:00
S4                       T4                 2/23/2007 12:59:00
S7                       T7                 2/23/2007 12:55:00
-------------------------------------------------------------------
```

Related Commands

- “create schedule” on page 52
- “create task” on page 56
- “show schedule-details” on page 188

show sensor-status

Description

Shows the status and current values reported by environmental sensors in each of the controllers and power supplies.

Input

```
show sensor-status
```

Output

Field	Description
`Sensor Name`	Where the sensor is located
`Value`	The value of the sensor
`Status`	Component status: • `Absent`: Component is not present • `Fault`: One or more subcomponents has a fault • `OK`: All subcomponents are operating normally. Temperature status OK indicates that the sensor is working properly, not that the temperature is within an acceptable range. • `N/A`: Status is not available

Example

Show the status and value of the sensors.

```
# show sensor-status
Sensor Name                          Value    Status
-----------------------------------------------------
CPU Temperature-Ctlr A               55       OK
CPU Temperature-Ctlr B               55       OK
FPGA Temperature-Ctlr A              59       OK
FPGA Temperature-Ctlr B              60       OK
Onboard Temperature 1-Ctlr A         38       OK
Onboard Temperature 1-Ctlr B         36       OK
Onboard Temperature 2-Ctlr A         46       OK
Onboard Temperature 2-Ctlr B         45       OK
Capacitor Temperature-Ctlr A         37       OK
Capacitor Temperature-Ctlr B         35       OK
CM Temperature, Upper-Ctlr A         40       OK
CM Temperature, Lower-Ctlr B         38       OK
Power Supply 1 Temperature           38       OK
Power Supply 2 Temperature           40       OK
Capacitor Pack Voltage-Ctlr A         8.29    OK
Capacitor Pack Voltage-Ctlr B         8.25    OK
Capacitor Cell 1 Voltage-Ctlr A       2.05    OK
Capacitor Cell 1 Voltage-Ctlr B       2.05    OK
Capacitor Cell 2 Voltage-Ctlr A       2.09    OK
Capacitor Cell 2 Voltage-Ctlr B       2.05    OK
Capacitor Cell 3 Voltage-Ctlr A       2.03    OK
Capacitor Cell 3 Voltage-Ctlr B       2.06    OK
Capacitor Cell 4 Voltage-Ctlr A       2.12    OK
Capacitor Cell 4 Voltage-Ctlr B       2.09    OK
Capacitor Charge-Ctlr A              100%     OK
Capacitor Charge-Ctlr B              100%     OK
Power Supply 1 Voltage, 12V          12.43    OK
Power Supply 1 Voltage, 5V            5.47    OK
Power Supply 1 Voltage, 3.3V          3.48    OK
Power Supply 2 Voltage, 12V          12.37    OK
Power Supply 2 Voltage, 5V            5.50    OK
Power Supply 2 Voltage, 3.3V          2.73    OK
Overall Unit Status                   OK      OK
-----------------------------------------------------
```

Related Commands

- “show enclosure-status” on page 158
- “show expander-status” on page 165

show shutdown-status

Description

Shows whether the RAID (storage) controllers are shut down.

Input

```
show shutdown-status
```

Output

Message stating whether each controller is up (operating) or down (shut down).

Example

Show the shutdown status for each controller.

```
# show shutdown-status

storage controller A is up
storage controller B is up
```

Related Commands

- "restart" on page 95
- "shutdown" on page 218

show snap-pools

Description

Shows information about snap pools owned by a specified controller or both controllers.

Input

```
show snap-pools [controller a|b|both]
```

Parameter	Description
`controller a|b|both`	Optional. Includes snap pools owned by controller A (upper) only, controller B (lower) only, or both.

Output

Field	Description
`Vdisk`	Virtual disk name
`Serial#`	Snap pool serial number
`Name`	Snap pool name
`Size`	Total size of the snap pool volume
`Free`	Amount of free space available in this snap pool
`MasterVols`	Number of master volumes associated with this snap pool
`Snapshots`	Number of snapshots using this snap pool
`Threshold`	Snap pool threshold level (Warning, Error, and Critical)
`%Usage`	Threshold value (percent of snap pool space used) that triggers the threshold's policy
`Policy`	Recovery policy invoked when threshold value is reached
`SizeToExpand`	Increment size by which the snap pool is automatically expanded each time the threshold level is reached. This parameter applies when autoexpand policy is selected; otherwise its value is 0.

Example

Show information for snap pools owned by either controller.

```
# show snap-pools

Vdisk           Serial#               Name          Size
  Free   MasterVols  Snapshots
-----------------------------------------------------------

NP   00c0ff0a48930048ea1a834601000000  SnP          30.0GB
   30.0GB   0           0
   Threshold  %Usage   Policy                  SizeToExpand
   ---------------------------------------------------------
   Warning    75%      Notify Only             N/A
   Error      90%      Delete Oldest Snapshot    0.0KB
   Critical   99%      Delete Snapshots        N/A
```

Related Commands

- “create snap-pool” on page 54
- “delete snap-pool” on page 73
- “expand snap-pool” on page 83
- “set snap-pool-policy” on page 129
- “set snap-pool-threshold” on page 131

show snapshots

Description

Shows information about snapshots for a specified controller, master volume, or snap pool. If no parameters are specified, information about all snapshots is shown.

Input

```
show snapshots [controller a|b|both] [master-volume volume]
[snap-pool volume]
```

Parameter	Description
`controller a|b|both`	Optional. Only includes snapshots owned by controller A (upper), controller B (lower) or both.
`master-volume` *`volume`*	Optional. Only includes snapshots associated with the specified master volume name or serial number. For the syntax to use, see "Volume Syntax" on page 25.
`snap-pool` *`volume`*	Optional. Only includes snapshots associated with the specified snap pool name or serial number. For the syntax to use, see "Volume Syntax" on page 25.

Output

Field	Description
`Vdisk`	Virtual disk name
`Serial#`	Snapshot serial number
`Name`	Snapshot name
`Creation Date/Time`	Date and time the snapshot was prepared or committed
`Status`	Indicates whether the snapshot is valid, pending, or invalid; whether the master volume is offline; and whether the associated snap pool is offline
`Master Volume Name`	Name of associated master volume
`Snap-pool Name`	Name of associated snap pool
`Snap Data`	Total amount of preserved and write data associated with the snapshot

Field	Description
UniqueData	Amount of preserved and write data that is unique to the snapshot
SharedData	Amount of preserved and write data that is shared between this snapshot and other snapshots

Example

This example shows information about snapshots associated with snap pool SP2.

```
# show snapshots snap-pool SP2

Vdisk Serial#                            Name  Creation Date/Time        Status
  Master Volume Name  Snap-pool Name    Snap Data  UniqueData  SharedData
------------------------------------------------------------------------------
VD1   00c0ff0a43180048ddc3134501000000  SS1   FRI SEP 22 11:07:09 2006  Valid
  V2                  SP2                 0.0KB      0.0KB       0.0KB
------------------------------------------------------------------------------
```

Related Commands

- “show master-volumes” on page 180
- “show snap-pools” on page 194

show snmp-parameters

Description

Shows current settings for SNMP notification.

Input

```
show snmp-parameters
```

Output

Field	Description
`SNMP`	• `enabled`: SNMP notification is enabled • `disabled`: SNMP notification is disabled
`SNMP Filter`	• `Critical`: Only critical events are sent as traps • `Warning`: All critical events and warnings are sent as traps • `Informational`: All events are sent as traps • `None`: No events are sent as traps and traps are disabled
`SNMP Trap Host IP#`	IP address of each trap host
`SNMP read community`	Community string for read-only access, not shown to monitor-level users
`SNMP write community`	Community string for write access, not shown to monitor-level users

Example

Show SNMP notification settings.

```
# show snmp-parameters

SNMP
----
SNMP: enabled
SNMP Filter: Critical, Warning
SNMP Trap Host IP1: 172.22.4.171
SNMP read community: public
SNMP write community: private
```

Related Commands

- "set snmp-parameters" on page 133

show system

Description

Shows information about the system.

Input

```
show system
```

Output

- System Name
- System Contact
- System Location
- System Information
- Vendor Name
- Product ID
- Product Brand
- SCSI Vendor ID
- Enclosure Count

Related Commands

- “set system” on page 134

show task-details

Description

Shows details of a configured task.

Input

```
show task-details task
```

Output (TakeSnapshot)

Field	Description
Task Name	Task name
Task Type	TakeSnapshot
Task Status	The status of the task: Ready, Active
Task State	The current state of the task: Init, Vol Verified, License Checked, Name Created, Snap Created, Snap Verified
Master Volume Name	Master volume name
Master Volume Serial	Volume serial number
Snapshot prefix	Label identifying the snapshot
Retention Count	Number of snapshots with this prefix that are retained
Last Snapshot Created	The name of the last snapshot taken
Error Message	Any error message associated with this task

Output (ResetSnapshot)

Field	Description
Task Name	Task name
Task Type	ResetSnapshot
Task Status	The status of the task: Ready, Active

Field	Description
Task State	The current state of the task: Init, Snap Verified
Snapshot Name	Name of the snapshot to be reset
Snapshot Serial	Serial number of the snapshot to be reset
Error Message	Any error message associated with this task

Output (VolumeCopy)

Field	Description
Task Name	Task name
Task Type	VolumeCopy
Task Status	The status of the task: Ready, Active
Task State	The current state of the task: Init, Vol Verified, Name Created, Vol Created
Source Volume Name	Name of the volume to be copied
Source Volume Serial	Serial number of the volume to be copied
Destination Vdisk Name	Name of the destination virtual disk
Destination Vdisk Serial	Serial number of the destination virtual disk
Destination Volume Prefix	Prefix of the destination volume
Include Modified Data	True or False
Last Copy Created	Name of the last copy created
Error Message	Any error message associated with this task

Example

Show the details of the volume copy task C1.

```
# show task-details C1
Task Details
------------
Task Name: C1
Task Type: VolumeCopy
Task Status: Ready
Task State: Init
Source Volume Name: VD1_V1
Source Volume Serial: 00c0ffd2710700481a8fcf4501000000
Destination Vdisk Name: VD2
Destination Vdisk Serial: 00c0ffd270280048b68ecf4500000000
Destination Volume Prefix: Copy
Include Modified Data: false
Last Copy Created: none
Error Message: none
```

Show the details of the take snapshot task T1.

```
# show task-details T1
Task Details
------------
Task Name: T1
Task Type: TakeSnapshot
Task Status: Active
Task State: Snap Created
Master Volume Name: VD1_V1
Master Volume Serial: 00c0ffd2710700481a8fcf4501000000
Snapshot Prefix: T1
Retention Count: 1
Last Snapshot Created: T1_S0963
Error Message: none

   Snapshot Name        Snapshot Serial
   ---------------------------------------------------
   T1_S0963             00c0ffd271070048ab53dd4501000000
```

Related Commands

- “create schedule” on page 52
- “create task” on page 56
- “delete task” on page 76
- “show schedule-details” on page 188
- “show tasks” on page 205

show tasks

Description

Shows configured tasks.

Input

```
show tasks
```

Output

Field	Description
`Task Name`	Task name
`Task Type`	Task type: `TakeSnapshot`, `ResetSnapshot`, `VolumeCopy`
`Task Status`	Task status: `Ready`, `Active`

Example

Show configured tasks for the system.

```
# show tasks
Task Name                    Task Type             Task Status
-------------------------------------------------------------
Task1                        TakeSnapshot          Ready
copyVol                      VolumeCopy            Active
Reset1                       ResetSnapshot         Ready
-------------------------------------------------------------
```

Related Commands

- “create task” on page 56
- “create schedule” on page 52
- “delete task” on page 76
- “show schedule-details” on page 188
- “show task-details” on page 201

show users

Description

Shows configured user profiles.

Input

```
show users
```

Output

Field	Description
Username	User name
Access Level	• Monitor: View-only access to selected user interfaces • Manage: Modify access to selected user interfaces
User Type	Applies to the WBI only. • Standard: Has access to standard administrative functions • Advanced: Has access to standard and advanced functions • Diagnostic: Has access to standard, advanced, and troubleshooting functions
WBI	Web-browser interface
CLI	Command-line interface
FTP	File transfer protocol interface

Example

Show configured users for a system.

```
# show users

Username    Access Level   User Type    WBI   CLI   FTP
-----------------------------------------------------------
monitor     Monitor        Standard      x     x
manage      Manage         Diagnostic    x     x     x
ftp         Manage         Standard                  x
jsmith      Manage         Advanced      x           x
-----------------------------------------------------------
```

Related Commands

- “create user” on page 59
- “delete user” on page 77
- “set user” on page 135
- “show users” on page 206

show vdisks

Description

Shows information for all or specific virtual disks.

Input

```
show vdisks [vdisks]
```

Parameter	Description
vdisks	Optional. Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25. If this parameter is omitted, information is shown for all virtual disks.

Output

Field	Description
Name	Virtual disk name
Size	Virtual disk size
Free	Virtual disk size
Own	Controller module that owns the virtual disk
RAID	Virtual disk RAID level
Dsk	Number of disk drives in the virtual disk
Spr	Number of vdisk spares assigned to the virtual disk
Chk	Virtual disk chunk size

Field	Description
Stat	• CRIT: The vdisk is online, however some drives are down and the vdisk is not fault tolerant • FTDN: The vdisk is online and fault tolerant, however some of the drives are down • FTOL: The vdisk is online and fault tolerant • OFFL: The vdisk is offline either because of initialization or because drives are down and data may be lost • QRCR: The vdisk is in a critical state and has been quarantined because some drives are missing • QROF: The vdisk is offline and has been quarantined because some drives are missing • UP: The vdisk is online and does not have fault tolerant attributes
Jobs	• DRSC: Disks within the vdisk are being scrubbed • EXPD: The vdisk is being expanded • INIT: The vdisk is initializing • LOWF: A low-level format is in progress • RCON: The vdisk is being reconstructed • VRFY: The vdisk is being verified • VRSC: The vdisk is being scrubbed
Serial#	Virtual disk serial number

Example

Show information about virtual disk VD1 only.

```
# show vdisks VD1

# show vdisks VD1

Name       Size     Free     Own  RAID   Dsk Spr Chk Stat Jobs
  Serial#
------------------------------------------------------------------
VD1          1.0TB  999.2GB  A    RAID0  2   0   64  UP   VSRC 65%
  00c0ff0a431800489f7c054500000000
------------------------------------------------------------------
```

Related Commands

- “abort create” on page 38
- “create vdisk” on page 61
- “delete vdisk” on page 78
- “expand vdisk” on page 85
- “set vdisk” on page 137
- “show vdisks” on page 208

show volumecopy-status

Description

Shows information about in-progress volume copy operations. While a volume copy is in progress, the destination volume cannot be accessed.

Input

```
show volumecopy-status [controller a|b]
```

Parameter	Description
`controller a|b`	Optional. Shows volume copy operations for volumes owned by controller A (upper) or controller B (lower) only. If this parameter is omitted, all volume copy operations are shown.

Output

Field	Description
`Name`	Destination volume name
`Serial#`	Destination volume serial number
`Vdisk`	Destination virtual disk name
`Source Volume`	Source volume name
`% Complete`	Percent complete of the volume copy

Example

Show information about volume copies in progress for controller A.

```
# show volumecopy-status controller a

  Name                    Serial#                              Vdisk
    Source Volume           % Complete
  ------------------------------------------------------------------
Copy                    00c0ffd2723600482ec8504501000000  VD1
    MV1                     35
------------------------------------------------------------------
```

Related Commands

- “abort volumecopy” on page 41
- “volumecopy” on page 226

show volume-maps

Description

Shows mapping information for a specified volume or for all volumes.

Input

```
show volume-maps [volume]
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see "Volume Syntax" on page 25. If this parameter is omitted, information for all volumes is shown.

Output

Field	Description
SN	Volume serial number
Name	Volume name
CH	Host channels that volume is mapped on
ID	Host channel ID
LUN	Logical unit number
Access	Access mode: read-write (rw) or read-only (ro)
Port-WWN	FC only: • HBA world wide port name (WWPN) • `all other hosts` for the default mapping of the volume
Host-Port-Identifier	iSCSI only: • IP address of the host port • `all other hosts` for the default mapping of the volume
Nickname	HBA name, or blank if not set or for `all other hosts`

Example

Show the mappings for all volumes on an iSCSI system.

```
 # show volume-maps
Info: Retrieving Data...
Volume [SN 00c0ff0a427d0048466dc04501000000, Name (R0_1_V1)] mapping view:
CH       ID LUN Access Host-Port-Identifier                    Nickname
------------------------------------------------------------------------------
0,1       0   5 rw     255.255.255.255

Volume [SN 00c0ff0a427d00484f28f54501000000, Name (R0_2_V1)] mapping view:
CH       ID LUN Access Host-Port-Identifier                    Nickname
------------------------------------------------------------------------------
0,1       0  25 rw     10.11.10.101                           AndiamoP4
0,1       0  26 rw     10.10.10.102                           AndiamoP5
0,1       0   2 ro     all other hosts

Volume [SN 00c0ff0a427d00489528f54501000000, Name (R0_2_V2)] mapping view:
CH       ID LUN Access Host-Port-Identifier                    Nickname
------------------------------------------------------------------------------
0,1       0   6 ro     all other hosts

Volume [SN 00c0ff0a591300483a21f54501000000, Name (R0_3_V1)] mapping view:
CH       ID LUN Access Host-Port-Identifier                    Nickname
------------------------------------------------------------------------------
0,1       0  23 rw     all other hosts
```

Show the mappings for volume V1 on a FC system.

```
# show volume-maps V1

Volume [SN 00c0ff0a43180048517e054501000000, Name (V1)] mapping view:
CH  ID  LUN  Access  Port-WWN          Nickname
-----------------------------------------------
0,1  0   15  rw      100000A0B8040BAC  HBA1
```

Related Commands

- "show volumes" on page 215

show volumes

Description

Shows volume information for all or specified virtual disks.

Input

```
show volumes [vdisk vdisks] [class standard|ptsnap]
[type snap-pool|mastervolume|snapshot|standard]
```

Parameter	Description
`vdisk` *`vdisks`*	Optional. Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
`class standard|ptsnap`	Optional. Specifies the class of volumes to show.
`type snap-pool| mastervolume| snapshot|standard`	Optional. Specifies the type of volumes to show.

Output

Field	Description
`Vdisk`	Name of the virtual disk
`Volume Name`	Name of the volume
`Size`	Volume size
`WR Policy`	Cache policy (write-back or write-through)
`Class`	`Standard`, `PTSNAP` (snapshot-related), or `unknown`
`Volume Serial Number`	Volume serial number
`Cache Opt`	Cache optimization mode (standard or super-sequential)

Field	Description
Type	• standard: Standard volume; followed by an asterisk for an in-progress volume copy. • snap-pool: Snap-pool volume • mastervol: Master volume • snapshot: Snapshot volume • unknown: Unknown

Example

Show volume information for standard volumes only. Show volume information for

```
# show volumes type standard

# show volumes type standard

Vdisk     Volume Name    Size     WR Policy    Class
  Volume Serial Number                Cache Opt    Type
----------------------------------------------------------
VD1       V1             10.0GB  writeback    standard
  00c0ff0a43180048aff0074501000000    standard     standard
----------------------------------------------------------
```

virtual disk VD1 only.

```
# show volumes vdisk VD1

Vdisk     Volume Name    Size     WR Policy    Class
  Volume Serial Number                Cache Opt    Type
----------------------------------------------------------
VD1       V1             10.0GB  writeback    standard
  00c0ff0a43180048aff0074501000000    standard     standard
VD1       SP1            10.0GB  writeback    PTSNAP
  00c0ff0a43180048d9f0074501000000    standard     snap-pool
VD1       V2             10.0GB  writeback    PTSNAP
  00c0ff0a43180048f3f0074501000000    standard     mastervol
VD1       SS1            10.0GB  writeback    PTSNAP
  00c0ff0a4318004821f1074501000000    standard     snapshot
----------------------------------------------------------
```

Related Commands

- “create volume” on page 64
- “delete volume” on page 81
- “expand volume” on page 87
- “set volume” on page 139
- “show vdisks” on page 208
- “show volume-maps” on page 213

shutdown

Description

Cleanly shuts down the RAID controller in either or both controller modules. This ensures that any data in the controller's write-back cache is written to disk. When both RAID controllers are shut down, hosts cannot access the system's data. Perform a shut down before removing a controller module or powering down the system.

Caution – You can continue to use the CLI when either or both RAID controllers are shut down, but information shown might be invalid.

Input

```
shutdown a|b|both
```

Parameter	Description
`a|b|both`	Specifies whether to shut down the RAID controller in controller module A (upper), controller module B (lower), or both.

Output

Messages are displayed when the RAID controllers are shut down.

Example

Shut down RAID controller A.

```
# shutdown a
```

Related Commands

- "restart" on page 95

stty

Description

Sets and shows terminal information.

Input

```
stty info | hardwrap | rows # | columns #
```

Parameter	Description
`info`	Shows current information about the terminal
`hardwrap`	Toggles the hard wrapping of output. Terminals usually wrap at the screen width without truncating output, but turning On hard wrapping ensures this.
`rows #`	Specifies the number of rows that a terminal can display. The terminal usually sets this value; this is an override. The info parameter shows this as screen height.
`columns #`	Specifies the number of columns that a terminal can display. The terminal usually sets this value; this is an override. The info parameter shows this as screen width.

Example

Show information about the terminal.

```
# stty info

Terminal Type: ANSI
Screen width : 140
Screen height: 60
Hard wrap    : Off
```

trust

Description

Enables an offline virtual disk to be brought online for emergency data collection only. It must be enabled before each use.

Caution – This command can cause unstable operation and data loss if used improperly. It is intended for disaster recovery only.

The `trust` command re-synchronizes the time and date stamp and any other metadata on a bad disk drive. This makes the disk drive an active member of the virtual disk again. You might need to do this when:

- One or more disks of a virtual disk start up more slowly or were powered on after the rest of the disks in the virtual disk. This causes the date and time stamps to differ, which the system interprets as a problem with the "late" disks. In this case, the virtual disk functions normally after being trusted.
- A virtual disk is offline because a drive is failing, you have no data backup, and you want to try to recover the data from the virtual disk. In this case, trust may work, but only as long as the failing drive continues to operate.

When the "trusted" virtual disk is back online, back up its data and audit the data to make sure that it is intact. Then delete that virtual disk, create a new virtual disk, and restore data from the backup to the new virtual disk. Using a trusted virtual disk is only a disaster-recovery measure; the virtual disk has no tolerance for any additional failures.

Input

To enable the trust command:

```
trust enable
```

To trust a virtual disk:

```
trust vdisk vdisk
```

Parameter	Description
`enable`	Enables the trust command before use

Parameter	Description
vdisk *vdisk*	Specifies the virtual disks by name or serial number. For the syntax to use, see “Virtual Disk Syntax” on page 25.

Example

Enable the trust command and then trust virtual disk VD1.

```
# trust enable

Trust Virtual-disk Enabled.

# trust vdisk VD1

Are you sure? yes
Virtual-disk VD1 has been trusted.
```

unmap volume

Description

Unmaps a volume from specified host channels.

Input

```
unmap volume volume mapping mapping [host host]
```

Parameter	Description
volume	Specifies the volume by its name or serial number. For the syntax to use, see “Volume Syntax” on page 25.
mapping *mapping*	Specifies the host-to-volume mapping. For the syntax to use, “Volume Mapping Syntax” on page 26.
host *host*	Optional. If this parameter is omitted, mapped volumes for all hosts are shown. For FC - Specifies the HBA's nickname or 16-hex-digit WWPN. For iSCSI - Specifies the IP address in IPv4 format or a nickname. For the nickname syntax to use, see “Host Nickname Syntax” on page 26.

Example

Remove the volume default mapping from LUN 8 on channels 0-3.

```
# unmap volume V1 mapping 0-3.8
```

Remove the volume mapping from LUN 8 on channel 1 associated with host HBA1.

```
# unmap volume V1 mapping 1.8 host HBA1
```

Related Commands

- “map volume” on page 89
- “show host-maps” on page 169
- “show host-wwn-names” on page 175
- “show volumes” on page 215
- “show volume-maps” on page 213

verify vdisk

Description

For specified RAID 3, 5, 6, and 50 virtual disks, this command verifies all parity blocks and corrects any bad parity. For specified RAID 1 and 10 virtual disks, the command compares the primary and secondary drives and if it finds a mismatch it copies the primary to the secondary.

Input

```
verify vdisk vdisks
```

Parameter	Description
vdisks	Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Verify two virtual disks.

```
# verify vdisk VD1,VD2
```

Related Commands

- "abort verify" on page 40
- "scrub vdisk" on page 100
- "show vdisks" on page 208

versions

Description

Shows the hardware and software versions for each controller module. Alias: `show versions`.

Input

```
versions
```

Output

- Storage Controller CPU Type
- Storage Controller Firmware
- Storage Controller Memory
- Storage Controller Loader
- Management Controller Firmware
- Management Controller Loader
- Expander Controller Firmware
- CPLD Revision
- Hardware Revision
- Host Interface Module
- Host Interface Module Model

volumecopy

Description

Copies a snapshot or a master volume to a new standard volume. The command creates the destination volume you specify, which must be in a vdisk owned by the same controller as the source volume.

Input

```
volumecopy source-volume volume1 dest-vdisk vdisk [modified-
snapshot yes|no] volume2
```

Parameter	Description
`source-volume` *volume1*	Specifies the virtual disks by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
`dest-vdisk` *vdisk*	Specifies the destination virtual disk by name or serial number. For the syntax to use, see "Virtual Disk Syntax" on page 25.
`modified-snapshot yes|no`	Optional. Specifies whether to include or exclude modified write data from the snapshot in the copy. This parameter applies only when the source volume is a snapshot; it is ignored if the source volume is a master volume. • `yes`: Include modified snapshot data. • `no`: Exclude modified snapshot data. If this parameter is omitted for a snapshot, modified snapshot data is excluded.
volume2	Specifies a name for the volume to create on the destination vdisk. For the syntax to use, see "Virtual Disk Syntax" on page 25.

Example

Copy master volume MV1 to new volume RO1 on virtual disk VD2.

```
# volumecopy source-volume MV1 dest-vdisk VD2 RO1
```

Related Commands

- abort volumecopy
- “create task” on page 56
- “show vdisks” on page 208
- “show volumecopy-status” on page 211
- “show volumes” on page 215

Glossary

The glossary defines terms and acronyms used in Phoenix storage system documentation. Definitions obtained from the Storage Networking Industry Association (SNIA) Dictionary are indicated with "(SNIA)" at the end. For the complete SNIA Dictionary, go to www.snia.org/education/dictionary.

active-active Synonym for *dual active* components or controllers. A pair of components, such as the controllers in a failure tolerant storage subsystem that share a task or class of tasks when both are functioning normally. When one of the components fails, the other takes on the entire task. Dual active controllers are connected to the same set of storage devices, improving both I/O performance and failure tolerance compared to a single controller. (SNIA)

address A data structure or logical convention used to identify a unique entity, such as a particular process or network device.

AL_PA See *arbitrated loop physical address (AL_PA)*.

ANSI American National Standards Institute.

arbitrated loop physical address (AL_PA) An 8-bit value used to identify a participating device in an Arbitrated Loop. (SNIA)

API Application programming interface.

ARP Address Resolution Protocol.

array See *storage system*.

block A single sector on a disk. The smallest unit of data that is stored (written) to or retrieved (read) from a disk.

broadcast write Technology that provides simultaneous caching of write data to both RAID controllers' cache memory with positive direct memory access acknowledgement (certified DMA).

cache A high speed memory or storage device used to reduce the effective time required to read data from or write data to a lower speed memory or device. Read cache holds data in anticipation that it will be requested by a client. Write cache holds data written by a client until it can be safely stored on more permanent storage media such as disk or tape. (SNIA)

See also *write-back cache*, *write-through cache*.

capacitor pack The controller module component that provides backup power to transfer unwritten data from cache to Compact Flash memory in the event of a power failure. Storing the data in Compact Flash provides unlimited backup time. The unwritten data can be committed to the disk drives when power is restored.

CAPI Phoenix Configuration API.

channel A physical path used for the transfer of data and control information between storage devices and a RAID controller or a host; or, a SCSI bus in a controller module. The terms channel and port are used interchangeably within the user interfaces.

chassis An enclosure's metal housing.

chunk size The number of contiguous blocks in a stripe on a disk drive in a virtual disk. The number can be adjusted to improve performance. Generally, larger chunks are more effective for sequential reads. See *block*.

CLI The command-line interface that system administrators can use to configure, monitor, and manage Phoenix storage systems. The CLI is accessible from any management host that can access a controller module through an out-of-band Ethernet or RS-232 connection.

controller The control logic in a storage subsystem that performs command transformation and routing, aggregation (RAID, mirroring, striping, or other), high-level error recovery, and performance optimization for multiple storage devices. (SNIA)

A controller is also referred to as a RAID controller.

controller enclosure An enclosure that contains disk drives and one or two controller modules. See *controller module*.

controller module A FRU that contains: a storage controller processor; a management controller processor; out-of-band management interfaces; a LAN subsystem; cache protected by a capacitor pack and Compact Flash memory; host, expansion, and management ports; and midplane connectivity. If a controller enclosure contains redundant controller modules, the upper one is designated *A* and the lower one is designated *B*.

CPLD Complex programmable logic device. A generic term for an integrated circuit that can be programmed in a laboratory to perform complex functions.

DAS See *direct attach storage (DAS)*.

data host A host that reads/writes data to the storage system. A data host can be connected directly to the system (direct attached storage, or DAS) or can be connected to an external switch that supports multiple data hosts (storage area network, or SAN).

data mirroring Data written to one disk drive is simultaneously written to another disk drive. If one disk fails, the other disk can be used to run the virtual disk and reconstruct the failed disk. The primary advantage of disk mirroring is 100 percent data redundancy: since the disk is mirrored, it does not matter if one of the disks fails; both disks contain the same data at all times and either can act as the operational disk. The disadvantage of disk mirroring is that it is expensive because each disk in the virtual disk is duplicated. RAID 1 and 10 use mirroring.

data striping The storing of sequential blocks of incoming data on all the different disk drives in a virtual disk. This method of writing data increases system throughput because multiple disks are working simultaneously, retrieving and storing. RAID 0, 10, 3, 5 and 50 use striping.

DHCP Dynamic Host Configuration Protocol.

direct attach storage (DAS) A dedicated storage device that connects directly to one or more servers. (SNIA)

disk mirroring See *data mirroring*.

DMA Direct memory access.

drive module A FRU consisting of a disk drive and drive sled.

dynamic spare An available disk drive that is used to replace a failed drive in a virtual disk, if the Dynamic Spares feature is enabled and no vdisk spares or global spares are designated.

EIA Enterprise information architecture.

EMP See *enclosure management processor (EMP)*.

enclosure A physical storage device that contains disk drives. If the enclosure contains integrated RAID controllers it is known as a controller enclosure; otherwise it is an expansion enclosure.

enclosure management processor (EMP) A device in the enclosure from which the system can inquire about the enclosure's environmental conditions such as temperature, power supply and fan status, and the presence or absence of disk drives.

expansion enclosure An enclosure that contains disk drives and one or two expansion modules. See *expansion module*.

expansion module A FRU that contains: host, expansion, and management ports; an Enclosure Management Processor; and midplane connectivity. If a system contains redundant expansion modules, the upper one is designated A and the lower one is designated B.

fabric A Fibre Channel switch or two or more Fibre Channel switches interconnected in such a way that data can be physically transmitted between any two N_Ports on any of the switches. (SNIA)

fabric port (F_Port) An F_Port that can support an attached arbitrated loop. An FL_Port on a loop has the AL_PA hex '00' and is the gateway to the fabric for NL_Ports on a loop.

fabric switch A Fabric switch functions as a routing engine that actively directs data transfer from source to destination and arbitrates every connection. Bandwidth per node via a Fabric switch remains constant when more nodes are added, and a node on a switch port uses a data path of up to 100 Mbyte/sec to send or receive data.

fabric-loop port (FL_Port) An F_Port can support an attached arbitrated loop. An FL_Port on a loop has the AL_PA hex'00', giving the fabric highest priority access to the loop. An FL_Port is the gateway to the fabric for NL_Ports on a loop.

failback See *recovery*.

failover In an active-active configuration, failover is the act of temporarily transferring ownership of controller resources from a failed controller to a surviving controller. The resources include virtual disks, cache data, host ID information, and LUNs and WWNs. See also *recovery*.

fault tolerance The capacity to cope with internal hardware problems without interrupting the system's data availability, often by using backup systems brought online when a failure is detected. Many systems provide fault tolerance by using RAID architecture to give protection against loss of data when a single disk drive fails. Using RAID 1, 3, 5, 10, or 50 techniques, the RAID controller can reconstruct data from a failed disk drive and write it to a spare or replacement disk drive.

fault-tolerant virtual disk A virtual disk that provides protection of data in the event of a single disk drive failure by employing RAID 1, 10, 3, 5, or 50.

FC See *Fibre Channel (FC)*.

FC-AL See *Fibre Channel-Arbitrated Loop (FC-AL)*.

Fibre Channel (FC) A set of standards for a serial I/O bus capable of transferring data between two ports at up to 100 Mbyte/sec, with standards proposals to go to higher speeds. Fibre Channel supports point-to-point, arbitrated loop, and switched topologies. Fibre Channel was completely developed through industry cooperation, unlike SCSI, which was developed by a vendor and submitted for standardization after the fact. (SNIA)

Fibre Channel-Arbitrated Loop (FC-AL)
A form of Fibre Channel network in which up to 126 nodes are connected in a loop topology, with each node's L_Port transmitter connecting to the L_Port receiver of the node to its logical right. Nodes connected to a Fibre Channel Arbitrated Loop arbitrate for the single transmission that can occur on the loop at any instant using a Fibre Channel Arbitrated Loop protocol that is different from Fibre Channel switched and point to point protocols. An arbitrated loop may be private (no fabric connection) or public (attached to a fabric by an FL_Port). (SNIA)

field-replaceable unit (FRU)
An assembly component that is designed to be replaced on site, without the system having to be returned to the manufacturer for repair.

FRU
See *field-replaceable unit (FRU)*.

Gbyte
Gigabyte (GB).

global spare
A spare disk drive that is available to all virtual disks in a system.

HBA
See *host bus adapter (HBA)*.

host bus adapter (HBA)
An I/O adapter that connects a host I/O bus to a computer's memory system (SNIA).

hot swap
The ability to remove and replace a FRU while the system is powered on and operational.

in-band management
Transmission of a protocol other than the primary data protocol over the same medium as the primary data protocol. Management protocols are a common example of in-band transmission. (SNIA)

This type of access is available through use of the Phoenix Configuration API (CAPI) to develop a programmed interface.

initialization
The process of writing a specific pattern to all data blocks on all disk drives in a virtual disk. This process overwrites and destroys existing data on the disk drives and the virtual disk. Initialization is required to make the entire virtual disk consistent at the onset. Initialization ensures that virtual-disk verifications performed in the future are executed correctly.

I/O Input/output.

IP Internet Protocol.

JBOD Just a Bunch of Disks. An expansion enclosure that is directly attached to a host.

Kbyte Kilobyte (KB).

LAN See *local area network (LAN)*.

leftover drive A disk drive that contains metadata but is no longer part of a virtual disk.

local area network (LAN) Local Area Network. A communications infrastructure designed to use dedicated wiring over a limited distance (typically a diameter of less than five kilometers) to connect to a large number of intercommunicating nodes. (SNIA)

logical unit number (LUN) The SCSI identifier of a logical unit with a target. (SNIA)

For example, a LUN identifies the mapping between a volume (logical unit) and a host port (target).

loop address Indicates the unique ID of a node in FC loop topology. A loop address is sometimes referred to as a Loop ID.

loop port (L_Port) A “Loop” port is capable of performing arbitrated loop functions and protocols. NL_Ports and FL_Ports are examples of loop-capable ports. (SNIA)

loop topology See *Fibre Channel-Arbitrated Loop (FC-AL)*.

LUN See *logical unit number (LUN)*.

management controller (MC) The processor (located in a controller module) that is primarily responsible for human-computer interface and computer-computer interface functions, and interacts with the storage controller.

management host A workstation with a direct or local connection to the system and that is used to manage the system.

management information base (MIB) A database of managed objects accessed by network management protocols. An SNMP MIB is a set of parameters that an SNMP management station can query or set in the SNMP agent of a network device (for example, a router).

master volume A volume that is enabled for snapshots. A master volume must be owned by the same controller as the associated snap pool.

Mbyte Megabyte (MB).

MC See *management controller (MC)*.

metadata Data in the first sectors of a disk drive that the system uses to identify virtual disk members.

MIB See *management information base (MIB)*.

node port (N_Port) A port on a computer, disk drive, or other device through which the device does its FC communication.

node-loop port (NL_Port) An N_Port that can operate on FC-AL topology.

Non-RAID The RAID level option that can be used for a virtual disk having a single disk drive and that does not need the data redundancy or performance benefits of RAID. The capacity of a non-RAID virtual disk equals the capacity of its disk drive. For fault tolerance, use RAID 0 or above.

out-of-band management Method of accessing and managing a system using the RS-232 or Ethernet connection.

ownership In an active-active configuration, one controller has ownership of the following resources: virtual disks and vdisk spares. When a controller fails, the other controller assumes temporary ownership of its resources.

PHY Hardware component that converts between digital and analog in the signal path between the storage controller, expander controller, disk drives, and SAS ports.

PID Primary controller identifier number.

port interconnect A dual-controller enclosure includes host port interconnect circuitry which can be used to connect the host ports on the upper controller module to those on the lower controller module. When enabled, the port interconnect gives each host access to all the volumes assigned to both controllers and makes it possible to create a redundant configuration without using an external Fibre Channel switch. The port interconnect should only be enabled when the system is used in direct attach configurations. When using a switch attached configuration, the port interconnect must be disabled.

power and cooling module A FRU that includes an AC power supply and two cooling fans. An enclosure has two modules for failure tolerance and can operate with only one module.

priority Priority enables controllers to serve other I/O requests while running jobs (utilities) such as rebuilding virtual disks. Priority ranges from low, which uses the controller's minimum resources, to high, which uses the controller's maximum resources.

RAID Redundant Array of Independent Disks, a family of techniques for managing multiple disks to deliver desirable cost, data availability, and performance characteristics to host environments. (SNIA)

RAIDar The web browser interface that system administrators can use to configure, monitor, and manage Phoenix storage systems. RAIDar is accessible from any management host that can access a system through an out-of-band Ethernet connection.

RAID controller See *controller*.

RAS Reliability, availability, and serviceability. These headings refer to a variety of features and initiatives all designed to maximize equipment uptime and mean time between failures, minimize downtime and the length of time necessary to repair failures, and eliminate or decrease single points of failure in favor of redundancy.

rebuild The regeneration and writing onto one or more replacement disks of all of the user data and check data from a failed disk in a virtual disk with RAID level 1, 10, 3, 5, and 50. A rebuild can occur while applications are accessing data on the system's virtual disks.

recovery In an active-active configuration, recovery (also known as failback) is the act of returning ownership of controller resources from a surviving controller to a previously failed (but now active) controller. The resources include virtual disks, cache data, host ID information, and LUNs and WWNs.

remote scripting CLI client A command-line interface (CLI) that enables you to manage the system from a remote management host. The client communicates with the management software through a secure out-of-band interface, HTTPS, and provides the same control and monitoring capability as the browser interface. The client must be installed on a host that has network access to the system.

rollback The process of resetting a volume's data to become identical to a snapshot taken of that volume.

SAN See *Storage Area Network (SAN)*.

SAS Serial Attached SCSI.

SATA Serial Advanced Technology Attachment.

SC See *storage controller (SC)*.

SCSI Small Computer System Interface. A collection of ANSI standards and proposed standards which define I/O buses primarily intended for connecting storage subsystems or devices to hosts through host bus adapters. (SNIA)

SCSI Enclosure Services (SES) An ANSI X3T10 standard for management of environmental factors such as temperature, power, voltage, etc. (SNIA)

In Phoenix storage systems, SES data is managed by enclosure management processors (EMPs).

SFP Small form-factor pluggable connector, used in FC controller module host ports. An SFP is a FRU.

SID Secondary controller identifier number.

SMART Self-Monitoring Analysis and Reporting Technology. The industry-standard reliability prediction indicator for both the IDE/ATA and SCSI hard disk drives. Hard disk drives with SMART offer early warning of some hard disk failures so critical data can be protected.

SMI-S Storage Management Interface - Specification.

SMTP Simple Mail Transfer Protocol. A protocol for sending email messages between servers and from mail clients to mail servers. The messages can then be retrieved with an email client using either POP or IMAP.

snap pool A volume that is configured to store snapshot data.

snapshot A fully usable copy of a defined collection of data that contains an image of the data as it appeared at the point in time at which the copy was initiated. (SNIA)

SNIA Storage Networking Industry Association.

SNMP Simple Network Management Protocol. An IETF protocol for monitoring and managing systems and devices in a network. The data being monitored and managed is defined by a MIB. The functions supported by the protocol are the request and retrieval of data, the setting or writing of data, and traps that signal the occurrence of events. (SNIA)

spare See *dynamic spare*, *global spare*, *vdisk spare*.

standard volume A volume that is not enabled for snapshots.

standby See *spare*.

state The current operational status of a disk drive, a virtual disk, or controller. A controller module stores the states of drives, virtual disks, and the controller in its nonvolatile memory. This information is retained across power interruptions.

Storage Area Network (SAN) A storage system consisting of storage elements, storage devices, computer systems, and/or appliances, plus all control software, communicating over a network. (SNIA)

storage controller (SC) The processor (located in a controller module) that is primarily responsible for RAID controller functions. The storage controller is also referred to as the RAID controller.

storage system One or more enclosures, referred to in a logical (as opposed to physical) sense.

strip size See *chunk size*.

stripe size The number of data disks in a virtual disk multiplied by the chunk size.

subvirtual disk One of multiple RAID 1 virtual disks across which data is striped to form a RAID 10 virtual disk; or one of multiple RAID 5 virtual disks across which data is striped to form a RAID 50 virtual disk.

system See *storage system*.

Tbyte Terabyte (TB).

TCP/IP Transmission Control Protocol/Internet Protocol.

topology The logical layout of the components of a computer system or network and their interconnections. Topology deals with questions of what components are directly connected to other components from the standpoint of being able to communicate. It does not deal with questions of physical location of components or interconnecting cables. (SNIA)

trap A type of SNMP message used to signal that an event has occurred. (SNIA)

UPS Uninterruptible Power Supply.

vdisk Abbreviation for virtual disk.

vdisk spare A disk drive that is marked as a spare to support automatic data rebuilding after a disk drive associated with a virtual disk fails. For a vdisk spare to take the place of another disk drive, it must be at least equal in size to the failed disk drive and all of the virtual disks dependent on the failed disk drive must be redundant—RAID 1, 10, 3, 5, or 50.

verify A process that checks the integrity of the redundant data on fault-tolerant virtual disks. For RAID 3, 5, and 50, the verify process recalculates the parity of data stripes in each of the virtual disk's RAID stripe sets and compares it with the stored parity. If a discrepancy is found, an error is reported and the new correct parity is substituted for the stored parity. For RAID 1 and 10, the verify process checks for mirror mismatches. If an inconsistency is encountered, data is copied from the master disk drive to the slave disk drive. If a bad block is encountered when the parity is regenerated, the data is copied from the other disk drive, master or slave, to the reporting disk drive reallocating the bad block.

virtual disk For Phoenix storage systems, a set of disk drives that share a RAID level and disk type, and across which host data is spread for redundancy or performance.

volume A logical subdivision of a virtual disk. Multiple LUNs can be assigned to the same volume, one for each host port given access to the volume. See also *standard volume*.

volume mapping The process by which volume permissions (read only, read/write, or none) and LUNs are assigned to a host port.

WBI See *RAIDar*.

web-browser interface (WBI) See *RAIDar*.

world wide name (WWN) A unique 64-bit number assigned by a recognized naming authority (often via block assignment to a manufacturer) that identifies a node process or node port. (SNIA)

Phoenix storage systems derive WWNs from the serial numbers of controller modules and expansion modules.

world wide node name (WWNN) A globally unique 64-bit identifier assigned to each Fibre Channel node process. (SNIA)

world wide port name (WWPN) A globally unique 64-bit identifier assigned to each Fibre Channel port. (SNIA)

write policy A cache-writing strategy used to control write operations. The write policy options are CIFS write-back and write-through cache.

write-back cache A caching technique in which the completion of a write request is signaled as soon as the data is in cache, and actual writing to non-volatile media occurs at a later time. Write-back cache includes an inherent risk that an application will take some action predicated on the write completion signal, and a system failure before the data is written to non-volatile media will cause media contents to be inconsistent with that subsequent action. For this reason, good write-back cache

implementations include mechanisms to preserve cache contents across system failures (including power failures) and to flush the cache at system restart time. (SNIA)

This is how Phoenix storage systems operate. See also *write-through cache*.

write-through cache

A caching technique in which the completion of a write request is not signaled until data is safely stored on non-volatile media. Write performance with a write-through cache is approximately that of a non-cached system, but if the data written is also held in cache, subsequent read performance may be dramatically improved. (SNIA)

Phoenix storage systems use write-through cache when write-back cache is disabled or when cache backup power is not working. See also *write-back cache*.

Index

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X