SANworks by Compaq

Storage System Scripting Utility V2.0 for Enterprise Virtual Array Reference Guide

Part Number: AA-RPK7B-TE

Second Edition (July 2002)

Product Version:2.0

This guide describes the commands available in the Storage System Scripting Utility V2.0 for the Enterprise Virtual Array. The purpose of these commands is to configure and control HSV controllers.

COMPAQ

Printed in the U.S.A.

Storage System Scripting Utility V2.0 for Enterprise Virtual Array Reference Guide
Second Edition (July 2002)
Part Number: AA-RPK7B-TE

Contents

A Configuration Examples

Glossary

Index

Tables

About this Guide

The following sections are covered:

- Intended Audience
- Related Documentation
- Text Conventions
- Symbols in Text
- Symbols on Equipment
- Rack Stability
- Getting Help
- Compaq Authorized Reseller

Intended Audience

This guide is intended for storage administrators, who are experienced with the following:

- Configuration of SAN fabrics.
- Customer host environments, such as Windows 2000, Windows NT, Sun Solaris, Compaq OpenVMS, Compaq Tru64 UNIX, HP-UX, and IBM AIX.

Related Documentation

This section lists documents that provide additional information:

- *SANworks by Compaq Host Operating System V2.0 Kit for Enterprise Virtual Array Installation and Configuration Guide* (available for Sun Solaris, IBM AIX, Windows, Tru64 UNIX, OpenVMS, and HP-UX)
- *SANworks by Compaq Host Operating System V2.0 Kit for Enterprise Virtual Array Release Notes* (available for Sun Solaris, IBM AIX, Windows, Tru64 UNIX, OpenVMS, and HP-UX)
- *StorageWorks by Compaq Enterprise Virtual Array Users Guide*
- *StorageWorks by Compaq Enterprise Virtual Array - How to Get More Information*
- *StorageWorks by Compaq Enterprise Virtual Array Release Notes*
- *StorageWorks by Compaq Enterprise Virtual Array Storage System Read Me First*
- *StorageWorks by Compaq Enterprise Virtual Array Storage System World Wide Name Label*
- *StorageWorks by Compaq Interactive Help for HSV Element Manager*
- *StorageWorks by Compaq Enterprise Virtual Array Configuration Guide*
- *StorageWorks by Compaq Enterprise Virtual Array Upgrade Instructions*
- *StorageWorks by Compaq Enterprise Virtual Array License Instructions*
- *StorageWorks by Compaq Enterprise Virtual Array Catalog of Associated Documentation*
- *StorageWorks by Compaq Enterprise Virtual Array Storage System Installation Instructions*
- *StorageWorks by Compaq Heterogeneous Open SAN Design Reference Guide*

Text Conventions

The conventions included in Table 1 apply in most cases.

Table 1: Text Conventions

Element	Convention	Examples
• **Named Keys** • **Key Sequences**	**Bold** A plus sign (**+**) between two keys means that you should press them simultaneously.	**Home, Print Screen, Num Lock, Esc, PgUp** **Ctrl+A**, **Ctrl+Home**, **Alt+Ctrl+Del** To back up files, click **Backup Now**.
• **Menu Items and Sequences** • **Buttons** • **Dialog Box Names**	**Bold** A right angle bracket (**>**) between items refers to the sequence for navigating through menu selections.	Choose **Start > Programs > Backup.** Click **Make Backup**. In the **Save As** dialog box, choose the drive, then the folder.
• Directory Names and Paths	Initial Caps (for UNIX-based directory names, the exact case is displayed).	Save the file in the C:\StorageSets\Default directory. (UNIX-based): Save the file in the /home/newuser/practice directory.
• *file names*	*Italics* unless the file name is included in a directory name/path.	To configure storage, edit *storageset.ini*. (Directory name/path): Errors are logged to \syslog\errors\config_errors.txt.

Table 1: Text Conventions (Continued)

Element	Convention	Examples
• User Input and System Responses (Output and Error Messages) • COMMAND NAMES • Drive Names	`Monospace font.` `COMMAND NAMES` appear in uppercase, unless they are case sensitive (UNIX-based command names are case sensitive and will not appear in uppercase). Entered `<variables>` are displayed in angle brackets (`< >`) and all lowercase.	User Input and System Responses: • To exit from the program, type `exit`. • At the prompt, type this command: `SHOW THIS_CONTROLLER` (no variable). • To see your settings, give the command: `SHOW <storagesets> FULL` (with variable).You will see the `Disk Full` message. Command Names • Use `SET THIS_CONTROLLER` to change parameters. • To manage storage, `RUN SYSMGR.EXE`. • (UNIX-based): To list files, use the `ls` command. • Drive Names: Navigate to your CD-ROM drive (usually `D:` or `E:`).
URLs	Sans serif font.	For update notices, visit: http://www.compaq.com/products/updates

Symbols in Text

These symbols may be found in the text of this guide. They have the following meanings.

WARNING: Text set off in this manner indicates that failure to follow directions in the warning could result in bodily harm or loss of life or damage to equipment.

CAUTION: Text set off in this manner indicates that failure to follow directions could result in damage to equipment or loss of information.

IMPORTANT: Text set off in this manner presents clarifying information or specific instructions.

NOTE: Text set off in this manner presents commentary, sidelights, or interesting points of information.

Symbols on Equipment

Any enclosed surface or area of the equipment marked with these symbols indicates the presence of electrical shock hazards. Enclosed area contains no operator serviceable parts.

WARNING: To reduce the risk of injury from electrical shock hazards, do not open this enclosure.

Any RJ-45 receptacle marked with these symbols indicates a network interface connection.

WARNING: To reduce the risk of electrical shock, fire, or damage to the equipment, do not plug telephone or telecommunications connectors into this receptacle.

Any surface or area of the equipment marked with these symbols indicates the presence of a hot surface or hot component. Contact with this surface could result in injury.

WARNING: To reduce the risk of injury from a hot component, allow the surface to cool before touching.

Power supplies or systems marked with these symbols indicate the presence of multiple sources of power.

WARNING: To reduce the risk of injury from electrical shock, remove all power cords to completely disconnect power from the supplies and systems.

Any product or assembly marked with these symbols indicates that the component exceeds the recommended weight for one individual to handle safely.

WARNING: To reduce the risk of personal injury or damage to the equipment, observe local occupational health and safety requirements and guidelines for manually handling material.

Rack Stability

WARNING: To reduce the risk of personal injury or damage to the equipment, be sure that:

- **The leveling jacks are extended to the floor.**
- **The full weight of the rack rests on the leveling jacks.**
- **In single rack installations, the stabilizing feet are attached to the rack.**
- **In multiple rack installations, the racks are coupled together.**
- **Only one rack component is extended at any time. A rack may become unstable if more than one rack component is extended for any reason.**

Getting Help

If you still have a question after reading this guide, contact your service representatives or visit our website.

Compaq Technical Support

In North America, call the Compaq technical support at 1-800-OK-COMPAQ. This service is available 24 hours a day, 7 days a week.

NOTE: For continuous quality improvement, calls may be recorded or monitored.

Outside North America, call Compaq technical support at the nearest location. Telephone numbers for worldwide technical support are listed on the Compaq website: http://www.compaq.com.

Be sure to have the following information available before you call Compaq:

- Technical support registration number (if applicable)
- Product serial numbers
- Product model names and numbers
- Applicable error messages
- Operating system type and revision level
- Detailed, specific questions

Compaq Website

The Compaq website has the latest information on this product as well as the latest drivers. Access the Compaq website at: http://www.compaq.com/storage. From this website, select SANworks.

Compaq Authorized Reseller

For the name of your nearest Compaq Authorized Reseller:

- In the United States, call 1-800-345-1518.
- In Canada, call 1-800-263-5868.
- Elsewhere, see the Compaq website for locations and telephone numbers.

1

Introduction

This chapter introduces the *SANworks*™ by Compaq Enterprise Storage System Scripting Utility (SSSU) V2.0 for the *StorageWorks*™ by Compaq Enterprise Virtual Array. The following topics are covered:

Description

The Enterprise Storage System Scripting Utility (SSSU) for Enterprise Virtual Array is a command line application that allows you to configure and control HSV controllers.

You can execute configuration requests using the HSV Element Manager or SSSU. Simple or initial configuration requests can be handled easily and expediently through the HSV Element Manager, but repetitious and complex configuration steps can be scripted and executed through the character cell interface of Storage System Scripting Utility.

Formatting and Output of Returned Data

All returned SHOW command data is formatted in the form of:

```
XMLtag: Data
```

Where the XML tag displays before the data, and then the tag's data displays. No additional formatting is available, except where noted in the command reference (see Chapter 2).

Definitions

The definitions in Table 1–1 translate Storage System Scripting Utility terms into HSV Element Manger terms.

Table 1–1: Definition of Terms

Term	Definition
Cell	A controller or controller pair that is treated conceptually as one controller. In the HSV Element Manager, this appears as the name you give the initialized storage system. Cell is also referred to as a **Storage Cell**.
Disk	A physical storage device (disk drive) connected to the cell.
Group	A collection of disks that storage is created from. Group is also referred to as a **Disk Group**.

Table 1–1: Definition of Terms (Continued)

Term	Definition
Spare Policy	Disk failure protection level: NONE, SINGLE, and DOUBLE. • NONE—reserves no space within a disk group to allow for data reconstruction for failure of disk drives • SINGLE—reserves space within a disk group to allow for data reconstruction for failure of a single disk drive • DOUBLE—reserves space within a disk group to allow for data reconstruction for failure of two disk drives
Host	A system that has data path access to the logical unit numbers (LUNs) defined on the cell.
Manager	The object that controls configuration of the cell. The StorageWorks by Compaq HSV Element Manager is the manager in the case of the Enterprise Virtual Array.
Storage	A collection of blocks created on one or more disks that can be used by a host for information storage and retrieval. Storage is also referred to as a **Virtual Disk**.
LUN	The host-accessible presentation of storage on the cell.

Installing the Enterprise Storage System Scripting Utility

SSSU is installed from your host operating system kits. See the host operating system's installation guide for details on how to install SSSU.

Once installed, SSSU can run as a simple executable. For convenience, you may wish to add the directory containing the Storage System Scripting Utility executable to your path, or copy the executable to a directory already in your path. Ensure that the Storage System Scripting Utility executable's attributes are set with the correct security and execution flags appropriate for your environment.

The executable file is named *sssu* or *SSSU.EXE*, depending on the operating system environment.

Starting the Enterprise Storage System Scripting Utility

In order to execute the Storage System Scripting Utility, ensure the password access to the controller (if enabled) is already set up from the HSV Element Manager software. You cannot set this password from within Storage System Scripting Utility.

NOTE: The SET OPTIONS command (see "SET OPTIONS" on page 2–42) is used to control how SSSU behaves. The options you set with the SET OPTIONS command are in effect for the current session only. Each time you start SSSU, the default options are reinstated.

Storage System Scripting Utility is started at a command prompt window or equivalent.

IMPORTANT: The CAPTURE CONFIGURATION command (see "CAPTURE CONFIGURATION" on page 2–19) is the only way to capture, save, and re-create a storage system's configuration. After you have successfully created a storage system, use the CAPTURE CONFIGURATION command to create a script that you can use to re-create a storage system in the event of failure.

Syntax

```
SSSU <additional arguments>
```

If Storage System Scripting Utility is started without arguments, a generic application `NoCellSelected>` prompt is displayed on the terminal and input can be accepted.

If Storage System Scripting Utility is started with arguments, those commands are echoed to the input terminal and executed, and then the utility exits.

NOTE: You must enclose in double quotes any commands or object names that have embedded blanks (spaces).

Examples

```
SSSU

SSSU "FILE snap d1.txt"
```

The first example starts Storage System Scripting Utility without additional arguments and prompts you for commands.

The second example starts Storage System Scripting Utility and executes the file named *snap d1.txt* and then exits.

Understanding Paths and Naming

This section describes important information about paths and naming conventions with Storage System Scripting Utility.

NOTE: In order to explain path and naming conventions, commands are used that have not yet been explained. For help understanding the actual commands, refer to the appropriate section of Chapter 2, "Enterprise Storage System Scripting Utility Command Reference."

There are three important root folders you will use in organizing your storage system:

- \Hosts
- "\Virtual Disks"
- "\Disk Groups"

NOTE: The double quotes ("") around some names are necessary because of the space within the name.

You cannot create root-level folders. These three are the only root folders, but you can create additional folders below the \Hosts and "\Virtual Disks" root levels to organize your storage system (see "ADD FOLDER" on page 2–7).

NOTE: Storage System Scripting Utility requires that you qualify specified names with full paths. However, if you are using the default placement when adding Hosts and Virtual Disks, it is not necessary to include the full path.

Hosts Example

For example, to add a host named *engineering* to the root Hosts folder:

```
ADD HOST \Hosts\engineering WORLD_WIDE_NAME=1111-2222-3333-4444
```

Whenever you refer to this host, you must give the full path. For example, when adding a LUN to this host:

```
ADD LUN 4 HOST=\Hosts\engineering STORAGE=<virtual_disk_name>
```

If you create a folder structure within the root Hosts folder, you must include all levels of the folder structure in your commands.

For example, to add a host named *engineering* to a sub-folder named *resources* within the Host root folder:

```
ADD HOST \Hosts\resources\engineering
WORLD_WIDE_NAME=1111-2222-3333-4444
```

This example creates a host named *engineering* in the root Hosts folder.

Virtual Disk Example

Because of the space in the root name for the Virtual Disks folder, you must always enclose strings that include the name of this folder within double quotes (“”).

For example, to add a virtual disk family named *new_code* to the root folder Virtual Disks:

```
ADD STORAGE "\Virtual Disks\new_code" SIZE=10
```

This example creates a 10 GB virtual disk family in the default disk group named *new_code*.

NOTE: This is a virtual disk family. The actual virtual disk name (the active virtual disk) is *"\Virtual Disks\new_code\ACTIVE."* The family (*new_code*) only consists of this one virtual disk (*ACTIVE*), unless a copy or snapshot is made.

Whenever you refer to this virtual disk, you must always give the full path and enclose it in double quotes.

For example, referencing this same virtual disk (*new_code*) within another command, give the full path, add the *\ACTIVE,* and be sure to include everything within double quotes:

```
SET STORAGE "\Virtual Disks\new_code\ACTIVE" WRITE_PROTECT
```

If you create a virtual disk (ADD STORAGE) within a deeper folder structure, you need to include this full path within double quotes.

Disk Groups Example

As mentioned previously, when a path includes the root folder “\Disk Groups” you must enclose the entire path in double quotes:

```
SHOW GROUP "\Disk Groups\admin"

ADD STORAGE "\Virtual Disk\new_code" SIZE=10 GROUP="\Disk
Groups\admin"
```

Getting Help

You can display a list of options for each command by entering a space and question mark (?) after the command or option name. This displays a list of parameters or options that you can enter with the present command.

For example, to get information on what options are available with the ADD command:

Type `ADD ?` at the command prompt:

```
NoCellSelected> ADD ?
```

The following list of options is presented:

```
The options are:
CELL
COPY
FOLDER
GROUP
HOST
LUN
SNAPSHOT
STORAGE
```

The prompt returns and you can continue entering commands.

In addition there is a HELP command that descrives how to use the command syntax help (see “HELP” on page 2–29).

2

Enterprise Storage System Scripting Utility Command Reference

This chapter defines all the commands available in Storage System Scripting Utility. The following topics are covered:

Syntax and examples are given for each command. The commands are presented in alphabetical order.

Command Introduction

Most of the commands have subcommands or switches that you can also use. The generic term “options” is used for both subcommands and switches in the utility’s help. These options are described under the entry for the command in this chapter.

You can display a list of options for each command directly within the interface by entering a “?” after the command or option name. For example, “ADD ?” displays all of the subcommands available with the ADD command, and “ADD CELL ?” displays the available switches for the ADD CELL command (see “Getting Help” on page 1–7).

NOTE: Remember that when you issue commands:

— Specified names must be qualified by their full path

— If a path name contains a space, the entire name must be enclosed in double quotes(“”)

For a full discussion of these requirements, see “Understanding Paths and Naming” on page 1–5.

Appendix A, “Configuration Examples,” provides a simple configuration example to help you understand how the commands are used in sequence to create and present a single LUN to a host. Sample output from SHOW commands are also provided.

ADD

The ADD commands are used to create new cells, folders, groups, virtual disks, copies/snapshots, as well as to add LUNs and hosts within the Enterprise Virtual Array.

NOTE: Creating snapshots of virtual disks is dependent on licensing level. See the documentation that came with your hardware for more information.

ADD CELL

The ADD CELL command creates an initialized cell and the default disk group. You use this command to initialize an uninitialized storage system. If the cell is already initialized, the command is rejected.

Attempting to add a cell (create storage system) without the license, will return an error message. This message indicates that you need to enter your licensing information in the HSV Element Manager. That is, you must have a license and follow the registration process.

You must select an uninitialized storage cell before issuing the ADD CELL command. Use the SHOW CELL command (see "SHOW CELL" on page 2–48) to discover the names of the available uninitialized storage systems. The uninitialized storage system display as: *Uninitialized Storage System#>*, where # represents a number. After initialization is complete, SSSU changes its default prompt back to NoCellSelected>. You are required to select the cell again using the new name of the cell. This requirement is because the cell's name changed from the uninitialized string to the name given with the ADD CELL command.

NOTE: Appendix A gives a configuration example that explains the sequence of commands to initialize an uninitialized storage system (see "Creating a Simple Configuration" on page A–2).

Syntax

```
ADD CELL <cell_name>
```

Switches

The following switches are available with the ADD CELL command:

- COMMENT=
- CONSOLE _LUN_ID=
- DEVICE _COUNT=
- SPARE_POLICY=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the new cell. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

CONSOLE_LUN_ID=

The LUN used for console communication after cell creation. If set to zero, no console LUN is presented to the host.

NOTE: This value is used for IBM AIX (set to zero), OpenVMS (required), and Tru64 UNIX (recommended). Other host operating systems will ignore the value. See the host operating system installation guides for more information.

DEVICE_COUNT=

The number of physical disks to use for this cell. The limit is the number of available disks in the storage system. If no value is specified, all unused physical disks in the storage system are used.

SPARE_POLICY=

This parameter determines the amount, if any, of storage space set aside for use in the event disks fail. The default SPARE_POLICY is SINGLE.

- NONE—reserves no space within a disk group to allow for data reconstruction for failure of disk drives
- SINGLE—reserves space within a disk group to allow for data reconstruction for failure of a single disk drive (the default)
- DOUBLE—reserves space within a disk group to allow for data reconstruction for failure of two disk drives

NOTE: The space set aside is not in numbers of physical disks. It is the equivalent amount of storage space spread across all disks.

Example

```
ADD CELL payroll DEVICE_COUNT=12 SPARE_POLICY=SINGLE
```

This example creates an initialized storage system (cell) named *payroll* with 12 physical disks and a spare policy of *SINGLE*.

ADD COPY

The ADD COPY command creates a copy of the specified storage. The ADD COPY command is equivalent to creating a clone within the HSV Element Manager software.

Syntax

```
ADD COPY <copy_name> STORAGE=<storage_to_copy>
```

Switches

The following switches are available with the ADD COPY command:

- GROUP=
- OS_UNIT_ID=
- STORAGE=
- WORLD_WIDE_LUN_NAME=

These switches are described in the following paragraphs.

GROUP=

The disk group name where you want to create the virtual disk copy. The disk group must already exist to use this switch. If not specified, a controller-dependent default is used.

NOTE: You must specify the complete path from the root directory to the GROUP and enclose the entire string in double quotes.

OS_UNIT_ID=

The ID that is presented to the host operating system. If set to zero, no ID will be presented to the host.

NOTE: This is a required field for OpenVMS and is recommended for use with Tru64 UNIX. It is optional for all other operating systems.

STORAGE=

The storage name that is being copied.

NOTE: You must specify the complete path from the root directory to the STORAGE and enclose the entire string in double quotes.

WORLD_WIDE_LUN_NAME=

Used to set the World Wide LUN Name on unpresented storage (either /ACTIVE or snapshot). This switch is rejected when issued against storage that is presented to a host.

Required Switch

The following switch is required with the ADD COPY command:

- STORAGE=

Examples

```
ADD COPY wednesday_night STORAGE="\Virtual Disks\payroll\ACTIVE"

ADD COPY save_reports STORAGE="\Virtual
Disks\daily_business\ACTIVE" GROUP="\Disk Groups\small_disks"
```

The first example creates a copy of the virtual disk named *payroll* to a copy named *wednesday_night*. The second example creates a copy of *daily_business* named *save_reports* within the specified group.

NOTE: To delete storage created by ADD COPY, use the DELETE STORAGE command.

ADD FOLDER

The ADD FOLDER command creates a new folder within the current folder to aid in organizing your storage system. Folders can only be created under the "Virtual Disks" and "Hosts" root folders. You cannot create root folders.

For example, if you have a controller that is serving Human Resources and Engineering, you could create four folders—two to separate the virtual disks and two to separate the hosts:

- \Virtual Disks\Engineering and \Virtual Disks\HR
- \Hosts\Engineering and \Hosts\HR

Creating these folders allows you to put engineering storage and hosts in the Engineering folders and Human Resources storage/hosts in the HR folders. This makes it easier to keep track of the pieces within your storage system. If you want to add multiple folder layers, they must be added one layer at a time.

Syntax

```
ADD FOLDER <folder_name>
```

Switch

Only the COMMENT switch is available with the ADD FOLDER command.

COMMENT=

This is an optional switch that associates a user-defined comment with the new folder. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

Examples

```
ADD FOLDER \Hosts\human_resources

ADD FOLDER "\Virtual Disks\colorado\colorado_springs\engineering"
```

The first example creates a new folder named *human_resources* within the root folder *Hosts*. The second example creates a folder named *engineering* in the folder *colorado_springs*, which is two levels deep from the root folder *Virtual Disks (“\Virtual Disks\colorado\colorado_springs”)*.

The folder structure in the second example must already have been in place before creating the new folder *engineering*. Also note that the entire path is enclosed in double quotes because of the space in the root folder name (*Virtual Disks*).

ADD GROUP

The ADD GROUP command adds a disk group to the controller. It is good practice to use multiple disk groups because it allows for some failure isolation.

If all the disks are in one group, any disk failure could cause all the virtual disks to run in degraded mode until the data on the disk that was lost is recreated on other disks. By creating multiple disk groups, a disk failure will only affect the virtual disks in that one group.

CAUTION: Vraid1 and Vraid5 virtual disk family with a group spare policy of SINGLE or DOUBLE ensures the space needed to regenerate data is available. If the spare policy is set to NONE, you cannot be certain that there will be enough available space in the disk group to regenerate Vraid1 or Vraid5 data.

Syntax

```
ADD GROUP <group_name>
```

Switches

The following switches are available with the ADD GROUP command:

- COMMENT=
- DEVICE_COUNT=
- OCCUPANCY_ALARM=
- SPARE_POLICY=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the new group. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

DEVICE_COUNT=

The number of physical disks to use for this group. The limit is the number of available disks in the storage system. If no value is specified, all unused physical disks in the storage system are used.

OCCUPANCY_ALARM=

The point when a defined percentage of space is used. When this point is reached, an event log is generated and sent to the Management Appliance (optionally the host system) informing the administrator that the group is becoming full. Do not use the percent sign (%) after the number.

SPARE_POLICY=

This parameter determines the amount, if any, of storage space set aside for use in the event disks fail.

- NONE—reserves no space within a disk group to allow for data reconstruction for failure of disk drives
- SINGLE—reserves space within a disk group to allow for data reconstruction for failure of a single disk drive
- DOUBLE—reserves space within a disk group to allow for data reconstruction for failure of two disk drives

NOTE: The space set aside is not in numbers of physical disks. It is the equivalent amount of storage space spread across all disks.

Example

```
ADD GROUP "\Disk Groups\human_resources" DEVICE_COUNT=12
SPARE_POLICY=SINGLE OCCUPANCY_ALARM=75
```

This example creates a new disk group named *human_resources*. It has 12 physical disks with the equivalent of one disk set aside as a spare and sends an event log to the Management Appliance when 75% of capacity is reached.

ADD HOST

The ADD HOST command adds the World Wide Name (WWN) of the host port to the list of hosts that can connect to virtual disks within the current cell.

The ADD HOST command will add the first FCA only. Each subsequent FCA is added with the SET HOST command (see page 2–39).

Syntax

```
ADD HOST <host_name> WORLD_WIDE_NAME=<world_wide_name>
```

Switches

The following switches are available with the ADD HOST command:

- COMMENT=
- OPERATING_SYSTEM=
- WORLD_WIDE_NAME=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the new host. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

OPERATING_SYSTEM=

The operating system type of the specified host. The operating system may be one of the following (they are spelled here as SSSU expects):

- HPUX
- IBMAIX
- OPEN_VMS
- SOLARIS
- TRU64
- WINDOWS

If unspecified, the operating system defaults to Windows.

WORLD_WIDE_NAME=

The World Wide Name of the host port.

Required Switch

The following switch must be set to ensure the HSV Element Manager recognizes the host added with the ADD HOST command.

- WORLD_WIDE_NAME

Example

```
ADD HOST \Hosts\development WORLD_WIDE_NAME=5000-1fe1-ff00-0000
```

This example adds a host named *development* with an adapter at a WWN of *5000-1fe1-ff00-0000*.

ADD LUN

The ADD LUN command makes previously created storage available to a host.

Syntax

```
ADD LUN <LUN-name> STORAGE=<storage_name> HOST=<host_name>
```

Switches

The following switches are available with the ADD LUN command:

- HOST=
- STORAGE=

These switches are described in the following paragraphs.

HOST=

The name of the host that the LUN will be presented to.

STORAGE=

The name of the storage that will be presented to the host.

NOTE: You must specify the complete path from the root directory to both STORAGE and HOST. In the case of STORAGE, you must enclose the entire string in double quotes.

Required Switches

The following switches must be set for the HSV Element Manager to recognize the LUN you add with the ADD LUN command.

- HOST
- STORAGE

Examples

```
ADD LUN 12 STORAGE="\Virtual Disks\accounting\payroll\ACTIVE"
HOST=\Hosts\sanfran
```

```
ADD LUN 175 STORAGE="\Virtual Disks\user_disk\ACTIVE"
HOST=\Hosts\corporate
```

The first example adds LUN 12, presenting the virtual disk (storage) named *payroll* to the host called *sanfran.* The second example presents `user_disk` to host *corporate* as LUN *175*.

ADD SNAPSHOT

The ADD SNAPSHOT command creates an instantaneous snapshot of the specified virtual disk.

NOTE: Creating snapshots of virtual disks is dependent on your licensing level.

Attempting to add a snapshot without the license, will return an error message. This message indicates that you need to enter your licensing information in the HSV Element Manager. That is, you must have a license and follow the registration process.

Syntax

```
ADD SNAPSHOT <snapshot_name> STORAGE=<storage_to_snapshot>
```

Switches

The following switches are available with the ADD SNAPSHOT command.

- ALLOCATION_POLICY=
- OS_UNIT_ID
- STORAGE=
- WORLD_WIDE_LUN_NAME=

These switches are described in the following paragraphs.

ALLOCATION_POLICY=

The ALLOCATION_POLICY switch indicates how the space for the snapshot is allocated.

- DEMAND—Storage is allocated for the snapshot only when required. As the snapshot and the original storage's information diverges, the space allocated for the snapshot will increase.
- FULLY—The default. All storage required to fully contain a snapshot is allocated when the snapshot is created. This is the conservative method to create a snapshot, because adequate space is guaranteed for a full snapshot.

OS_UNIT_ID=

The ID that is presented to the host operating system. If set to zero, no ID will be presented to the host.

NOTE: This is a required field for OpenVMS and is recommended for use with Tru64 UNIX. It is optional for all other operating systems.

STORAGE=

The name used for this snapshot.

NOTE: You must specify the complete path from the root directory to the STORAGE and enclose the entire string in double quotes.

WORLD_WIDE_LUN_NAME=

Used to set the World Wide LUN Name on unpresented storage (either /ACTIVE or snapshot). This switch is rejected when issued against storage that is presented to a host.

Required Switch

The following switch is required with the ADD SNAPSHOT command:

- STORAGE=

Examples

```
ADD SNAPSHOT "\Virtual Disks\payroll_backup" STORAGE="\Virtual
Disks\payroll\ACTIVE" ALLOCATION_POLICY=DEMAND

ADD SNAPSHOT "\Virtual Disks\wed_night_business" STORAGE="\Virtual
Disks\daily_business\ACTIVE" ALLOCATION_POLICY=FULLY
```

The first example creates a snapshot named *payroll_backup*, from the storage named *payroll* which uses capacity only as needed. The second example creates a snapshot named *wed_night_business*, from the virtual disk named *daily_business* while setting aside all capacity necessary to create the snapshot.

NOTE: To delete virtual disks created by ADD SNAPSHOT, use the DELETE STORAGE command.

ADD STORAGE

The ADD STORAGE command creates a virtual disk with the specified name and parameters.

NOTE: This command actually creates a virtual disk family, not the virtual disk. After you create the family, an initial virtual disk named “ACTIVE” is placed as the only virtual disk in the family. This is referred to as the active virtual disk. When you later refer to this virtual disk, specify \ACTIVE after the family name. See the examples that follow the descriptions of the switches.

Syntax

```
ADD STORAGE <storage_name> SIZE=<n>
```

Where “n” is the virtual disk size in whole GBs.

Switches

The following switches are available with the ADD STORAGE command:

- COMMENT=
- GROUP=
- MIRRORED_WRITEBACK
- NOMIRRORED_WRITEBACK
- OS_UNIT_ID=
- PREFERRED_PATH=
- NOPREFERRED_PATH
- READ_CACHE
- NOREAD_CACHE
- REDUNDANCY=
- SIZE=
- WORLD_WIDE_LUN_NAME=
- WRITE_PROTECT
- NOWRITE_PROTECT

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the new storage. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

GROUP=

The disk group name where you want to create the virtual disk. The disk group must already exist to use this switch. If not specified, a controller-dependent default is used.

NOTE: You must specify the complete path from the root directory to the GROUP and enclose the entire string in double quotes.

MIRRORED_WRITEBACK

All writes are stored in mirrored cache. After the data is stored in both caches, the write is reported as complete. If one mirrored cache is unusable, the write will not be reported as complete until the data is on the media.

NOMIRRORED_WRITEBACK

No writes are stored in mirrored cache. After the data is stored in one cache, the write is reported as complete. The write will be reported as complete even if a mirror copy of cache is not available.

OS_UNIT_ID=

The ID that is presented to the host operating system. If set to zero, no ID will be presented to the host.

NOTE: This is a required field for OpenVMS and is recommended for use with Tru64 UNIX. It is optional for all other operating systems.

PREFERRED_PATH=

The default controller path that is specified to handle all I/O for the virtual disks. If a controller fails, the path always reverts to the working controller.

- **PATH_A_BOTH**—Controller path As will failover the virtual disks to controller B if controller A fails. When the controller restarts, the virtual disks failed over will failback to the original controller. This is failover/failback mode.
- **PATH_A_FAILOVER**—Controller path As will failover the virtual disks to controller B if controller A fails. However, when the controller restarts, the virtual disks failed over will *not* failback to the original controller. This is failover only mode.

- **PATH_B_BOTH**—Controller path Bs will failover the virtual disks to controller A if controller B fails. When the controller restarts, the virtual disks failed over will failback to the original controller. This is failover/failback mode.
- **PATH_B_FAILOVER**—Controller path Bs will failover the virtual disks to controller A if controller B fails. However, when the controller restarts, the virtual disks failed over will *not* failback to the original controller. This is failover only mode.

NOPREFERRED_PATH

Allows the I/O to be handled by either controller.

READ_CACHE

Reads are satisfied from the controller's cache.

NOREAD_CACHE

Reads are always satisfied from the physical disks, not the controller's cache.

REDUNDANCY=

This is the amount of protection used in creating the virtual disk (storage). If not specified, a controller-dependent default is used (the default is Vraid0).

- **VRAID0**—No failure tolerance of data is supported.
- **VRAID1**—All data is duplicated elsewhere within the storage system. This is the highest level of storage use with the lowest amount of read/write overhead.
- **VRAID5**—All data is protected by parity. This is the lowest level of storage use while maintaining redundancy, at a cost of a higher amount of read/write overhead.

SIZE=

This is a required switch. Size of the storage to be created. You can specify size in whole gigabytes only, fractions are not allowed (1GB – 2000GB, limited by the actual amount of space available within the disk group).

WORLD_WIDE_LUN_NAME=

Used to set the World Wide LUN Name on unpresented storage (either /ACTIVE or snapshot). This switch is rejected when issued against storage that is presented to a host.

NOTE: This switch is commonly used to allow a host to point to a new version of snapshot storage by giving the new snapshot the same WWN as the old snapshot.

WRITE_PROTECT

Does not allow writing to the virtual disk for all presented LUNs.

NOWRITE_PROTECT

Allows writing to the virtual disk for all presented LUNs.

Required Switch

The following switch is required with the ADD STORAGE command:

- SIZE=

Switch Defaults

The following defaults are used for the ADD STORAGE command when nothing is specified:

- MIRRORED_WRITEBACK
- NOWRITE_PROTECT
- OS_UNIT_ID= defaults to zero
- PREFERRED_PATH= defaults to NOPREFERRED_PATH
- REDUNDANCY= defaults to Vraid0
- GROUP= default disk group

Examples

```
ADD STORAGE "\Virtual Disk\scratch" SIZE=10 REDUNDANCY=VRAID5
READ_CACHE
```

```
ADD STORAGE "\Virtual Disks\engineering\gene_research" SIZE= 2
GROUP="\Disk Groups\small_disks" MIRRORED_WRITEBACK
```

The first example creates a 10 GB virtual disk within the default disk group named *scratch* using read cache and a *Vraid5* redundancy level. The second example creates a 2 GB disk at *Vraid0* redundancy level in the *small_disks* group named *small_disks* using mirrored writeback cache.

NOTE: Remember that this name is the <family_name> and that *"\Virtual Disks\<family_name>\ACTIVE"* is the name of the virtual disk. You will need to use this full path (enclosed in double quotes) when referring to it with any other commands.

CAPTURE CONFIGURATION

The CAPTURE CONFIGURATION command queries the currently selected cell and creates an Storage System Scripting Utility script. The script can then be used to re-create the cell's configuration—including all of the groups, folders, hosts, virtual disks, and LUNs.

The CAPTURE CONFIGURATION command is a powerful command. It is the only method you can use to re-create a storage system's configuration.

Once you have a satisfactory configuration, use the CAPTURE CONFIGURATION command to create a configuration file and save it in a safe place. In the event that you need to re-create the same configuration later, you can use this saved script.

NOTE: The information within the saved script made from using the CAPTURE CONFIGURATION command is only the cell's configuration. The data stored on the virtual disks is not saved. There is no guarantee that the exact same physical disks will belong to the original disk group—only the count or capacity is equivalent.

Storage reconfigured from a saved configuration is normal storage and copies only. Snapshots cannot be recreated.

Syntax

```
CAPTURE CONFIGURATION <file_name>
```

NOTE: The generated script displays on the console unless a file is specified. You can copy and paste the console display into a text editor.

If a file is specified, the generated script is saved to this file. When saving to a file, periods (.) are printed on the console to show progress. Depending on the size and configuration, this command may take a long time to complete.

An extension is not required, although you can choose one appropriate to your environment.

CAUTION: Do not reconfigure the selected cell while the CAPTURE CONFIGURATION command is executing. Changing the configuration during the configuration capture may result in a script that does not work as expected.

You can, however, continue to read and write to the existing virtual disks.

CAPTURE CONFIGURATION makes a number of assumptions when generating the reconfiguration script. If any of these assumptions are violated, the generated script may fail to reconfigure the system correctly. However, all of the assumptions involve group size and, if violated, you can easily edit the standard text file saved by the CAPTURE CONFIGURATION command to correct any "mistakes" made.

The following assumptions are made when using the CAPTURE CONFIGURATION command:

- All disks in the cell are the same size
- The disks in the new configuration are the same size and at least the same quantity as the disks in the configuration that was captured

If either of these assumptions is incorrect, simply edit the configuration script to correct the problem.

To use a saved script for re-creating the storage configuration, you must first select a manager and an uninitialized cell. Then, execute the script generated by the CAPTURE CONFIGURATION command.

Examples

```
CAPTURE CONFIGURATION c:\reconfigure_sales_cell.txt

SELECT MANAGER sf_site USERNAME=XXX PASSWORD=XXX
SELECT CELL "uninitialized storage system1"
FILE c:\reconfigure_sales_cell.txt
```

The first example shows a configuration being captured and the script saved to the file *reconfigure_sales_cell.txt.*

The second example uses the saved script to re-create the configuration. First the manager and cell are selected, and then the script is run from the saved file to re-create the storage system configuration.

DELETE

The DELETE commands are used to remove cells, folders, groups, storage, hosts, and LUNs from the storage system. The behavior of all DELETE commands can be controlled by the SET OPTIONS command (see "SET OPTIONS" on page 2–42 for details on how to use the SAFE_DELETE and NOSAFE_DELETE switches).

NOTE: All of the DELETE commands require the full path from the root directory to the object you want to delete.

DELETE CELL

The DELETE CELL command permanently removes the cell from the storage system. The cell will no longer be accessible, and all data on it is lost.

CAUTION: The DELETE CELL command removes the entire storage configuration. All customer data on the entire storage cell will be lost upon execution of this command. In addition all information about LUNs and HOSTs will be lost as well. Be sure you can afford to lose all data on the storage system if you use this command.

Syntax

```
DELETE CELL <cell_name>
```

Switch

There are no switches available with the DELETE CELL command.

Example

```
DELETE CELL payroll
```

This examples deletes the storage system (cell) named *payroll*.

DELETE FOLDER

The DELETE FOLDER command deletes the specified folder. If you wish to delete multiple layers of folders, you must do so sequentially.

IMPORTANT: Ensure folders are empty before deleting them, as the DELETE FOLDER command is rejected if there are contents inside the folder.

Syntax

```
DELETE FOLDER <folder_name>
```

Switch

There are no switches available with the DELETE FOLDER command.

Examples

```
DELETE FOLDER \Host\human_resources

DELETE FOLDER \Hosts\colorado\colorado_springs\engineering
```

The first example deletes a folder named *human_resources* within default Hosts folder. The second example deletes the folder *engineering* nested in the path *\Hosts\colorado\colorado_springs.*

DELETE GROUP

The DELETE GROUP command deletes a disk group from the controller. The command is rejected if any virtual disks (storage) are present in the disk group.

Syntax

```
DELETE GROUP <group_name>
```

Switch

There are no switches available with the DELETE GROUP command.

Example

```
DELETE GROUP "\Disk Groups\human_resources"
```

This example deletes a disk group named *human_resources*. Note the need for the double quotes because of the space in the Disk Groups folder name.

DELETE HOST

The DELETE HOST command removes the specified host from the list of hosts that connect to the Enterprise Storage System. The command is rejected if any virtual disks (storage) are presented to the specified host.

Syntax

```
DELETE HOST <host_name>
```

Switch

There are no switches available with the DELETE HOST command.

Example

```
DELETE HOST \Hosts\development
```

This example deletes the host named *development*.

DELETE LUN

The DELETE LUN command disables access to a virtual disk through the LUN from the specified host.

Syntax

```
DELETE LUN <LUN-number>
```

Switch

There are no switches available with the DELETE LUN command.

Example

```
DELETE LUN "\Hosts\accounting department\12"
```

This example removes access to LUN 12 from the host named *accounting_department*.

DELETE STORAGE

The DELETE STORAGE command destroys the specified virtual disk (storage). The virtual disk will no longer be accessible, and all data on it is lost.

CAUTION: The DELETE STORAGE command removes the entire storage (virtual disk) from the storage system. All customer data on the storage specified will be lost upon execution of this command. In addition all information about the LUNs presented form this storage (virtual disk) will be lost as well. Be sure you can afford to lose all data on the storage specified if you use this command.

Syntax

```
DELETE STORAGE <storage_name>
```

Switch

There are no switches available with the DELETE STORAGE command.

Examples

```
DELETE STORAGE "\Virtual Disks\d12\ACTIVE"

DELETE STORAGE "\Virtual Disks\user_disk\ACTIVE"
```

The first example deletes the virtual disk named *d12*. The second example deletes the virtual disk *user_disk*.

EXIT

The EXIT command terminates the SSSU session.

If Storage System Scripting Utility is accepting input from the terminal or the command line, EXIT causes the program to terminate.

NOTE: Any commands on the command line after an EXIT command are not processed.

If Storage System Scripting Utility is processing a script from a file due to encountering a FILE command, the behavior of the EXIT command depends on how the FILE command was issued:

- If the FILE command was issued from the command line, an EXIT command in the file causes SSSU to terminate.
- If the FILE command was issued from the terminal, an EXIT command in the file causes SSSU to return to accepting input from the terminal.
- If the FILE command was issued from a file that is being executed from a previous FILE command, an EXIT command in the file causes control to return to the previous file being executed. The execution continues with the command following the FILE command that started the execution of the file.

Commands that are found in a file after the EXIT command are not executed. EXIT causes an immediate return to the “calling” layer.

Syntax

```
EXIT
```

Example

```
EXIT
```

SSSU terminates.

FILE

The FILE command causes the current mode of input to suspend, and redirects the scripting utility to accept input from the specified file. Either the end of the file or an EXIT command in the specified file causes SSSU to again accept input from the previous input source.

FILE commands may be nested. Nesting FILE commands means that a file being executed through a FILE command can have FILE commands within its command set. The only limitation on how deep FILE commands may be nested is based on the host system's resources.

Syntax

```
FILE <filename.ext>
```

Where <ext> stands for extension. An extension is not required by Storage System Scripting Utility. Although you can choose one appropriate to your environment.

Examples

```
FILE snapd1.txt

FILE d:\scripts\snapshots\d27.txt
```

The first example executes the file named *snapd1.txt* from the present directory. The second example executes a file named *d27.txt* located in the specified directory.

HELP

The HELP command displays information on using the built-in help.

Syntax

```
HELP
```

Example

```
HELP

Command syntax help is available at any point within a command by
typing a '?'

For example:

Yekao> ADD STORAGE user ?

The options are:

   COMMENT
   GROUP
   MIRRORED_WRITEBACK
   NOMIRRORED_WRITEBACK
   OS_UNIT_ID
   PREFERRED_PATH
   NOPREFERRED_PATH
   READ_CACHE
   NOREAD_CACHE
   REDUNDANCY
   SIZE
   WORLD_WIDE_LUN_NAME
   WRITE_PROTECT
   NOWRITE_PROTECT

Arguments that require keywords may also be queried, for example:

Yekao> ADD STORAGE Enicar REDUNDANCY=?

The options are:

   VRAID0
   VRAID1
   VRAID5
```

As shown above, you can type a question mark wherever you would normally include a parameter, at any level of a command line. The help system lists the options available for that parameter.

RESTART

The RESTART command causes any power object shown by the SHOW POWER command to restart.

A power object is an object that you can control in terms of powering off and on. An example is the HSV110 controller.

Syntax

```
RESTART <power_object_name>
```

Switches

The following switches are available for the RESTART command:

- ALL_PEERS
- NOALL_PEERS

These switches are described in the following paragraphs.

ALL_PEERS

All peer power objects (both HSV110 controllers) on this cell will be restarted together.

NOALL_PEERS

Only the one power object (the specified HSV110 controller) specified will be restarted (this is the default).

Examples

```
RESTART "\Hardware\Rack 1\Controller Enclosure 7\Controller B"
ALL_PEERS
```

```
RESTART "\Hardware\Rack 1\Controller Enclosure 7\Controller A"
NOALL_PEERS
```

The first example restarts *Controller B,* including its peer power objects. The second example restarts the *Controller A,* but not its peer power objects.

SELECT

The SELECT command selects a cell or management appliance. The prompt changes to reflect the selected cell or management appliance.

SELECT CELL

The SELECT CELL command directs the command prompt to the selected cell. All configuration commands then affect the selected cell. If the cell name has spaces in it, the name must be surrounded by double quotes (“”).

NOTE: After selecting a cell, the prompt name changes to reflect the selected cell.

Syntax

```
SELECT CELL <cell_name>
```

Examples

```
SELECT CELL employees

SELECT CELL "payroll storage"
```

The first example selects the cell named *employees* and returns the prompt `employees>`. The second example selects a cell named *payroll storage* and returns the prompt `payroll storage>`.

SELECT MANAGER

The SELECT MANAGER command directs the command prompt to the selected element manager. All configuration commands then affect the selected element manager. If the element manager name has spaces in it, the name must be surrounded by double quotes (“”).

Syntax

```
SELECT MANAGER <manager_name> USERNAME=<username>
PASSWORD=<password>
```

Switches

The following switches are available with the SELECT MANAGER command:

- PASSWORD=
- USER=

These switches are described in the following paragraphs.

PASSWORD=

Password associated with the username to access the element manager.

USER=

Username to access the element manager.

Required Switches

The following switches are required with the SELECT MANAGER command:

- PASSWORD=
- USER=

Examples

```
SELECT MANAGER north_campus USERNAME=XXX PASSWORD=XXXX
```

```
SELECT MANAGER "secure area" USERNAME=XXX PASSWORD=XXXX
```

The first example selects the manager named *north_campus*. The second example selects the manager named *secure area*.

The username and password for the manager are validated on a subsequent command.

SET

The SET commands allow you to rename cells, disks, folders, groups, hosts, and the monitor. It also allows you to reset existing storage settings and application options.

NOTE: All of the SET commands require the full path from the root directory to the object you want to rename.

SET CELL

The SET CELL command changes the specified cell's name.

Syntax

```
SET CELL <name>
```

Switches

The following switches are available for the SET CELL command:

- COMMENT=
- CONSOLE_LUN_ID=
- NAME=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the cell. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

CONSOLE_LUN_ID=

The LUN used for console communication. If set to zero, a console LUN is not presented to the host.

NOTE: This value is used for IBM AIX (set to zero), OpenVMS (required), and Tru64 UNIX (recommended). Other host operating systems will ignore the value. See the host operating system installation guides for more information.

NAME=

New name to give to the specified cell.

Example

```
SET CELL engineering NAME=accounting
```

This example renames the cell named *engineering* to *accounting*.

SET DISK

The SET DISK command changes the specified disk's name.

Syntax

```
SET DISK <name>
```

Switches

The following switches are available for the SET DISK command:

- COMMENT=
- NAME=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the disk. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

NAME=

New name to give to the specified disk.

Example

```
SET DISK "\Disk Groups\Ungrouped Disks\Disk 005" NAME=5Disk
```

This example renames the disk named *Disk 005* to *5Disk*.

SET FOLDER

The SET FOLDER command changes the specified folder's name.

Syntax

```
SET FOLDER <name>
```

Switches

The following switches are available for the SET FOLDER command:

- COMMENT=
- NAME=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the folder. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

NAME=

New name to give to the specified folder.

Example

```
SET FOLDER "\Virtual Disks\top_secret" NAME=everyone_knows
```

This example renames the folder named *top_secret* to *everyone_knows*.

SET GROUP

The SET GROUP command changes the specified disk group's name.

Syntax

```
SET GROUP <name>
```

Switches

The following switches are available with the SET GROUP command:

- ADD=
- COMMENT=
- DELETE=
- NAME=
- OCCUPANCY_ALARM=

These switches are described in the following paragraphs.

ADD=

The number of disk drives to add to the specified disk group.

COMMENT=

This is an optional switch that associates a user-defined comment with the disk group. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

DELETE=

The name of the disk drive to be removed from the disk group.

NAME=

New name to give to the specified disk group.

OCCUPANCY_ALARM=

The point when a defined percentage of space is used. When this point is reached, an event log is generated and sent to the Management Appliance (optionally the host system) informing the administrator that the group is becoming full. Do not use the percent sign (%) after the number.

Example

```
SET GROUP "\Disk Groups\pool" NAME="\Disk Groups\nuclear_secrets"
OCCUPANCY_ALARM=75
```

This example renames the disk group named *pool* to *nuclear_secrets* and sends an event log to the Management Appliance when 75% capacity is reached.

SET HOST

The SET HOST command is used to add or delete World Wide Names, set the operating systems for a host, and add comments to a specified host.

NOTE: Use the SET HOST command to add additional FCAs to hosts added to the storage system with the ADD HOST command.

Syntax

```
SET HOST <name>
```

Switches

The following switches are available with the SET HOST command.

- ADD_WORLD_WIDE_NAME=
- DELETE_WORLD_WIDE_NAME=
- COMMENT=
- OPERATING_SYSTEM=

These switches are described in the following paragraphs.

ADD_WORLD_WIDE_NAME=

Adds the World Wide Name of the host port.

DELETE_WORLD_WIDE_NAME=

Deletes the World Wide Name of the host port.

COMMENT=

This is an optional switch that associates a user-defined comment with the host. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

OPERATING_SYSTEM=

The operating system type of the specified host. The operating system may be one of the following (they are spelled here as SSSU expects):

- HPUX
- IBMAIX

- OPEN_VMS
- SOLARIS
- TRU64
- WINDOWS

If unspecified, the operating system type for the host defaults to Windows.

Examples

```
SET HOST \Hosts\install OPERATING_SYSTEM=OPEN_VMS

SET HOST \Hosts\install ADD_WORLD_WIDE_NAME=1000-0000-C922-36CA
```

The first example assigns OpenVMS as the operating system type for the host named *install*. The second example adds a WWN (FCA) to the host named *install*.

SET MONITOR

The SET MONITOR command changes the specified monitor's name.

Syntax

```
SET MONITOR <name>
```

Switches

The following switches are available with the SET MONITOR command:

- COMMENT=
- NAME=

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the monitor. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

NAME=

New name to give to the specified monitor.

Example

```
SET MONITOR "\Hardware\Rack 1\Controller Enclosure 7\Controller A"
COMMENT="cabinet_near_the_door"
```

This example adds the comment *cabinet_near_the_door* to the monitor named *Controller A*.

NOTE: Remember to enclose in double quotes any paths or object names that include spaces.

SET OPTIONS

The SET OPTIONS command allows you to change the characteristics of the Storage System Scripting Utility.

NOTE: The options you set with the SET OPTIONS command are in effect for the current session only. Each time you start SSSU, the default options are reinstated.

Syntax

```
SET OPTIONS
```

Switches

The following switches are available with the SET OPTIONS command:

- COMMAND_DELAY=
- NOCOMMAND_DELAY
- DISPLAY_WIDTH=
- ON_ERROR=
  - CONTINUE
  - EXIT_ON_ERROR
  - HALT_ON_ERROR
- RETRIES=
- NORETRIES
- SAFE_DELETE
- NOSAFE_DELETE

These switches are described in the following paragraphs.

COMMAND_DELAY=

The number of seconds (0 - 300 seconds) to wait between issuing commands when running a script using the FILE command. The default is set to 10 seconds. This has no effect when you are typing commands.

NOCOMMAND_DELAY

Specifies no wait time between commands issued from within a FILE command.

DISPLAY_WIDTH=

Sets the amount of characters displayed on a line for SHOW commands. The default is 80. If the Storage System Scripting Utility output is to be parsed, it is useful to set a high line width. A long line keeps the lines from wrapping, making the output easier to cut, paste, and parse. Line widths from 70 to 500 may be specified.

ON_ERROR=

- **CONTINUE**—This is the default. In this mode, only the EXIT command will cause the scripting utility to halt. This is the recommended mode when manually typing commands.
- **EXIT_ON_ERROR**—Any error will cause the scripting utility to exit with an error code. This mode is useful when requesting the entire script to halt immediately if errors occur while executing a script. This causes the scripting utility to halt on *any* kind of error—failed command, syntax error, or ambiguous command.
- **HALT_ON_ERROR**—This is similar to EXIT_ON_ERROR. Any error causes the scripting utility to halt, but not exit until any key is pressed. Then the scripting utility exits with an error code. This allows you to see the error before the window closes on exit.

NOTE: When manually typing commands, Compaq recommends not using the EXIT_ON_ERROR or HALT_ON_ERROR modes. This is because the scripting utility exits on *any* kind of error, including a typographical one.

RETRIES=

Number of minutes to attempt "long" period retries—such as when the HSV Element Manager service is either busy or restarting. Specify in the range of 1–10 minutes (inclusive). The default is four minutes.

NORETRIES

Specify that you do not want the scripting utility to retry commands.

SAFE_DELETE

SAFE_DELETE, which is the default, requires the user to delete all overlying objects before the object specified can be deleted. For example, if you specified a deletion of storage which had LUNs presented, the DELETE STORAGE command would be rejected with an appropriate message saying that all the LUNs presented from this storage must be deleted before the storage may be deleted.

Deletion of overlaying objects is always required if SAFE_DELETE is specified.

NOSAFE_DELETE

NOSAFE_DELETE allows the deletion of an object even if overlying objects are present. The behavior of the NOSAFE_DELETE command depends on the mode that SSSU is running in.

If the input is from the terminal (a user typing in the commands), an appropriate warning is displayed along with the question: “Are you sure?” The user must respond “Yes” for the deletion to proceed. Note that the user must respond with a capital “Y” followed by lowercase “es” in order for the delete to proceed.

If the input is from a script (using the FILE command) or the command line, the deletion proceeds without prompting with a question.

CAUTION: Use of the NOSAFE_DELETE switch may cause loss of data.

Examples

```
SET OPTION ON_ERROR=HALT_ON_ERROR NORETRIES

SET OPTION ON_ERROR=NOHALT_ON_ERROR NOSAFE_DELETE DISPLAY_WIDTH=132
RETRIES=6
```

The first example sets the options for SSSU to stop upon encountering an error and not to retry the commands. The second example sets the display option to 132, not to stop on error, to allow deletion of overlaying objects, and to retry for 6 minutes.

SET STORAGE

The SET STORAGE command allows changes to the properties of specific virtual disks (storage).

Syntax

```
SET STORAGE <storage_name>
```

Switches

The following switches are available for the SET STORAGE command:

- COMMENT=
- MIRRORED_WRITEBACK
- NOMIRRORED_WRITEBACK
- NAME=
- OS_UNIT_ID=
- PREFERRED_PATH=
- NOPREFERRED_PATH
- READ_CACHE
- NOREAD_CACHE
- SIZE=
- WORLD_WIDE_LUN_NAME=
- WRITE_PROTECT
- NOWRITE_PROTECT

These switches are described in the following paragraphs.

COMMENT=

This is an optional switch that associates a user-defined comment with the storage. Maximum of 128 characters are allowed and the string must be enclosed in double quotes.

MIRRORED_WRITEBACK

All writes are stored in mirrored cache. After the data is stored in both caches, the write is reported as complete. If one mirrored cache is unusable, the write is not reported as complete until the data is on the media.

NOMIRRORED_WRITEBACK

No writes are stored in mirrored cache. After the data is stored in one cache, the write is reported as complete. The write is reported as complete even if a mirror copy of cache is not available.

NAME=

New name to give to the specified storage (virtual disk).

OS_UNIT_ID=

The ID that is presented to the host operating system. If set to zero, no ID will be presented to the host.

NOTE: This is a required field for OpenVMS and is recommended for use with Tru64 UNIX. It is optional for all other operating systems.

PREFERRED_PATH=

The controller path that is, as a default, specified to handle all I/O for the virtual disks (storage). If a controller fails, the path always reverts to the working controller.

- **PATH_A_BOTH**—Controller path As will failover the virtual disks to controller B if controller A fails. When the controller restarts, the virtual disks failed over will failback to the original controller. This is failover/failback mode.
- **PATH_A_FAILOVER**—Controller path As will failover the virtual disks to controller B if controller A fails. However, when the controller restarts, the virtual disks failed over will *not* failback to the original controller. This is failover only mode.
- **PATH_B_BOTH**—Controller path Bs will failover the virtual disks to controller A if controller B fails. When the controller restarts, the virtual disks failed over will failback to the original controller. This is failover/failback mode.
- **PATH_B_FAILOVER**—Controller path Bs will failover the virtual disks to controller A if controller B fails. However, when the controller restarts, the virtual disks failed over will *not* failback to the original controller. This is failover only mode.

NOPREFERRED_PATH

Allows the I/O to be handled by any controller.

READ_CACHE

Reads are satisfied from the controller's cache.

NOREAD_CACHE

Reads are always satisfied from media, not the controller's cache.

SIZE=

This is a required switch. New size of the storage in gigabytes. You can specify size in whole gigabytes only, fractions are not allowed (1GB – 2000GB, limited by the actual amount of space available within the disk group).

WORLD_WIDE_LUN_NAME=

Used to set the World Wide LUN Name on unpresented storage (either /ACTIVE or snapshot). This switch is rejected when issued against storage that is presented to a host.

NOTE: This switch is commonly used to allow a host to point to a new version of snapshot storage by giving the new snapshot the same WWN as the old snapshot.

WRITE_PROTECT

Does not allow writing to the virtual disk for all presented LUNs.

NOWRITE_PROTECT

Allows writing to the virtual disk for all presented LUNs.

Example

```
SET STORAGE "\Virtual Disks\archive\ACTIVE" WRITE_PROTECT
WORLD_WIDE_LUN_NAME=6000-1fe1-ff00-0000
```

The example creates a write-protected virtual disk named *archive* and assigns the specified World Wide LUN Name.

SHOW

The SHOW commands display information about various elements in the currently selected storage system (cell).

NOTE: All of the SHOW commands require the full path from the root directory to the object you want to display.

SHOW CELL

The SHOW CELL command displays the cells currently managed by this element manager, or, if <*cell_name*> is specified, detailed information about that specified cell.

Syntax

```
SHOW CELL <cell_name>
```

Switch

Only the FULL switch is available with the SHOW CELL command.

FULL

FULL is used in the absence of a <*cell_name*> and displays detailed information about all cells currently selected through the SELECT MANAGER command.

Examples

```
SHOW CELL

SHOW CELL payroll
```

The first example displays a list of available cells (storage systems) on this manager. The second example displays specific information about the cell named *payroll*.

SHOW DISK

The SHOW DISK command displays disk configuration information for physical disks connected to the currently selected cell.

Syntax

```
SHOW DISK <disk_name>
```

Switch

Only the FULL switch is available for the SHOW DISK command.

FULL

FULL is used in the absence of a <*disk_name*> and displays detailed information about all the physical disks currently selected through the SELECT CELL command.

Examples

```
SHOW DISK
```

```
SHOW DISK "\Disk Groups\Ungrouped Disks\Disk 039"
```

The first example displays a list of all the physical disks in the currently selected cell. The second example displays specific information about the physical disk 039.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW GROUP

The SHOW GROUP command displays the disk groups currently configured by this cell, or, if <*group_name*> is specified, detailed information about the specified disk group.

Syntax

```
SHOW GROUP <group_name>
```

Switch

Only the FULL switch is available with the SHOW GROUP command.

FULL

FULL is used in the absence of a <*group_name*> and displays detailed information about all disk groups currently selected through the SELECT CELL command.

Examples

```
SHOW GROUP

SHOW GROUP "\Disk Groups\Default Disk Group"
```

The first example displays a list of the available groups in the currently selected cell. The second example displays information about the default disk group.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW HOST

The SHOW HOST command displays the hosts currently supported by this cell, or, if <*host_name*> is specified, information about the specified host.

Syntax

```
SHOW HOST <host_name>
```

Switch

Only the FULL switch is available with the SHOW HOST command.

FULL

FULL is used in the absence of a <*host_name*> and displays detailed information about all hosts currently selected through the SELECT CELL command.

Examples

```
SHOW HOST

SHOW HOST \Hosts\med_lab
```

The first example displays the list of available hosts on the currently selected cell. The second example displays specific information about the host named *med_lab*.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW LUN

The SHOW LUN command displays the LUNs currently configured on this cell.

Syntax

```
SHOW LUN <LUN_number>
```

Switch

Only the FULL switch is available with the SHOW LUN command.

FULL

FULL is used in the absence of a specific LUN and displays detailed information about all LUNs currently selected through the SELECT CELL command.

Examples

```
SHOW LUN

SHOW LUN \Hosts\SVT165\13
```

The first example displays the list of available LUNs on the selected cell. The second example displays specific information about the LUN named *\Hosts\SVT165\13*.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW MANAGER

The SHOW MANAGER command displays the managers currently available, or, if <*manager_name*> is specified, detailed information about that specific element manager.

Syntax

```
SHOW MANAGER <manager_name>
```

Examples

```
SHOW MANAGER

SHOW MANAGER cxo_campus
```

The first example displays all the available managers in the selected cell. The second example displays information about the specific manager named *cxo_campus*.

NOTE: Remember to enclose in double quotes any names that contain spaces.

SHOW MONITOR

The SHOW MONITOR command displays the monitors currently installed on this cell, or, if <*monitor_name*> is specified, detailed information about that specified monitor.

Syntax

```
SHOW MONITOR <monitor_name>
```

Switch

Only the FULL switch is available for the SHOW MONITOR command.

FULL

FULL is used in the absence of a <*monitor_name*> and displays detailed information about all monitors currently selected through the SELECT CELL command.

Examples

```
SHOW MONITOR

SHOW MONITOR "\Hardware\Rack 1\blue_cab"
```

The first example displays information about all of the monitors available on the currently selected cell. The second example displays information about the monitor named *blue_cab*.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW OPTIONS

The SHOW OPTIONS command displays the current Storage System Scripting Utility options.

Syntax

```
SHOW OPTIONS
```

Example

```
SHOW OPTIONS
Options:
    NOCOMMAND_DELAY
    DISPLAY_WIDTH = 80
    ON_ERROR = CONTINUE
    RETRIES = 4 (minutes)
    SAFE_DELETE
```

The example shows the current options set for Storage System Scripting Utility.

SHOW POWER

The SHOW POWER command displays the controllers currently configured on this cell, or, if <*controller_name*> is specified, detailed information about that specific controller.

Syntax

```
SHOW POWER <controller_name>
```

Switch

Only the FULL switch is available with the SHOW POWER command.

FULL

FULL is used in the absence of a <*controller_name*> and displays detailed information about all objects on which you can control power currently selected through the SELECT CELL command.

Examples

```
SHOW POWER

SHOW POWER "\Hardware\Rack 1\Controller Enclosure 7\Controller A"
```

The first example will display a list of all power objects within the selected cell. The second example displays specific information about *Controller A*.

NOTE: Remember to include full paths for objects. If there are spaces within paths or object names, enclose the entire string in double quotes.

SHOW STORAGE

The SHOW STORAGE command displays the storage currently configured on this cell, or, if <*storage_name*> is specified, detailed information about the specified storage.

Syntax

```
SHOW STORAGE <storage_name>
```

Switch

Only the FULL switch is available with the SHOW STORAGE command.

FULL

FULL is used in the absence of a <*storage_name*> and displays detailed information about all virtual disks (storage) currently selected through the SELECT CELL command.

Examples

```
SHOW STORAGE

SHOW STORAGE "\Virtual Disks\scratch_disk\ACTIVE"
```

The first example displays information about all virtual disks (storage) in the currently selected cell. The second example displays information about the specific active virtual disk named *scratch_disk\ACTIVE*.

NOTE: Remember to include full paths for objects and enclose in double quotes any paths that contain spaces.

SHOW WORLD_WIDE_NAME

The SHOW WORLD_WIDE_NAME command displays the host World Wide Names (WWNs) that are visible to this cell and not already assigned to a host.

Syntax

```
SHOW WORLD_WIDE_NAME
```

Example

```
SHOW WORLD_WIDE_NAME
```

This example displays all of the WWNs visible to this cell which have not been assigned to a host.

SHUTDOWN

The SHUTDOWN command causes any object displayed by the SHOW POWER command to shutdown.

A power object is an object that you can control in terms of powering off and on. An example is the HSV110 controller.

Syntax

```
SHUTDOWN <power_object_name>
```

Switches

The following switches are available with the SHUTDOWN command.

- ALL_PEERS
- NOALL_PEERS

These switches are described in the following paragraphs.

ALL_PEERS

All peer power objects (for example both HSV110 controllers) on this cell will be shut down together.

NOALL_PEERS

Only the specified power object (for example, an HSV110 controller) will be shut down as a result of this command (this is the default).

Examples

```
SHUTDOWN "\Hardware\Rack 1\Controller Enclosure 7\Controller A"
NOALL_PEERS
```

```
SHUTDOWN "\Hardware\Rack 1\Controller Enclosure 7\Controller B"
ALL_PEERS
```

The first example shuts down *Controller A*, leaving any peers alone. The second example shuts down *Controller B* and all associated peers.

A

Configuration Examples

This appendix provides examples of how to use commands to create a simple configuration and to view specific system information. The following topics are covered:

Creating a Simple Configuration

This section presents a simple configuration example in two ways:

- Annotated with steps (page A–2 through page A–5)
- Configuration as it appears after executing all of the commands (page A–5)

Creating the Storage System

- Log on to the manager for your Enterprise Virtual Array (SELECT MANAGER)
- View the cells available on the manager (Uninitialized Storage System1, if this is an uninitialized storage system), using the SHOW CELL and SELECT CELL commands
- Add a cell (create the storage system by initializing)

1. Start Storage System Scripting Utility from anywhere on the same IP network as your Enterprise Virtual Array.

   The `NoCellSelected>` prompt displays.

2. Enter the SELECT MANAGER command with the name of the manager and the logon information (a sample username, password, and manager are used in the example below).

```
NoCellSelected> SELECT MANAGER swma31k008 USERNAME=XXX PASSWORD=XXX
NoCellSelected>
```

NOTE: Remember that this username and password must be set through the HSV Element Manager software—not in Storage System Scripting Utility.

3. At the prompt, enter the SHOW CELL command to display all the available cells on this manager.

```
NoCellSelected> SHOW CELL
Cells available on this Manager:
   Uninitialized Storage System1
NoCellSelected>
```

   The cells that are available on this manager display. The default for an uninitialized storage system is one cell named *Uninitialized Storage System1*, as in the example shown.

4. Select the Uninitialized Storage System1 cell with the SELECT CELL command.

```
NoCellSelected> SELECT CELL "Uninitialized Storage System1"
Uninitialized Storage System1>
```

The prompt changes to reflect the selected cell (*Uninitialized Storage System1*).

5. Initialize the storage system with the ADD CELL command, providing a name for the cell (such as "Yekao" below). This is the equivalent of creating one cell (one storage system), using all of the available physical disk space.

```
Uninitialized Storage System1> ADD CELL Yekao
NoCellSelected>
```

NOTE: The prompt returns to `NoCellSelected>`. This is because the cell's name has changed. You will need to re-select it with the name you just gave it (SELECT CELL Yekao).

Attempting to add a cell (create storage system) without the license, will return an error message. This message indicates that you need to enter your licensing information in the HSV Element Manager. That is, you must have a license and follow the registration process.

NOTE: The example here defaults to putting all of the available physical disks in the default disk group because no DEVICE_COUNT switch was used.

Creating and Presenting a Virtual Disk

To create a virtual disk and present it to a host, you:

- Select the newly created cell (storage system)
- Add a virtual disk (ADD STORAGE) to the storage system
- Add a host to the storage system
- Add a LUN as presentation to the host from the specified STORAGE

These steps are described below.

1. Use the SELECT CELL command to select the cell you just created.

```
NoCellSelected> SELECT CELL Yekao
Yekao>
```

The prompt changes to reflect the object selected. In this case, a cell (storage system) named *Yekao*.

2. Create a virtual disk using the ADD STORAGE command, supplying at the very least the storage (virtual disk) name and size in whole GBs.

```
Yekao> ADD STORAGE Enicar SIZE=4
Yekao>
```

In the example, a *4* GB virtual disk named *Enicar* is created with a Vraid0 redundancy.

IMPORTANT: The virtual disk (storage) can be between 1 GB (minimum) and 2000GBs (maximum). The increments must be in whole GBs.

To present this newly created virtual disk, you need to have at least one host server within the storage system. This host must be added to the Enterprise Virtual Array.

You can use the SHOW WORLD_WIDE_NAME command to see the available, unmapped FCA WWNs.

3. Use the ADD HOST command to add a host to the storage system, supplying the World Wide Name (WWN) for the host port (FCA).

   Use caution with WWN assignments as these are not verified when assigned.

In this example, only one port is added, but because the Enterprise Virtual Array requires multipathing, you need to have at least two WWNs for each host. Therefore, additional WWNs are added with the SET HOST command.

```
Yekao> ADD HOST \Hosts\SVT165 WORLD_WIDE_NAME=1000-0000-c923-6735
Yekao> SET HOST  \Hosts\SVT165
ADD_WORLD_WIDE_NAME=1000-0000-c923-6736
```

4. Add the LUN to the host. Use the ADD LUN command, supplying the full path to the virtual disk and the host machine.

   ```
   Yekao> ADD LUN 13 STORAGE="\Virtual Disks\Enicar\ACTIVE"
   HOST=\Hosts\SVT165
   ```

   Remember, for any path or name that contains spaces, you must enclose the entire string in double quotes, as in the example above.

   Now LUN *13,* which is *4* GBs in size, is presented to the host named *SVT165*.

Sample Configuration Command Output

The following example shows the commands that were entered during the previous procedures (page A–2 through page A–5), without the interruption of the instructions.

```
NoCellSelected> SELECT MANAGER swma31k008 USERNAME=XXX PASSWORD=XXX
NoCellSelected> SHOW CELL
Cells available on this Manager:
   Uninitialized Storage System1
NoCellSelected> SELECT CELL "Uninitialized Storage System1"
Uninitialized Storage System1> ADD CELL Yekao
NoCellSelected> SELECT CELL Yekao
Yekao> ADD STORAGE Enicar SIZE=4
Yekao> ADD HOST SVT165 WORLD_WIDE_NAME=1000-0000-c923-6735
Yekao> SET HOST  \Hosts\SVT165
ADD_WORLD_WIDE_NAME=1000-0000-c923-6736
Yekao> ADD LUN 13 STORAGE="\Virtual Disks\Enicar\ACTIVE"
HOST=\Hosts\SVT165
```

Using the SHOW Commands

This section provides example uses of the SHOW command, based on the storage system, host, and virtual disk created in the previous section, “Creating a Simple Configuration” on page A–2. The purpose of this presentation is to provide a sample of Storage System Scripting Utility output.

As shown below, to view information about various cell attributes, you first select the cell and then use the SHOW commands.

SHOW CELL

Use the SHOW CELL command to view information about the selected cell (storage system).

1. At the `NoCellSelected>` prompt, type SHOW CELL.

```
NoCellSelected> SHOW CELL
Cells available on this Manager:
    Yekao
NoCellSelected>
```

   The available cells on this manager are displayed.

2. Select the cell you want with the SELECT command.

```
NoCellSelected>SELECT CELL Yekao
```

3. Use the SHOW CELL command to display information about the cell.

```
Yekao> SHOW CELL Yekao

Yekao information:
 Initialized Enterprise System Properties:
                 Name : Yekao
      Operational_State : Initialized - Good
    Total_storage_space : 2977.41 GB
     Storage_space_used : 0.00 GB
Available_storage_space : 2977.41 GB
 Device_addition_policy : Manual
        Console_LUN_ID : 0
       Controller_time : 27-Jan-2002 19:54:6 GMT
  Node_world_wide_name : 5000-1FE1-0003-0310
          Comments :
                   ID : 08000710b40805600c00010000a001000000e723
```

SHOW STORAGE

Use the SHOW STORAGE command to display information about the virtual disks (storage) within the selected cell (storage system).

1. Enter the SHOW STORAGE command (with a cell selected) to display a list of available storage. In the example, there is just one virtual disk available, named *Enicar*.

```
Yekao> SHOW STORAGE

Storage available on this Cell:
   \Virtual Disks\Enicar\ACTIVE
Yekao>
```

2. Use the SHOW STORAGE command to display information about the virtual disk.

NOTE: Remember that the ADD STORAGE command creates a virtual disk family and you must include "\ACTIVE" when referring to the actual virtual disk.

```
Yekao> SHOW STORAGE "\Virtual Disks\Enicar\ACTIVE"

\Virtual Disks\Enicar\ACTIVE information:
Identification:
             Name : \Virtual Disks\Enicar\ACTIVE
      Family_Name : Enicar
    World_Wide_ID : 0001-A000-2461-0000
Attributes:
       Disk_Group : \Disk Groups\Default Disk Group
     Capacity_Req : 4 GB
    Capacity_Used : 4 GB
      Redundancy : vRaid0
 Write_cache_Policy : Mirrored write-back
  Read_cache_Policy : On
     Comm_Protocol :
    Write_Protect : No
   Enable_Disable : Enable
Condition_State:
  Operational_State : 1
Date Time:
           Created : 27-Jan-2002 19:57:37
Presentation:
        OS_Unit_ID : 0
    Preferred_path : No preference
Comments :
               ID : 00200710b40805600c00010000a0010000006124
Yekao>
```

SHOW HOST

Use the SHOW HOST command to display information about the hosts available on the storage system.

1. With the cell selected, enter SHOW HOST to display the list of available hosts. In this example, there is one host named *\Hosts\SVT165*.

```
Yekao> SHOW HOST
Hosts available on this Cell:
   \Hosts\SVT165

Yekao>
```

2. Use the SHOW HOST command, followed by the specific host name, to display information about the host.

NOTE: Remember that you need to type the full path to the host object, and that if there are any spaces within the path or host name, the entire string needs to be enclosed within double quotes.

```
Yekao> SHOW HOST \Hosts\SVT165

\Hosts\SVT165 information:
Identification:
            Name : \Hosts\SVT165
         IPAddess : Dynamic IP Assignment
          Status : Initialized - Good
  Operating_System : Microsoft Windows
Fibre_Channel_adapter_ports:
   Fibre_Channel_adapter_ports [0]:
            Port : 1000-0000-c923-6735
Fibre_Channel_adapter_ports [1]:
            Port : 1000-0000-c923-6736
Presentation:
       Comments :
             ID : 00800710b40805600c00010000a0010000006524
Yekao>
```

SHOW LUN

Use the SHOW LUN command to display information about specific LUNs.

1. With the cell selected, enter the SHOW LUN command to display the list of available LUNs. In this example, there is one LUN (13) available, with the full name of *\Hosts\SVT165\13*.

```
Yekao> SHOW LUN
LUNs available on this Cell:
   \Hosts\SVT165\13
Yekao>
```

2. Use the SHOW LUN command, with the specific LUN named, to display the specific LUN information.

```
Yekao> SHOW LUN \Hosts\SVT165\13

\Hosts\SVT165\13 information:
             Name : \Hosts\SVT165\13
      StorageName : \Virtual Disks\Enicar\ACTIVE
  StorageclientName : \Hosts\SVT165
               ID : 00880710b40805600c00010000a0010000006624
```

Glossary

This glossary defines Enterprise Virtual Array terms used in this publication or related to this product and is not a comprehensive glossary of computer terms.

active virtual disk

A virtual disk (VD) is a simulated disk drive created by the controllers as storage for one or more hosts. An active virtual disk is accessible by one or more hosts for normal storage. An active virtual disk and its snapshot, if one exists, constitute a virtual disk family. An active virtual disk is the only necessary member of a virtual disk family.

See also **virtual disk, virtual disk copy, virtual disk family,** *and* **snapshot.**

array

All the physical disk drives in a storage system that are known to and under the control of a controller pair.

array controller

See **controller.**

block

Also called a sector. The smallest collection of consecutive bytes addressable on a disk drive. In integrated storage elements, a block contains 512 bytes of data, error codes, flags, and the block address header.

cache

High-speed memory that sets aside data as an intermediate data buffer between a host and the storage media. The purpose of cache is to improve performance.

See also **read cache, write cache,** *and* **mirrored cache.**

cell

SSSU term for the storage system.

communication logical unit number (LUN)

See **console LUN.**

console LUN

A SCSI-3 virtual object that makes a controller pair accessible by the host before any virtual disks are created. *Also* called a communication LUN.

console LUN ID

The ID that can be assigned when a host operating system requires a unique ID. The console LUN ID is assigned by the user, usually when the storage system is initialized.

See also **console LUN.**

controller

A hardware/firmware device that manages communications between host systems and other devices. Controllers typically differ by the type of interface to the host and provide functions beyond those the devices support.

controller pair

Two interconnected controller modules which together control a physical disk array. A controller pair and the disk array together constitute a storage system.

default disk group

The first disk group created at the time the system is initialized. The default disk group can contain the entire set of physical disks in the array or just a few of the disks.

See also **disk group.**

disk drive

A carrier-mounted storage device supporting random access to fixed size blocks of data.

disk failure protection

A method by which a controller pair reserves drive capacity to take over the functionality of a failed or failing physical disk.

disk group

A physical disk drive set or pool in which a virtual disk is created. A disk group may contain all the physical disk drives in a controller pair array or a subset of the array.

Element Manager

The graphical user interface through which a user controls and monitors a storage system. The HSV Element Manager software can be installed on more than one management appliance in a fabric. Each installation of the HSV Element Manager software is a management agent.

EMU

Environmental Monitoring Unit. SSSU refers to the EMU as a monitor. An element which monitors the status of an enclosure, including the power, air temperature, and blower status. The EMU detects problems and displays and reports these conditions to a user and the controller. In some cases, the EMU implements corrective action.

Enterprise Virtual Array

The Compaq name used to describe the storage system that includes HSV controllers, storage devices, enclosures, cables, and power supplies. *Also* called the Enterprise Storage System.

environmental monitoring unit

See **EMU.**

event log

System information sent to the Management Appliance or host operating system.

fabric

A Fibre Channel fabric switch or two or more interconnected Fibre Channel switches allowing data transmission.

fabric port

A port which is capable of supporting an attached arbitrated loop. This port on a loop will have the AL_PA hexadecimal address 00 (loop ID 7E), giving the fabric the highest priority access to the loop. A loop port is the gateway to the fabric for the node ports on a loop.

failover

The process that takes place when one controller assumes the workload of a failed companion controller. Failover continues until the failed controller is operational.

FC HBA

Fibre Channel Host Bus Adapter. An interchangeable term for Fibre Channel adapter.

See also **FCA.**

FCA

Fiber Channel Adapter. An adapter used to connect the host server to the fabric. *Also* called a Host Bus Adapter (HBA) or a Fibre Channel Host Bus Adapter (FC HBA).

See also **FC HBA**

fiber

The optical media used to implement Fibre Channel.

fibre

The international spelling that refers to the Fibre Channel standards for optical media.

Fibre Channel

A data transfer architecture designed for mass storage devices and other peripheral devices that require very high bandwidth.

Fibre Channel adapter

See **FCA.**

GB

Gigabyte. A unit of measurement defining either:

- A data transfer rate.

  See also **GBps**

- A storage or memory capacity of 1,073,741,824 (2^{30}) bytes.

GBps

Gigabytes per second. A measurement of the rate at which the transfer of bytes of data occurs. A GBps is a transfer rate of 1,000,000,000 (10^9) bits per second.

See also **GB.**

Giga (G)

The notation to represent 10^9 or 1 billion (1,000,000,000).

HBA

Host Bus Adapter.

See **FCA.**

host

A computer that runs user applications and uses (or can potentially use) one or more virtual disks created and presented by the controller pair.

Host Bus Adapter

See **FCA.**

host computer

See **host.**

host ports

A connection point to one or more hosts through a Fibre Channel fabric. A host is a computer that runs user applications and that uses (or can potentially use) one or more of the virtual disks that are created and presented by the controller pair.

host-side ports

See **host ports.**

HSV Element Manager

See **Element Manager.**

initialization

A process that prepares a storage system for use. Specifically, the system binds controllers together as an operational pair and establishes preliminary data structures on the disk array. Initialization also sets up the first disk group, called the default disk group.

LUN

Logical Unit Number. A unique identifier used on a SCSI bus to distinguish between devices that share the same bus. SCSI is a parallel interface that allows up to eight devices to be connected along a single cable.

management agent

See **Element Manager**.

manager

SSSU term for the HSV Element Manager.

See also **Element Manager**.

MB

Megabtye. A term defining either:

- A data transfer rate.

  See also **MBps.**

- A measure of either storage or memory capacity of 1,048,576 (2^{20}) bytes.

MBps

Megabytes per second. A measure of bandwidth or data transfers occurring at a rate of 1,000,000 (10^6) bytes per second.

Mega

A notation denoting a multiplier of 1 million (1,000,000).

mirrored caching

A process in which half of each controller's write cache mirrors the companion controller's write cache. The total memory available for cached write data is reduced by half, but the level of protection is greater.

mirroring

The act of creating an exact copy or image of data.

monitor

SSSU term for Environmental Monitoring Unit (EMU).

See **EMU**.

OSM

Open SAN Manager. A centralized, appliance-based monitoring and management interface that supports multiple applications, operating systems, hardware platforms, storage systems, tape libraries and SAN-related interconnect devices. It is included and resides on the SANworks Management Appliance, a single aggregation point for data management.

parity

A method of checking if binary numbers or characters are correct by counting the ONE bits. In odd parity, the total number of ONE bits must be odd; in even parity, the total number of ONE bits must be even. Parity information can be used to correct corrupted data.

parity bit

A binary digit added to a group of bits that checks to see if there are errors in the transmission.

parity check

A method of detecting errors when data is sent over a communications line. With even parity, the number of ONE bits in a set of binary data should be even. With odd parity, the number of ONE bits should be odd.

password

A security interlock whose purpose is to allow:

- a management agent control of only certain storage systems
- only certain management agents control of a storage system

physical disk

A disk drive mounted in a disk drive enclosure that communicates with a controller pair through the device-side Fibre Channel loops. A physical disk is hardware with embedded software, as opposed to a virtual disk, which is constructed by the controllers. Only the controllers can communicate directly with the physical disks.

The physical disks, in aggregate, are called the array and constitute the storage pool from which the controllers create virtual disks.

physical disk array

See **array**.

port

A Fibre Channel connector on a Fibre Channel device.

port_name

A 64-bit unique identifier assigned to each Fibre Channel port. The port_name is communicated during the login and port discovery processes.

preferred path

A preference for which controller of the controller pair manages the virtual disk. This preference is set by the user through the HSV Element Manager when creating the virtual disk. A host can change the preferred path of a virtual disk at any time. The primary purpose of preferring a path is load balancing.

read caching

A cache method used to decrease subsystem response times to a read request by allowing the controller to satisfy the request from the cache memory rather than from the disk drives. Reading data from cache memory is faster than reading data from a disk. The read cache is specified as either on or off for each virtual disk. The default state is on.

read ahead caching

A cache management method used to decrease the subsystem response time to a read request by allowing the controller to satisfy the request from the cache memory rather than from the disk drives.

redundancy

1. Element Redundancy—The degree to which logical or physical elements are protected by having another element that can take over in case of failure. For example, each loop of a device-side loop pair normally work independently but can take over for the other in case of failure.
2. Data Redundancy—The level to which user data is protected. Redundancy is directly proportional to cost in terms of storage usage; the greater the level of data protection, the more storage space is required.

small computer system interface

See **SCSI.**

snapshot

A temporary virtual disk (VD) that reflects the contents of another virtual disk at a particular point in time. A snapshot operation is only done on an active virtual disk. Only one snapshot of an active virtual disk can exist at any point. The active disk and its snapshot constitute a virtual family.

See also **active virtual disk, virtual disk copy,** *and* **virtual disk family.**

spare policy

This is the disk failure protection level set for the storage system (cell) and disk groups.

See also **disk failure protection**.

SSN

Storage System Name. An HSV Element Manager-assigned, unique 20-character name that identifies a specific storage system.

storage

The term SSSU uses for virtual disk as in the ADD, DELETE, and SET STORAGE commands.

storage pool

The aggregated blocks of available storage in the total physical disk array.

storage system

The controllers, storage devices, enclosures, cables, and power supplies and their software.

Enterprise Storage System Scripting Utility (SSSU)

A command-line application supplied in the host operating system kits that lets you configure and control HSV controllers.

Storage System Name

See **SSN.**

StorageWorks by Compaq

The Compaq trademarked name used to describe the set of rack-mounted enclosures containing controllers, transceivers, I/O modules, EMUs, disk drives, cables, blowers, and power supplies used to design and configure a solution-specific storage system.

switch

An electro-mechanical device that initiates an action or completes a circuit.

TB

Terabyte. A term defining either:

- A data transfer rate.

  See also **TBps.**
- A measure of either storage or memory capacity of 1,099,5111,627,776 (2^{40}) bytes.

TBps

Terabytes per second. A data transfer rate of 1,000,000,000,000 (10^{12}) bytes per second.

uninitialized storage system

A state in which the storage system is not ready for use.

See also **initialization**.

Virtual Disk

A simulated disk drive created by the controllers as storage for one or more hosts. The virtual disk characteristics, chosen by the storage administrator, provide a specific combination of capacity, availability, performance, and accessibility. A controller pair simulates the characteristics of the virtual disk by deploying the disk group from which the virtual disk was created.

The host computer sees the virtual disk as "real," with the characteristics of an identical physical disk.

See also **active virtual disk, virtual disk copy, virtual disk family,** *and* **virtual disk snapshot**

virtual disk copy

A clone or exact replica of another virtual disk at a particular point in time. Only an active virtual disk can be copied. A copy immediately becomes the active disk of its own virtual disk family.

See also **active virtual disk, virtual disk family,** *and* **virtual disk snapshot**

virtual disk family

A virtual disk and its snapshot, if a snapshot exists, constitute a family. The original virtual disk is called the active disk. When you first create a virtual disk family, the only member is the active disk.

See also **active virtual disk, virtual disk copy,** *and* **virtual disk snapshot.**

virtual disk snapshot

See **snapshot.**

Vraid0

A virtualization technique that provides no data protection. Data host is broken down into chunks and distributed on the disks comprising the disk group from which the virtual disk was created. Reading and writing to a Vraid0 virtual disk is very fast and makes the fullest use of the available storage, but there is no data protection (redundancy) unless there is parity.

Vraid1

A virtualization technique that provides the highest level of data protection. All data blocks are mirrored or written twice on separate physical disks. For read requests, the block can be read from either disk, which can increase performance. Mirroring takes the most storage space because twice the storage capacity must be allocated for a given amount of data.

Vraid5

A virtualization technique that uses parity striping to provide moderate data protection. Parity is a data protection mechanism for a striped virtual disk. A striped virtual disk is one whose data to and from the host is broken down into chunks and distributed on the physical disks comprising the disk group in which the virtual disk was created. If the striped virtual disk has parity, another chunk (a parity chunk) is calculated from the set of data chunks and written to the physical disks. If one of the data chunks becomes corrupted, the data can be reconstructed from the parity chunk and the remaining data chunks.

World Wide Name

See **WWN.**

write back caching

A controller process that notifies the host that the write operation is complete when the data is written to the cache. This occurs before transferring the data to the disk. Write back caching improves response time since the write operation completes as soon as the data reaches the cache. As soon as possible after caching the data, the controller then writes the data to the disk drives.

write caching

A process when the host sends a write request to the controller, and the controller places the data in the controller cache module. As soon as possible, the controller transfers the data to the physical disk drives.

WWN

World Wide Name. A unique Fibre Channel identifier consisting of a 16-character hexadecimal number. A WWN is required for each Fibre Channel communication port.

Index

A

E

F

G

H

I

K

L

M