Compaq StorageWorks

# RA8000/ESA12000 Fibre Channel Solution Software V8.4 for *Compaq Tru64™UNIX*

**Installation Reference Guide**

AA–RFAUA–TE/387389-001

**Compaq Computer Corporation**

**April 1999**



This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the manuals, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense. Restrictions apply to the use of the local-connection port on this series of controllers; failure to observe these restrictions may result in harmful interference. Always disconnect this port as soon as possible after completing the setup operation. Any changes or modifications made to this equipment may void the user's authority to operate the equipment.

Warning!
This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Achtung!
Dieses ist ein Gerät der Funkstörgrenzwertklasse A. In Wohnbereichen können bei Betrieb dieses Gerätes Rundfunkstörungen auftreten, in welchen Fällen der Benutzer für entsprechende Gegenmaßnahmen verantwortlich ist.

Attention!
Ceci est un produit de Classe A. Dans un environnement domestique, ce produit risque de créer des interférences radioélectriques, il appartiendra alors à l'utilisateur de prendre les mesures spécifiques appropriées.

JAPAN

この装置は，情報処理装置等電波障害自主規制協議会（ＶＣＣＩ）の基準に基づくクラスＡ情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

USA

This equipment generates, uses, and may emit radio frequency energy. The equipment has been type tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules, which are designed to provide reasonable protection against such radio frequency interference. Operation of this equipment in a residential area may cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference. Any modifications to this device - unless expressly approved by the manufacturer - can void the user's authority to operate this equipment under part 15 of the FCC rules.

# *Contents*

## 3 Installing the Command Console Client



## 4 Creating Your Storage Configuration with the CLI

## Figures

**Figures (cont.)**

## Tables

# *Revision Record*

*This Revision Record provides a concise publication history of this manual. It lists the manual revision levels, release dates, and reasons for the revisions.*

The following revision history lists all revisions of this publication and their effective dates. The publication part number is included in the Revision Level column, with the last entry denoting the latest revision. This publication supports the StorageWorks RA8000/ESA12000 Subsystem for the Compaq Tru64 UNIX Operating Systems.

| Revision Level | Date | Summary of Changes |
|---|---|---|
| AA-RFAUA-TE/387389-001 | April 1999 | Original Release |

# *About This Guide*

*This section identifies the audience of this guide and describes the contents (chapter by chapter) and structure. In addition, it includes a list of associated documents and the conventions used in this guide, as well as Support and Services Contact Information.*

This guide provides the following descriptions:

- How to install StorageWorks software and the Command Console Agent
- How to install the Command Console Client (Optional)
- How to Create Your Storage Configuration with the Command Line Interpreter (CLI)

## Intended Audience

This guide is intended for administrators of StorageWorks RAID Array  Subsystems. Installing the StorageWorks RAID Array Subsystem requires a general understanding of Tru64 UNIX system administration and Tru64 UNIX hardware installation procedures.

## Document Structure

This guide contains the following chapters:

### Chapter 1:  Preparing Your Host for Use with the HSG80

This chapter describes how to access the storage units on your RA8000/ESA12000 Subsystem from your Tru64 UNIX Host Computer.

### Chapter 2: Installing and Configuring the Command Console (CC) Agent

This chapter describes how to install a copy of the Command Console (CC) Agent on your Tru64 UNIX Host System.

### Chapter 3:  Installing the Command Console (CC) Client

This chapter describes how to install, start, and use the Command Console client.

### Chapter 4:  Creating Your Storage Configuration with the CLI

This chapter contains instructions for creating an initial configuration for your HSG80 Controller using the Command Line Interpreter (CLI). It briefly describes the CLI and how to access it.

### Appendix A: Planning Your Storage Configuration

This appendix describes the RAID configuration options and RAID concepts that you need to know to create your storage configuration.

### Appendix B: Command Console LUN

This appendix discusses the Command Console LUN, a virtual disk that can be used for Agent communication.

### Appendix C: Understanding RAID Manager's Mail Messages

This appendix provides information to help you understand the contents of the automatic mail message(s) RaidManager sends upon the occurrence of an event, if so enabled.

### Appendix D: SNMP MIB Information

This appendix provides information about the SNMP traps that may be generated by the Agent.

### Appendix E: Fibre Channel Switch Installation and Basic Set-up

This Appendix is designed to give you basic instructions for installing the Fibre Channel switch and for making the necessary connections.

### Appendix F: Connections, Topography, and other Fibre Related Settings for the HSG80 on Compaq Tru64® UNIX

This appendix is a discussion with intent to get you started with some of the new Fibre Channel (FC) terms as they apply to the HSG80 connected to an Alpha server running the Compaq Tru64 UNIX (previously Digital Unix).

## Associated Documents

In addition to this guide, the following documentation is useful to the reader:

| Document Title | Order Number |
|---|---|
| *RA8000/ESA12000 Storage Subsystem User's Guide* | 387404-001/EK-SMCPR-UG |
| *Command Line Interpreter (HSG80) Reference Manual* | 387402-001/EK-HSG80-RG. A01 |
| *Command Console Version 2.1 (HSG80) User's Guide* | 387405-003/AA-RFA2C-TE |
| *Release Notes – HSG80 Solutions Software Version 8.4 for Compaq Tru64 UNIX* | 387383-001/AA-RFAUA-TE |
| Quick Setup Guide RAID Array 8000/ESA12000 Fibre Channel Storage Subsystem for Compaq Tru64 UNIX (ACS V8.4 for Fibre Channel Switch) | 387388-001/AA-RFASA-TE |
| Quick Setup Guide StorageWorks Fibre Channel Switch | AA-RHCOA-TE |
| Fibre Channel Switch User's Guide | AA-RHBYA-TE |

## Conventions

This guide uses the following documentation conventions:

| Style | Meaning |
|---|---|
| **boldface monospace type** | To be input by the user. |
| *italic type* | For emphasis, manual titles, utilities, menus, screens, and filenames |
| plain monospace type | Screen text. |
| # | Represents the Compaq Tru64 UNIX system prompt. Do not type it as part of the information given to be input by the reader. |

### Nomenclature Convention

| RAID Advisory Board Description | RAID Array |
|---|---|
| RAID 0 | STRIPEset |
| RAID 1 | MIRRORset |
| RAID 0+1 | STRIPED MIRRORset |
| RAID 3/5 | RAIDset |

## Getting Help

If you have a problem and have exhausted the information in this guide, you can get further information and other help in the following locations.

## Compaq Web Site

The Compaq Web Site has information on this product as well as the latest drivers and Flash ROM images. To access the Compaq Web Site, log on to the Internet at:

*http://www.compaq.com/products/storageworks*

## Telephone Numbers

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
For Compaq technical support:
In the United States and Canada, call 1-800-386-2172

# 1

# *Preparing Your Host for Use with the HSG80*

*This chapter describes how to access the storage units on your RA8000/ESA12000 Subsystem from your Tru64 UNIX Host Computer. It includes a description of concepts and terms that you need to know, a discussion on device naming, instructions for loading the StorageWorks RAID utilities, instructions for creating device special files in Tru64 UNIX, plus tips on some helpful utilities.*

**NOTE**

Tru64 UNIX was formerly called DEC OSF/1 and more recently, "Digital UNIX." The term "Tru64 UNIX" is used throughout this chapter.

You must perform all configuration set up and parameter definitions, and run all utilities from a maintenance terminal or a terminal emulator connected to your Controller's maintenance port.

## 1.1 Basic Configuration Steps

Once you have completed the initial setup and configuration of your RA8000/ESA12000 Subsystem, you will be able to use RAIDsets and other storage containers much like single disks.

### 1.1.1 Configuring for Tru64 UNIX V4.0F

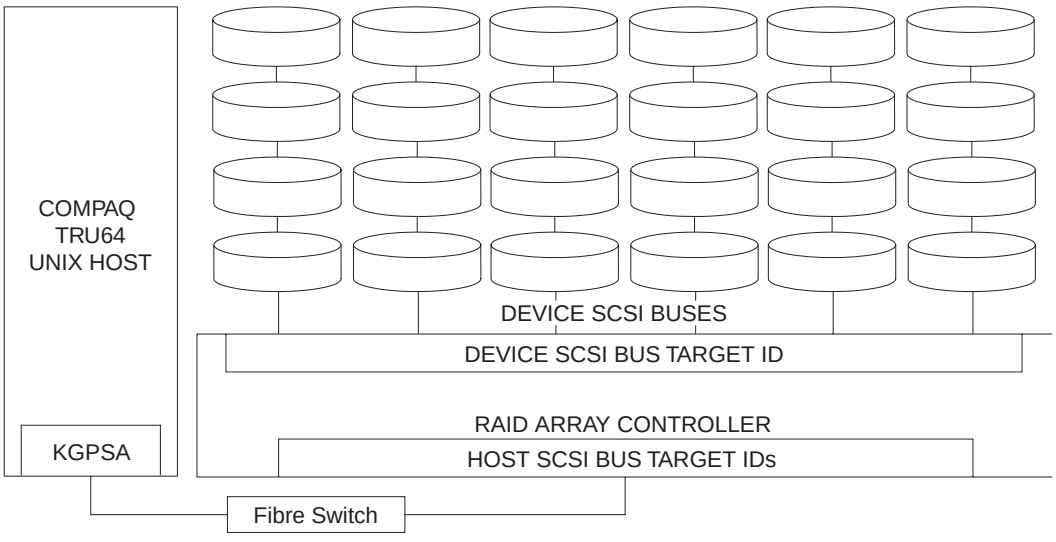
1. Install the DIGITAL Fibre Channel host adapter in your Tru64 UNIX System and connect the RAID Array to the switch (See Appendix E, and refer to the *Quick Setup Guide StorageWorks Fibre Channel Switch* for more information.). Refer to the *Release Notes* for supported adapters. Refer to the associated adapter manuals for installation instructions.
2. Boot the Tru64 UNIX System (see Section 1.2).
3. Complete the steps in Section 1.3 to make a serial connection to the HSG80 Controller and set up the RAID Array.
4. Create disklabels and partitions on the containers (see Section 1.3.1).
5. Create filesystems on the previously created partitions (see Section 1.3.2).

### 1.1.2 Words and Concepts That You Need To Know

It is important to understand the following concepts when working with your RA8000/ESA12000:

The HSG80 Controllers in the subsystem are active on several SCSI buses. The Controllers communicate with the Tru64 UNIX Host Computer on the host SCSI bus and communicate with the devices in the RA8000/ESA12000 on six device SCSI buses. In other words, the Controllers are targets on the host SCSI bus and initiators on the device SCSI buses.

**Figure 1–1 Block Diagram of Storage Connections in an RA8000/ESA12000 Subsystem**



The following terms appear in the block diagram (Figure 1-1) and throughout this chapter:

- *Host SCSI Bus* – The SCSI bus that connects to the host adapter via the Fibre Channel switch. This is sometimes called the "front end" SCSI bus or the "Host-side" bus. On the Host SCSI bus, a HSG80 Controller can be assigned from one to two target IDs. The Controller can present up to eight units through each target ID. The ability to respond to two target IDs instead of just one allows the Controller to present more units to the Host Computer. A Controller with two host-side target IDs can present up to 16 units.

  Target IDs must be in the range of 0–1, but they must not conflict with any other devices on the host SCSI bus, such as the host's SCSI adapter itself. (The host's adapter is usually set to ID 7.) Dual-Redundant Controllers are assigned the same target IDs so that if one Controller fails the other can continue to respond to the same addresses.

  The Controller's host-side target IDs are set using the Controller's Command Line Interpreter (CLI). The CLI is accessed using a maintenance terminal connected to the port on the front bezel of the Controller or via SWCC.

  Device SCSI Buses – The HSG80 Controller occupies one SCSI ID on each device bus (port). The device-side ID is determined by the Controller slot in which the Controller is installed.

  In a single Controller configuration, the Controller should be in the top Controller slot and is SCSI ID 7 on all device buses.

  In a Dual-Redundant Controller configuration, the Controller in the top Controller slot is SCSI ID 7 and the Controller in the bottom Controller slot is SCSI ID 6.
- *Controller Storage Container* – A single- or multiple-device storageset. A single disk can be a container; all stripesets, mirrorsets, and RAIDsets are containers.
- *Controller Unit* – "Unit" is the Controller term for a storage container that is available for use by the Tru64 UNIX Host. Until a storage container is made into a unit with the ADD UNIT command, it is not available to the host.
- *Tru64 UNIX Disk Device* – (As used in reference to a HSG80 Controller subsystem.) "Disk device" is the Tru64 UNIX term for a Controller Unit. You must create device special files to associate the Controller Units with Tru64 UNIX disk device names.
- *LUN and Logical LUN* – The term LUN (Logical Unit Number) refers to two different entities in an HSG80 Controller environment, the host-side LUN and the device-side LUN.

- *Host-side LUN* – The Host-side LUN is used by the Host Operating System to uniquely identify a container.
- *Device-side LUN* – In current HSOF firmware, this LUN is always 0. The device-side target ID and the port number are the only entities that are currently used by the HSG80 Controller to uniquely identify a device on the device-side SCSI Bus.
- *Device SCSI Bus* – The six SCSI buses that connect the HSG80 Controller to the SCSI–2 devices. These are sometimes called the "back end" SCSI buses or the "device-side" buses. These buses are also referred to as device ports and are numbered 1 to 6. The device buses are built into the backplane in your RA8000/ESA12000 Cabinet; there are no device bus cables.
- *Controller SCSI IDs* – Each Controller is assigned SCSI IDs on both the host bus and all device buses.

### 1.1.3 Device Special files

Tru64 UNIX does not strictly enforce one format for naming device special files; however, the format below is recommended.[1]

*Example:*

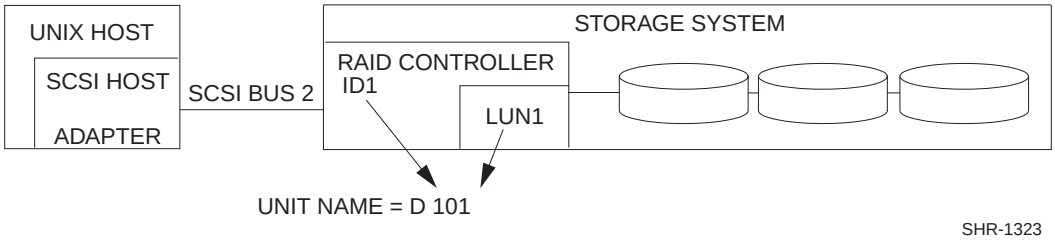
*/dev/rzxnny*

Where:

- *rz* indicates the SCSI drive
- *x* is the LUN indicator
- *nn* is the DEVICE logical number
- *y* is the partition designator

Figure 1-2 is a sample configuration of a single unit in a storage subsystem.

**Figure 1–2 Device Naming Example**




[1] Some TRU64 UNIX utilities, such as *iostat*, and certain startup procedures, do not recognize this device naming format, and recognize only LUN-0 devices; e.g., rz18c.

The sample configuration has the following characteristics:

- Host SCSI Bus = 2
- Controller Target ID = 1
- Storage Container = LUN 1
- Unit Number = D101
- Host Partition Letter c (not shown)

Using these values to derive the device name looks like this:

"rz" + LUN letter + ((8 * Host SCSI Bus #) + (Controller Target ID)) + partition letter

**or**

"rz" + "b" + ((8 * 2) + (1)) + "c"

**or**

rzb17c

The equivalent character mode device name would be:

rrzb17c

The preceding naming scheme is used throughout this chapter. However, be aware that Tru64 UNIX device naming schemes do not have to follow this format. This particular naming scheme was chosen to avoid conflicts with the Tru64 UNIX default SCSI device naming conventions.

## 1.2 Rebooting the Kernel for Tru64 UNIX V4.0F

Tru64 UNIX V4.0F provides Dynamic Device Recognition (DDR). DDR simplifies the process of adding new devices to your AlphaServer by allowing you to add new devices by rebooting the Kernel. The Kernel rebuild will automatically create device driver files for the storagesets you created on your subsystem.

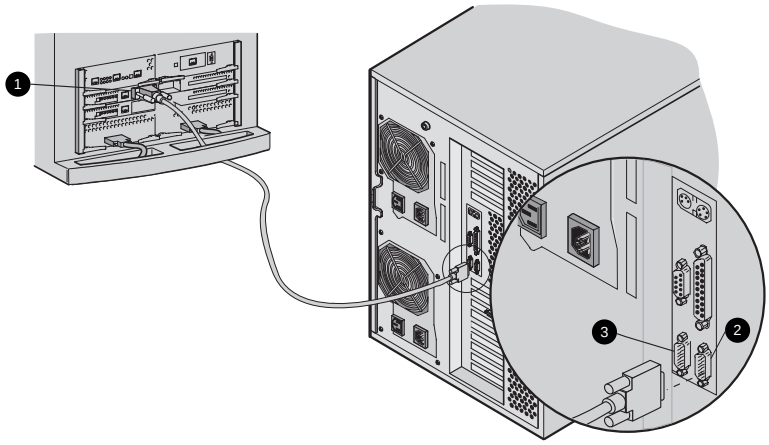
## 1.3 Access the Command Line Interpreter (CLI) to Configure the Subsystem

### 1.3.1 Connect a PC as a Maintenance Terminal

Perform the following steps to connect a PC and refer to Figure 1–3:

1. Plug the RJ-12 communication cable connector into the maintenance port of the subsystem HSG80 controller (1).
2. Plug the 9-pin communication cable connector into the COM1 or COM2 port of the PC (2) or (3).

**Figure 1–3 Connecting a PC as a Maintenance Terminal**




**NOTE**

Make sure that there is a ferrite bead installed on the communication cable.

3. Turn ON the power to the PC.

### 1.3.2 Establish a Connection Between the PC and the Subsystem Controller

Perform the following steps to establish a connection:

1. Start a communication program on the PC (for example: *Hyper Terminal*).
2. Set the program to use the PC serial port, COM1 or COM2 (whichever port the communication cable is connected to).
3. Set the communication parameters as follows:
   - 8 bits
   - 9600 baud
   - 1 stop bit
   - No parity

4. Issue a connect command to establish connection with the controller
5. Press the *Enter* key to display the CLI prompt:

   **HSG80>**

6. You must set the port topology to "FABRIC" and restart both controllers. Type these commands:

   **SET THIS PORT_1 TOPOLOGY = FABRIC**
   **SET THIS PORT_2 TOPOLOGY = FABRIC**
   **RESTART OTHER**
   **RESTART THIS**

7. To show the connections made by the controller tot he server via the switch, type `SHOW CONNECTION` as the CLI prompt. For basic instalation something similar to the following will appear:

```
Connection

Name Operating system Controller  Port Address  Status  Unit Offset
!NEWCON01   WINNT OTHER       1   210013   OL other      0
     HOST_ID=1000-0000-C920-A6C5   ADAPTER_ID=1000-0000-C920-A6C5

!NEWCON02   WINNT THIS        2   210013   OL this       100
     HOST_ID=1000-0000-C920-A6C5   ADAPTER_ID=1000-0000-C920-A6C5

HSG80>
```

8. You must change `operating system` to `Digital UNIX` for each connection. The syntax for the command is:

   **set {connection name} operating_system = DIGITAL_UNIX**

Where {*connection name*} is the name from the `SHOW CONNECTION` command. In this example you type:

**set !NEWCON01 operating_system = DIGITAL_UNIX**
**set !NEWCON02 operating_system = DIGITAL_UNIX**

9. Finally, type `SHOW CONNECTION` at the HSG80> prompt to show the change:

```
Connection

Name Operating system Controller  Port Address  Status  Unit Offset
!NEWCON01   DIGITAL_UNIX      1   210013   OL other      0
     HOST_ID=1000-0000-C920-A6C5   ADAPTER_ID=1000-0000-C920-A6C5

!NEWCON02   DIGITAL_UNIX      2   210013   OL this       100
     HOST_ID=1000-0000-C920-A6C5   ADAPTER_ID=1000-0000-C920-A6C5

HSG80>
```

**NOTE**

If the connections do not appear, either the installation or the power-on sequence was incorrect. Review and if necessary, perform Steps 1 through 9 again.

10. From your Tru64 console type: `file /dev/rrz*c` at the prompt.
11. In the output from this command you will see an entry for *HSG80CCL*; this is the virtual command console LUN.

## 1.4 Preparing LUNs for Access by Tru64 UNIX Filesystem

Tru64 UNIX treats a HSG80 LUN much like a SCSI disk; therefore, to prepare your LUN for access by the Tru64 UNIX filesystem, you must do the following:

- Create the partitions on the LUN using *disklabel*
- Create a filesystem on the LUN
- Mount the filesystem to be able to access it

### 1.4.1 Creating the Partitions on a LUN Using disklabel

You create the partitions on a LUN by issuing a *disklabel* command. The *disklabel* command partitions the LUN for access by the Tru64 UNIX Operating System. Tru64 UNIX defines only partitions a, b, c, and g for the HSG80 Controller. In addition, refer to the *rz* and *disktab man* pages for more information about the *disklabel* utility.
To create the read/write partitions on a LUN using the default partition sizes, enter the following:

```
# disklabel -rw device HSG80
```

Where:

- *HSG80* defines this LUN as attached to the RA8000/ESA12000 Controller. Always use the device label HSG80 , regardless of what drive types make up the LUN.
- *device* is the character device name as defined in Section 1.1.3.

For example, to create partitions on block device, *rza8*, enter:

```
# disklabel -rw rza8 HSG80
```

To view a LUN partition, enter:

```
# disklabel -r device
```

An example output of reading a disklabel follows. Note that some partitions overlap each other.

```
  # disklabel -r rza8
  # /dev/rrza8a:
  type: SCSI
  disk: HSG80
  label:
  flags: dynamic_geometry
  bytes/sector: 512
  sectors/track: 85
  tracks/cylinder: 16
  sectors/cylinder: 1360
  cylinders: 12085
  sectors/unit: 16435880
  rpm: 3600
  interleave: 1
  trackskew: 0
  cylinderskew: 0
  headswitch: 0           # milliseconds
  track-to-track seek: 0  # milliseconds
  drivedata: 0
8 partitions:
        size     offset   fstype   [fsize    bsize    cpg]
 a     131072       0     unused    1024     8192          #  (Cyl.     0 - 96*)
 b     262144    131072   unused    1024     8192          #  (Cyl.    96*- 289*)
 c   16435880       0     4.2BSD    1024     8192     16   #  (Cyl.     0 - 12085*)
 d       0          0     unused    1024     8192          #  (Cyl.     0 - -1)
 e       0          0     unused    1024     8192          #  (Cyl.     0 - -1)
 f       0          0     unused    1024     8192          #  (Cyl.     0 - -1)
 g   16042664   393216   unused    1024     8192          #  (Cyl.   289*- 12085*)
 h       0          0     unused    1024     8192          #  (Cyl.     0 - -1)
```

### 1.4.2 Creating a Filesystem on a LUN

**NOTE**

The *newfs* command is given here as an example. For Advanced File System (ADVFS) and for making devices available for Logical Storage Manager (LSM), similar types of commands exist. For additional information, please consult the related documentation.

Use the *newfs* command to create a UFS filesystem on a LUN the same way that you would create a filesystem on a disk device by entering the following:

```
# newfs device partition
```

Where:

- *device* is the character device name as defined in Section 1.1.3.
- *partition* is the letter representing the partition, a, b, c, or g on which you want to create a filesystem (usually c for the whole drive).

For example, to create a UFS filesystem on partition c of character device rrza8, enter the following:

```
# newfs /dev/rrza8c
```

This creates a UFS filesystem on LUN 0 of this subsystem at SCSI-bus address 2, target ID 0 on the c partition (whole disk-device).

### 1.4.3 Mounting the Filesystem

To access the LUN, mount it as a device filesystem to a mount point. For example:

```
# mount /dev/rza8c /mnt
```

To view the mounted filesystem, enter: **df**

The LUN is now accessible to the filesystem just as a disk device would be. The filesystem can not see the RAID functionality and number of physical devices attached to the HSG80 Controller. This device appears as a single LUN or "disk" to the user as viewed by the filesystem.

## 1.5 DECsafe Available Server Environment (ASE)

HSG80 disk devices can be used with the DECsafe Available Server Environment (ASE) for Tru64 UNIX provided a valid Host configuration (including Host adapters) is used to support them. Refer to the *Tru64 UNIX ASE Installation and User's Guide, Software Product Description, SPD: 44.17.xx,* for further information. Refer to the *Release Notes* for supported host adapters and Tru64 UNIX version levels for ASE.

## 1.6 Using *genvmunix*

If you use *genvmunix* to initialize the system and *doconfig* to build a new configuration file, the new configuration file will only list the HSG80 LUN 0 units; nonzero LUNs will not appear in the new configuration file.

Before rebuilding a configuration file using *genvmunix*, save any existing customized configuration file that has entries for HSG80 units. After rebuilding the configuration file, add the entries from the saved configuration file to the new configuration file.

## 1.7 HSG80 Units and Tru64 UNIX Utilities

This section contains notes on the interactions of some Tru64 UNIX utilities with storage units in your RA8000/ESA12000 Subsystem.

### 1.7.1 File

You can use the Tru64 UNIX *file* utility to determine if a Controller Unit can be accessed from the host.

The unit that you want to test must already have a character mode device special file and the correct disk label.

The following example uses the HSG80 unit D101. Run the file command and specify the character mode device special file, such as:

```
# /usr/bin/file /dev/rrzb17a
```

The device activity indicator (green light) will illuminate on the device if the information is not in cache. If the unit is a multi-device container, only one of the devices from that container will illuminate. The Tru64 UNIX Operating System should display something like the following output after the command is entered:

```
/dev/rrzb17a character special (8/33856) SCSI #2 HSG80 disk #146 (SCSI ID #1)
```

- 8 is the major number
- 33856 represents the minor number
- 2 is the SCSI host-side bus number
- 146 is the drive number as listed in the Configuration File
- 1 is the Controller Target ID

If the only output that is returned from the file command is the major and minor number, then either the device is not answering or the device special file does not have the correct minor number. Check the minor number to be sure that it matches the host SCSI bus number, the Controller target ID, and the LUN of Controller Unit.

If an error occurs regarding the disk label, there is good probability that the device can be accessed. This error can usually be fixed by creating the disk label with the Tru64 UNIX *disklabel* utility.

**NOTE**

For major and minor number calculations see the manual pages *, "man SCSI".*

### 1.7.2 Reading from the device, dd

Check the created device using *dd* on the 'raw' device to see if there is a full communication path between devices and Tru64 UNIX; for example:

```
# dd if=/dev/rrzb17a of=/dev/null
```

This will read the full disk-device until you press ^*c* or the device has been read. If the test is successful, the device activity LED (green) on the device lights. If the device consists of multiple disks, all these will be lit.

### 1.7.3 scu

You can use the *SCSI CAM Utility (scu)* program to see which HSG80 units are available to the Tru64 UNIX Operating System. It is located in the /sbin directory and documented in the REF Pages.
The *scu* command, *scan edt*, polls all devices on the host-side SCSI buses. This allows you to show what devices are available from all host-side SCSI buses. The device special files do not have to exist for *scu* to see the devices. Example: scan SCSI bus 2, where your RA8000/ESA12000 is connected:

```
# /sbin/scu scan edt bus 2
# /sbin/scu show edt bus 2

 CAM Equipment Device Table (EDT) Information:

 Bus: 2, Target: 1, Lun: 0, Device Type: Direct Access
 Bus: 2, Target: 1, Lun: 1, Device Type: Direct Access
 Bus: 2, Target: 1, Lun: 2, Device Type: Direct Access
 Bus: 2, Target: 1, Lun: 3, Device Type: Direct Access
```

For detailed information about one of the units, you may use:

```
# /sbin/scu show device bus 2 target 1 lun 0
```

The preceding command line gives you the SCSI inquiry of the selected device.

HSG80 units appear like any other SCSI device. All four entries for Bus 2 Target 1 in the example display are HSG80 units. The last entry would be for unit D103 on host SCSI bus 2.

### 1.7.4 iostat

You can use the *iostat* utility to view performance statistics on RA8000/ESA12000 storage units. (Set your terminal screen to 132 columns before running *iostat*.)

The output from *iostat* shows the number of devices (LUNs) that have been defined in the configuration file. It is much easier to interpret the output if the configuration file contains entries for all eight devices. If the configuration file does not contain entries for all devices, the *iostat* output has fewer columns and it is difficult to correlate each column with a specific device.

The *iostat* utility only recognizes device names in the format rznn where nn is calculated as:

(8 * Host SCSI Bus #) + (HSG80 Target ID)

(This is the same formula and format that is used for the configuration file.)
Invoke the *iostat* utility using the following format:

```
iostat rznn s t
```

Where:

- *rznn* is the device name
- The *s* is optional and denotes the amount of time, in seconds, between screen updates
- The *t* is optional and denotes the total number of screen updates

The output from *iostat* shows all devices that have device name rznn. The information for LUN 0 is in the first column, the information for LUN 1 is in the second column, and so forth.

```
# iostat rz16 5 4

 rz16     rz16     rz16     rz16     rz16     rz16     rz16     rz16
bps tps bps tps bps tps bps tps bps tps bps tps bps tps bps tps
0   0   0   0   0   0   0   0   0   0   0   0   0   0  126   3
0   0   0   0   0   0   0   0   0   0   0   0   0   0 1618  34
0   0   0   0   0   0   0   0   0   0   0   0   0   0 1639  34
0   0   0   0   0   0   0   0   0   0   0   0   0   0 1610  34
```

The above display shows activity on all 8 LUNs attached to one SCSI-bus (2) one Target 0. The above list represents information on rza16, rzb16 ... rzh16. The device with activity is device rzh16.

# 2

# *Installing and Configuring the Command Console (CC) Agent*

*This chapter describes how to install a copy of the Command Console (CC) Agent on your **Compaq Tru64 UNIX** Host System connected to StorageWorks RAID Array 8000 Subsystems. You can install and configure the CC Agent automatically using the **`setld`** command and the CD-ROM.*

## 2.1 Introduction

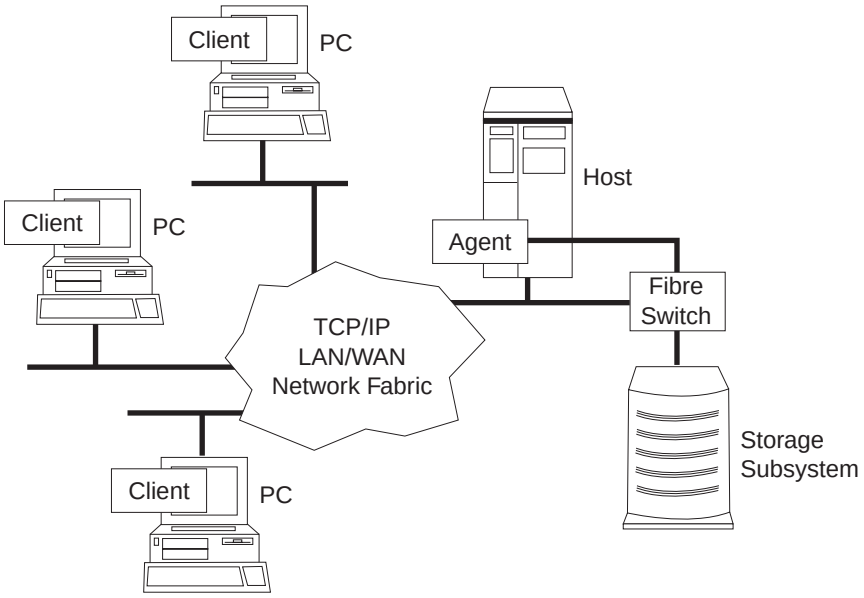
The Command Console (CC) Agent is a companion application to the Command Console (CC) Client Graphical User Interface (GUI) program. The CC Agent serves as Client's assistant in configuring, operating, and monitoring your Host's Storage Subsystems. The CC Agent runs on your Host System as a server application. It connects to CC Client sessions via the TCP/IP network protocol, as shown in Figure 2–1.

The CC Agent can also be used as a standalone application without the CC Client. In this mode of operation, referred to as Agent Only, the CC Agent monitors the status of the subsystem and provides local and remote notification in the event of a failure. Local notification can be made via the `syslog` facility. Remote notification can be made via E-mail and/or SNMP messages to an SNMP Monitor. The CC Client is not required for these types of notifications.

**NOTE**

If you plan to use a PC to configure and monitor your RAID Array Subsystem, you must install the StorageWorks Command Console client. The client is located on the distribution CD in your platform kit, and the instructions are in the Command Console User's Guide, also in your platform kit.

**Figure 2–1 Command Console Client/Agent Connection**




The CC Agent also provides asynchronous fault notification to CC Clients. The program monitors your Host's Storage Subsystems and sends notification messages to all Client sessions connected to it. You can configure the CC Agent to use TCP notification to Client and SNMP notification to an SNMP-compatible monitoring application.

### 2.1.1 Minimum Requirements

The CC Agent is designed to operate with Command Console Client Version 2.1 for Windows NT and Windows 95.

Before you can install the CC Agent, your Host's System resources must meet the minimum requirements specified in Table 2–1.

**Table 2–1 Minimum System Requirements**

| Host Feature | Requirement |
|---|---|
| Architecture | AlphaServer 2 MB free space in /tmp directory, 4 MB free space for the CC Agent installation directory |
| Operating System | Tru64 UNIX 4.0F |
| Controller Compatibility | StorageWorks 2 or 6-port controller running operating firmware Version 8.4 or higher. |

### 2.1.2 Selecting a CC Agent Installation and Configuration Method

You must install and run the CC Agent for Tru64 UNIX on each Alpha system you wish to connect to the CC Client over the network. The **`setld`** command is the supported method used for Tru64 UNIX.

**NOTE**

COMPAQ does **not** advise manually configuring the CC Agent for Tru64 UNIX.

### 2.1.3 Installing and Configuring Automatically

The install program performs the installation from a locally mounted CD-ROM. You must have access to the superuser (root) account and the CD-ROM Drive.

Prior to starting the installation, you need to gather some information (see Table 2–2).

**NOTE**

The CC Agent will automatically start when you install it and will restart automatically if it stops for any reason.

**Table 2–2 Information Needed to Install the Agent**

| Information Required to Complete Installation | How to Respond |
|---|---|
| Names of the Host Systems on which you will run the Client. | Enter the Network names for the Hosts from which you will run the GUI. |
| Level of access a client has for communicating with the storage subsystems. | You can give a client configuration level access only if you specified an access password.<br>The possible options are:<br>**0** = Overall Status<br>**1** = Detailed Status<br>**2** = Configuration |
| A password to gain access to the agent to configure the subsystem with the GUI. | Enter a 4 - 16 character password. The install program does not echo what you type. |
| Names of your subsystems. | Specify unique names for each RAID subsystem in your configuration. |
| The character special file name for accessing the storage subsystem (for example, rrza19c). | Enter the device name that your operating system assigned to the communications logical unit that you created in Appendix C. |
| How do you want a CC Agent server to notify a client when an error condition occurs?<br>Notification schemes available are:<br>**0** = No Error Notification<br>**1** = Notification via a TCP/IP Socket<br>**2** = Notification via the SNMP protocol<br>**3** = Notification via both TCP/IP and SNMP | Selecting this option results in:<br>**0** = No notification over network, but local notification through e-mail and an entry in the system error log file.<br>**1** = Notification messages sent to the GUI and local notification sent through e-mail and an entry in the system error log file.<br>**2** = Notification messages sent to the SNMP monitor and local notification sent through e-mail and an entry in the system error log file.<br>**3** = Notification messages sent to the GUI, the SNMP monitor, and local notification sent through e-mail and an entry in the system error log file. |
| User account of where to mail error and status information. | Specify a mail account where you want to mail the error and status information sent. |

To install and configure the CC Agent automatically, follow these steps:

1. Log on to the Host System as root (superuser).
2. *Type: # **mount -r -t cdfs -o rrip /dev/rz6c /mnt**
   # **cd /mnt/agent/SWCC210**
3. Type: # **setld -l .**
   to start the installation script.

Follow the text-based prompts and enter the appropriate information. Table 2–2 guides you in responding to many of the prompts.

### 2.1.4 Running the CC Agent

Your Agent program started automatically when you completed installation and it will restart automatically if the system is rebooted or stops for any reason.

The installation script places an entry in the "/etc/inittab" file to implement automatic execution of the Agent. The tag field in the file is "*steam*".

### 2.1.5 Reconfiguring the CC Agent

To reconfigure the Agent configuration, execute the *swcc_config* script in the installation directory (*/usr/opt/SWCC210/scripts*) and follow the prompts for the required modification.

### 2.1.6 Uninstalling the CC Agent

To remove the SWCC Agent and its configuration files from your system, use the *setld* utility.

```
# setld -d SWCC210
```

*The example given assumes your CD-ROM is at /dev/rz6c

## 2.2 Troubleshooting/Avoiding Problem Situations

Command Console Client may indicate that it cannot find your subsystem if you attempt to connect via the host SCSI port in either of the following situations:

- A storage subsystem is connected to the host, but the subsystem has no volumes configured on it.
- A storage subsystem is not connected to the host.

To avoid the problem, a Controller Subsystem with at least one volume configured on it must be connected to the host before you attempt to connect CC Client.

Your subsystem has been shipped with at least one volume configured on it to facilitate connection over the host SCSI bus. This is called the *virtual communication LUN* (CCL), which acts like a pseudo disk. The virtual communication LUN is only used as a communication end-point and can not be written as a normal disk. The CCL will show up as target 0, LUN 0 on the bus, unless you create a real RAIDset there; in which case, the CCL will relocate to the next available LUN. In this instance, LUN 1.

# 3

# *Installing the Command Console Client*

*This Chapter covers how to install, launch, and use Command Console Client.*

## 3.1 What is Command Console?

The Command Console (CC) Client is a Graphical User Interface (GUI) for StorageWorks Controllers. Command Console consists of two programs:

- **Command Console Client** is the GUI program designed for use on systems running the Microsoft Windows NT or Windows 95 Operating Systems. It provides a user-friendly method of configuring, operating, monitoring, and troubleshooting your storage subsystem.
- **Command Console Agent** is a companion program that enables the CC Client to communicate with your storage subsystems over a network. Agent is available for operation on a variety of popular Host Operating Systems.

The CC Client connects to your storage subsystem via a TCP/IP-compatible network. It sends (via the Agent) CLI commands to your subsystem's Controllers as you perform subsystem configuration, operation, monitoring, and troubleshooting tasks using its graphical interface. It displays subsystem status by interpreting CLI information returned by the subsystem.

Use the procedures within this chapter to:

- Set up Command Console and establish communication with your storage subsystems.
- Configure your storage devices to create host-accessible volumes by:
  a) Configuring the Client by adding systems to the Navigation Window.
  b) Adding physical devices
  c) Creating virtual disks

## 3.2 Installing Command Console Client

Command Console Client installs from a CD-ROM disk using a standard Windows installation routine on a Windows 95 or Windows NT platform. The program is self-extracting and stores Command Console Client into the directory `C:\Program Files\SWCC` by default. During setup, you have the option to change the disk or directory location.

**To install Command Console:**

1. Place the software CD in the CD-ROM Drive.
2. Run File Manager or Windows Explorer.
3. Navigate to the folder: *drive_letter:\swcc\client\intel*

   NOTE: SWCC for Alpha processors will be located in the folder: drive_letter:\swcc\client\alpha
4. Double-click on Setup.
5. Follow the instructions in the setup program to complete the installation.

Command Console Client installs the Program Group *Command Console* and places seven icons within the group (Figure 3–1). Client also inserts seven selections on the Start menu (Figure 3–2).


**Figure 3–1 Command Console Program Group**

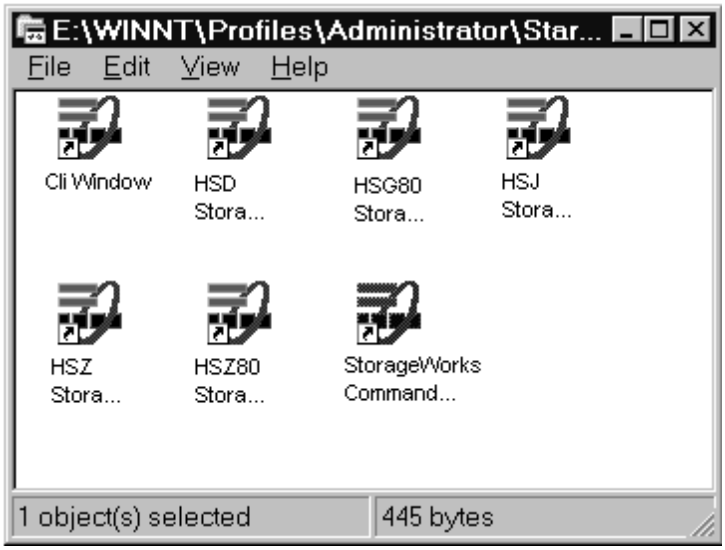
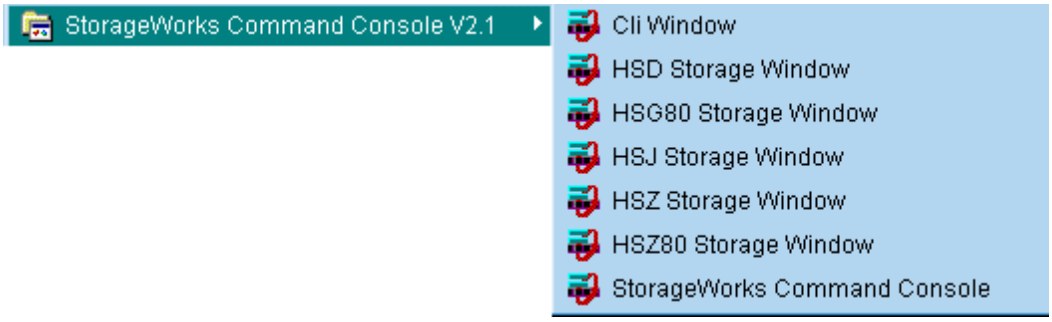


**Figure 3–2 Command Console Client's Start Menu**




## 3.3 Uninstalling StorageWorks Command Console Software

Remove the old version of Command Console software as follows:

1. From the **Start Programs** menu select *Settings*, then *Control Pane*l.
2. Double-click on the *Add/Remove Programs* icon.
3. Select Command Console applets from the list by clicking on them, then click the Add/Remove button.
4. Selection should no longer appear in display window.

## 3.4 Launching the Command Console Client

To start one of the applets, double-click on the applet's icon in the *Command Console* program group or, from the **Start** menu, select the applet's name. Five of the most commonly used applets available are as follows:

- **HSG80 Storage Window:** Displays the Storage Window for HSG80-based storage subsystems. This choice lets you monitor and configure one storage subsystem using Client's graphical user interface. All connection choices are provided: serial line, SCSI bus and network (TCP/IP). Refer to *Command Console V2.1(HSG80) RAID Array 8000/ESA12000 User's Guide, AA-RFA2B-TE/387405-002,* for complete information.
- **HSZ80 Storage Window:** Displays the Storage Window for HSZ80-based storage subsystems. This choice lets you monitor and configure one storage subsystem using Client's graphical user interface. All connection choices are provided: serial line, SCSI bus and network (TCP/IP).

- **HSZ Storage Window:** Displays the Storage Window for HSZ*0-based storage subsystems. This choice lets you monitor and configure HSZ70, HSZ50, HSZ40, SWXRC-04 and HSZ20-based storage subsystems using the Client graphical user interface. All connection choices are provided: serial line, SCSI bus, and network (TCP/IP). For more details refer to the *Command Console V2.1 RAID Array 8000/ESA12000 User's Guide*.
- **CLI Window:** Displays the CLI Window. The CLI window lets you access a controller's command line interface (CLI) for configuring and monitoring your subsystems using text commands. All connection choices are provided: serial line, SCSI bus and network (TCP/IP).
- **StorageWorks Command Console:** Displays the Navigation Window. The Navigation Window is a network navigation tool used to manage and monitor storage subsystems over a TCP/IP network. This choice lets you monitor and configure one or many storage subsystems over a network using Client's graphical user interface.

### 3.4.1 Establishing an Initial Host Connection and Verifying Subsystem Setup

**NOTE**

Prior to making a network connection between the Intel-based System on which you installed the CC Client GUI and your subsystem, you must have completed the steps in this guide to install the CC Agent in your Host System, the CC Client GUI in your Intel-based Client System, and to create a volume on your subsystem. Since the Agent uses a LUN for communication, you cannot make a connection to any subsystem unless the Command Console LUN is enabled or at least one volume has been created on the subsystem.

To establish a network connection proceed as follows:

1. Click on the *Start* button on the taskbar.
2. Click on *Programs.*
3. Click on *Command Console*.
4. Click on *HSG80 Storage Window.*
5. At the Connection Selection dialog box, select the *Network (TCP/IP)* option (Figure 3-3), then click *OK* to display the *Connect Network* dialog box (Figure 3-4).

**Figure 3–3 Connection Selection Dialog Box**

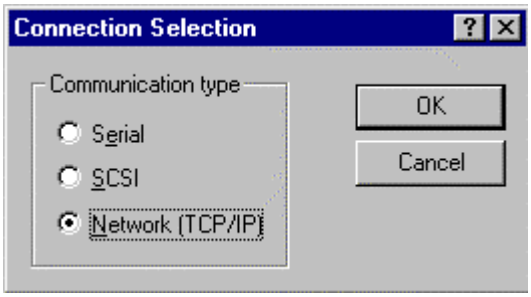
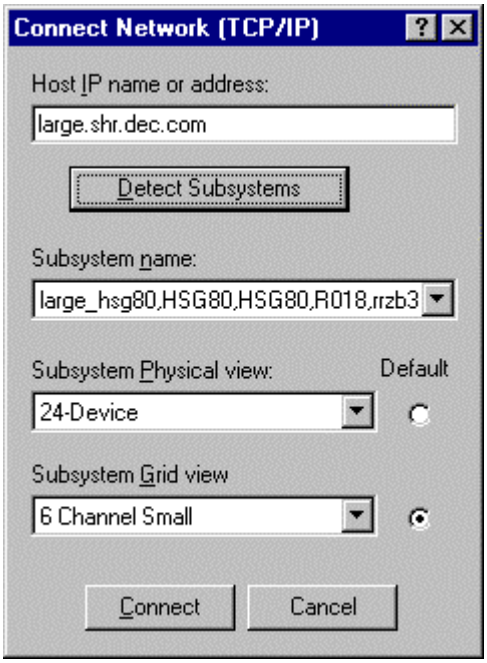



**Figure 3–4 Connect Network Dialog Box for Storage Window**

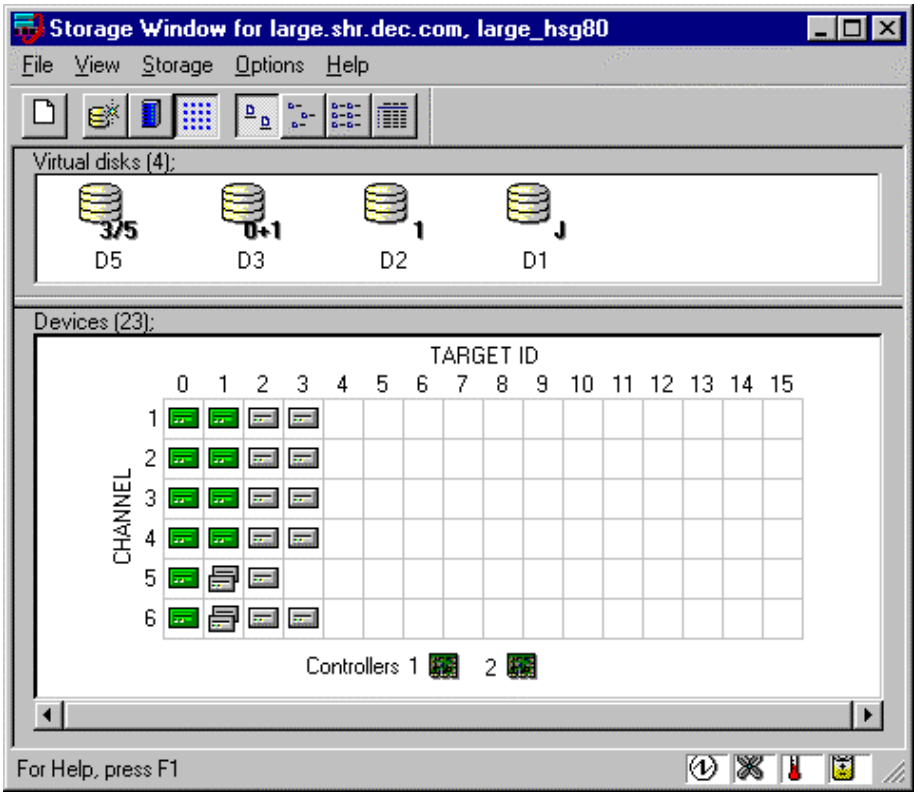



6. On the *Connect Network* dialog box, enter the HOST name or IP address where your HSG80 controller is connected. Then click the *Detect Subsystems* button to display the storage subsystem name. Then click *Connect* to connect to your storage subsystem The Storage Window appears as shown in Figure 3-5. The next step is to configure the properties of the controller.

**NOTE**

The top window panel displays the virtual disks that have been created. The bottom window panel shows the devices you have installed in the RA8000 Fibre Channel Subsystem. On startup, Command Console finds installed drives and displays them in a grid by channel and SCSI ID number.

**Figure 3–5 Storage Window**




## 3.5 Configuring Controller Properties

Your controller's operating parameters are stored in property sheets. Controller property sheets are accessed by double-clicking on a controller icon in the Storage Window or right clicking on the icon and selecting *Properties*. Property sheets are tabbed. To access a sheet, click on its tab.

Changes in all fields causes a controller restart to place the changes in effect. The program prompts you for confirmation before it restarts your controller. A restart of the controller will be detected by the host system. Do not make changes to the controller settings when either file systems are mounted on the RAID Array or applications are using the RAID Array.

**NOTE**

After you initialize a controller restart, there is approximately a 90 second delay while the controller reinitializes.

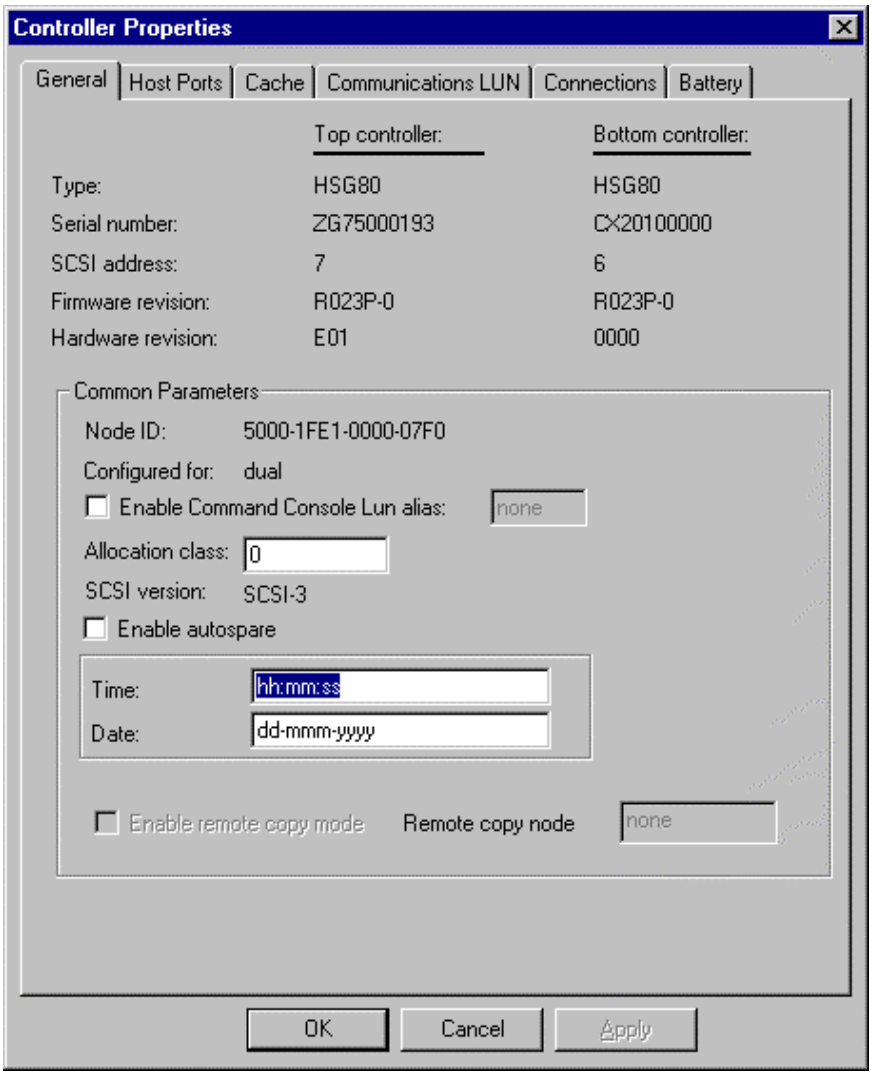
1. Access the controller's property sheets by double-clicking the controllers icon in the Storage Window. When you double-click on a controller's icon the *General* controller properties sheet displays (Figure 3-6). The controller has five other property sheets. Each sheet is accessed by clicking on its tab.

   Confirm the following:

   - Allocation class is 0
   - SCSI Version is SCSI-2

   (You may also set the time and date.)

**Figure 3–6 General Controller Properties Tab**



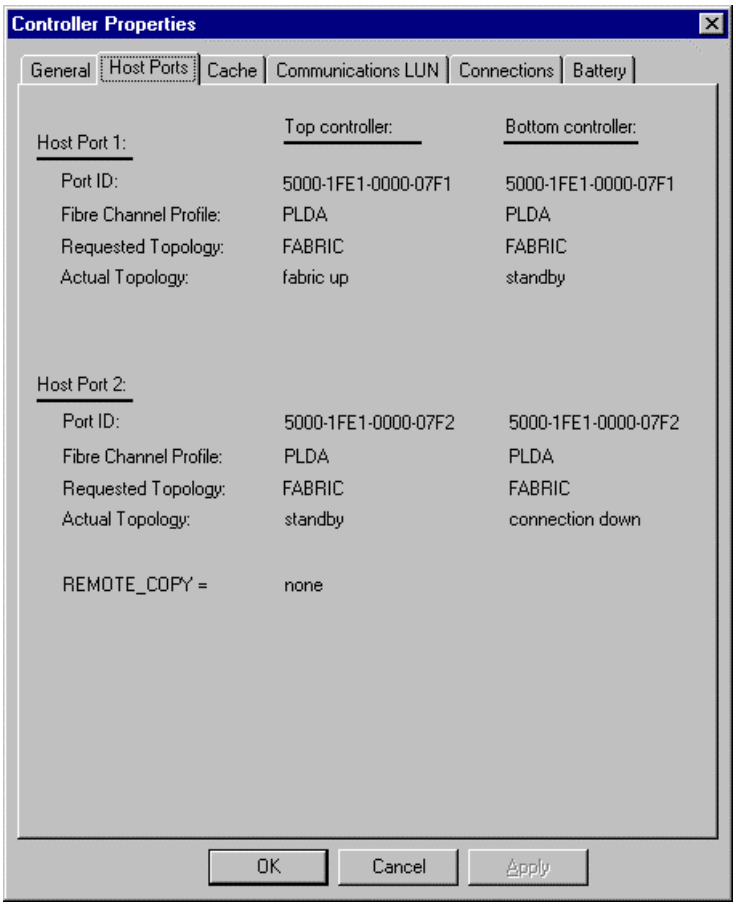
2. Click the *Host Ports* tab to display the host port operating parameters and verify the host port operating parameters shown in Figure 3-7.

   The settings displayed for:

   - Port ID
   - Actual Topology
   - Requested Port Address
   - Actual Port Address

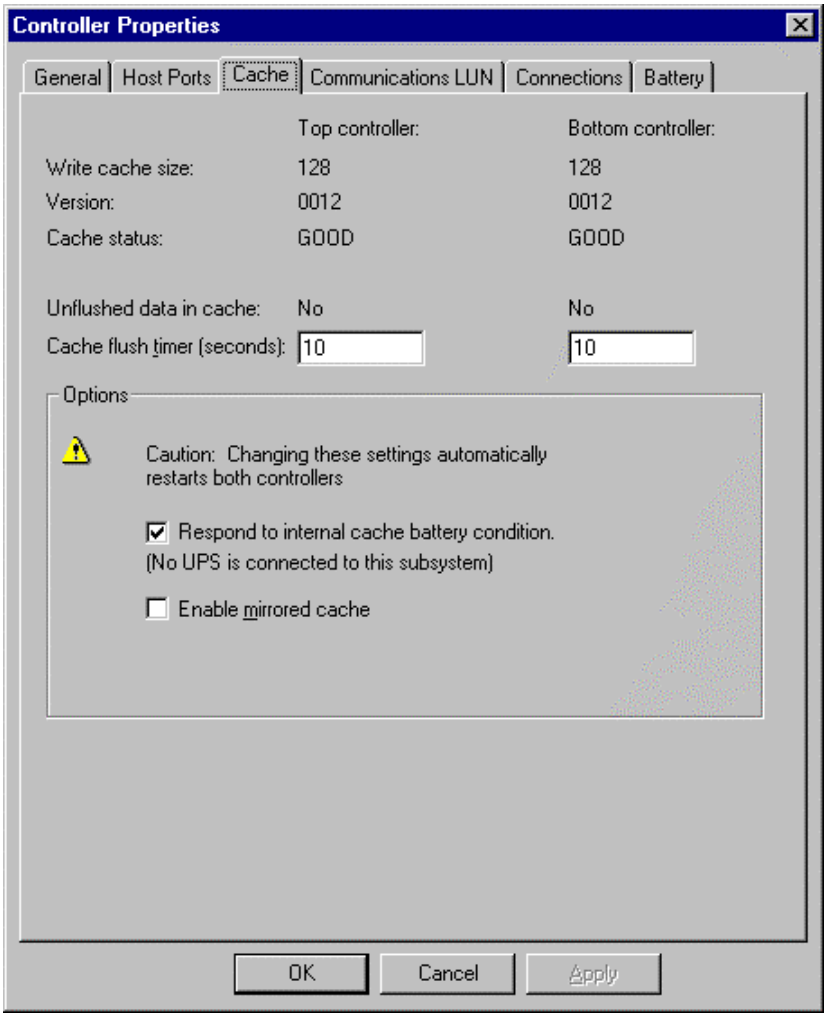
   will vary depending upon your cabling, loop configuration, ALPA settings and number of host ports in use.

**Figure 3–7 Host Ports Controller Properties Tab**



3. Click the *Cache* tab to check cache size. (Figure 3-8)
   Confirm the following:
   - Cache flush timer is 10 (seconds)
   - Respond to internal cache battery condition is selected

**Figure 3–8 Cache Controller Properties Tab**

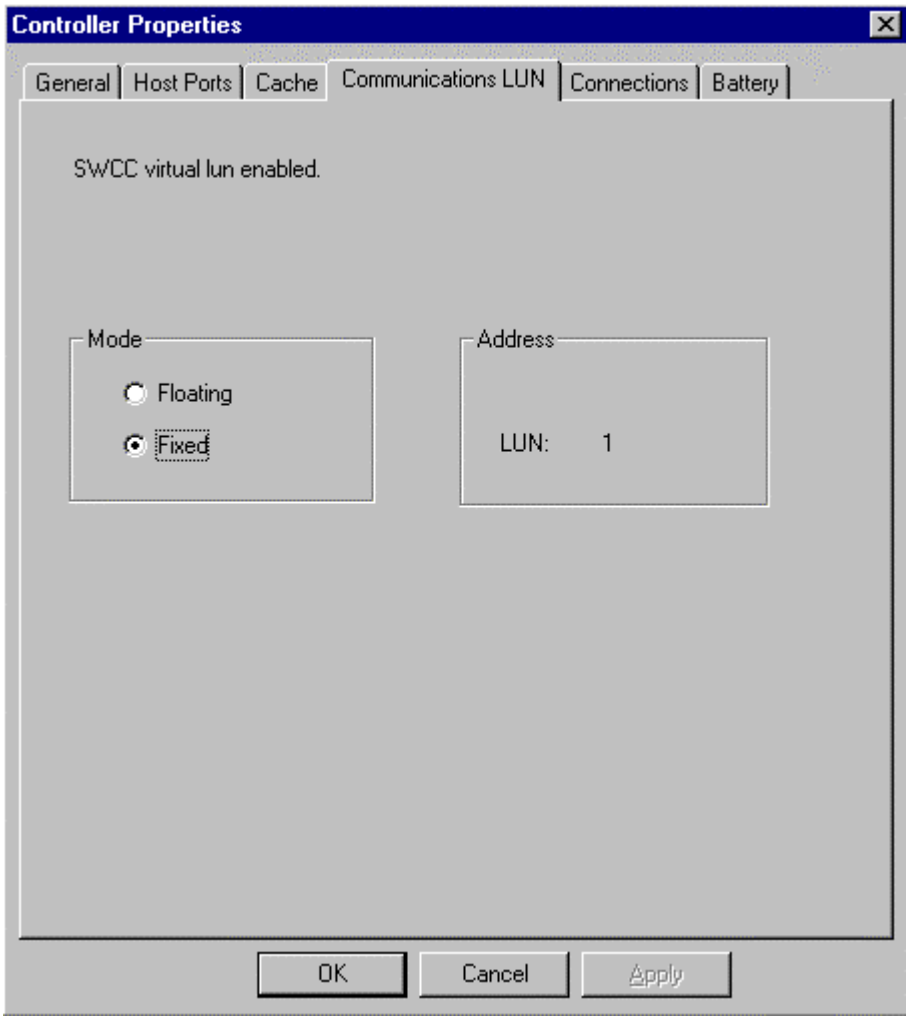


4. Click the *Communications LUN* tab. (Figure 3-9).

**NOTE**

Ensure that *Fixed* is set, and that the SWCC virtual LUN is enabled.

**Figure 3–9 Communications LUN Controller Properties Tab**

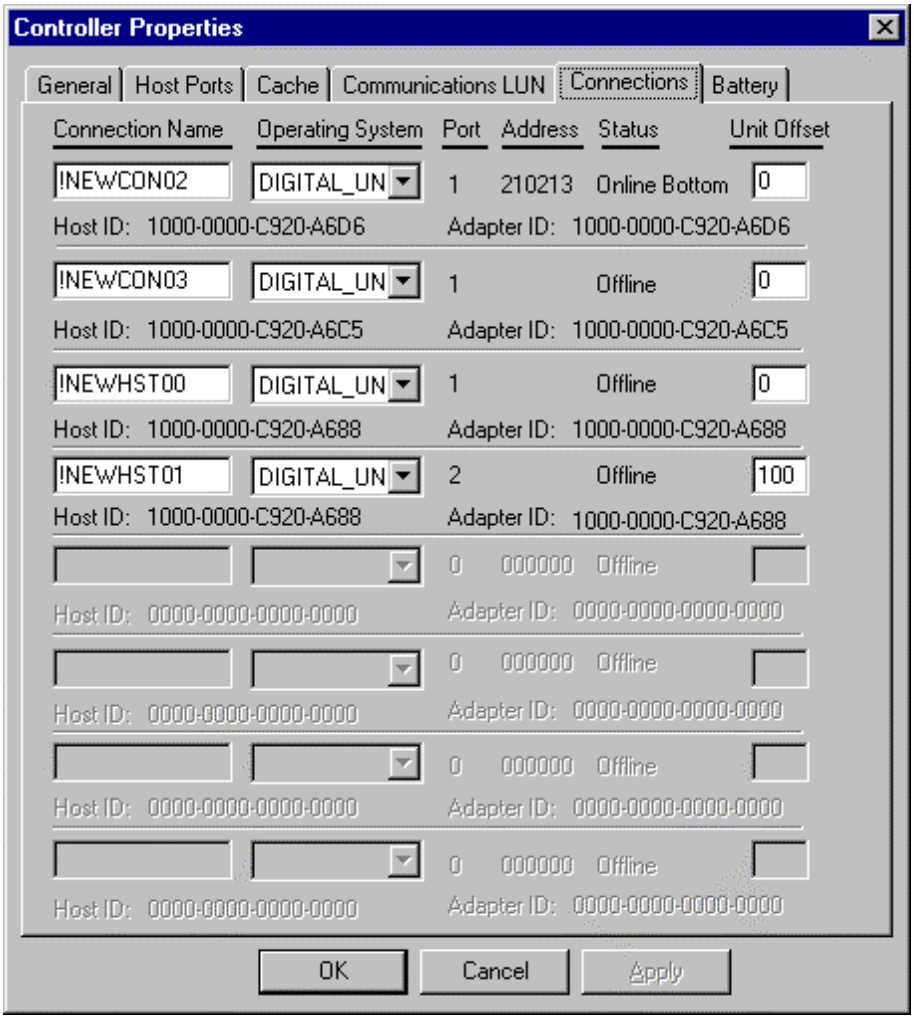


5. Click the *Connections* tab. (Figure 3-10).

   Confirm the following:

   - Use the pull-down menu to change the operating system to Sun.
   - Unit Offset should be 0 for Port 1
   - Unit Offset should be 100 for Port 2

**Figure 3–10 Connections Controller Properties Tab**

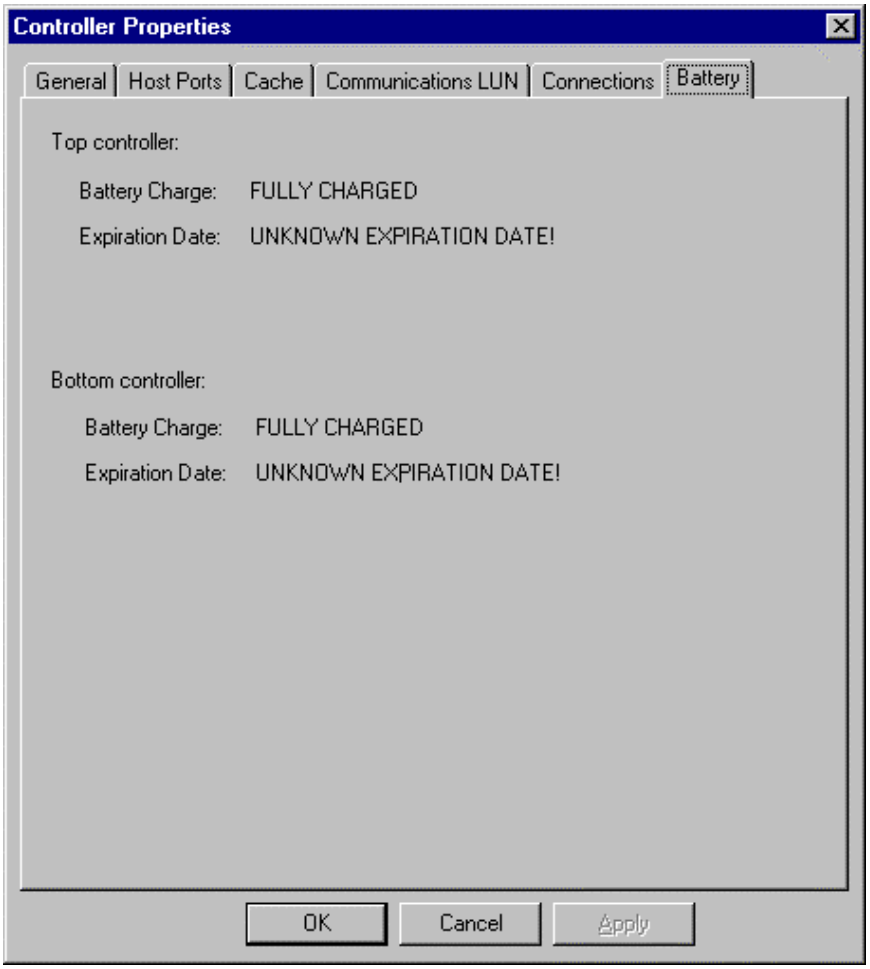


6. Click the *Battery* tab. (Figure 3–11)

**NOTE**

Confirm the that the battery is fully charged. If not, wait until it is fully charged, before using the system

**Figure 3–11 Battery Controller Properties Tab**



Click on the *Apply* button. Once all of the changes have been made, click the OK button to complete controller configuration. The storage window is now displayed. To begin using SWCC to create Virtual Disks, refer to the StorageWorks Command Console User's Guide, to the section entitled *Creating Your First Virtual Disk*.

## 3.6 Using the Navigation Window

The Navigation Window is a network management tool that you can use to create, monitor and configure a network of storage subsystems. The Navigation Window has its own Menu Bar. It can be moved and minimized. It can be sized by dragging on its corners and borders. You start building a network of RAID Array 8000 storage subsystems by adding Host Systems to the Navigation Window.

An example of a 1-host network is shown in (Figure 3–12). The host system, *large.shr.dec.com* is connected to a subsystem named, *large_hsg80*. The folders are shown expanded.

**Figure 3–12 Navigation Window Showing One Host**

### 3.6.1 Adding a Host System to the Navigation Window

To add a system:

1. From the **File** menu select *Add System.*
2. Use the *Add System* dialog box to add host systems to the Command Console Client. A system is any machine running a Command Console Agent. Enter the Domain Name Service (DNS) name or the Internet Protocol (IP) address in the "Host name or TCP/IP address text box" (shown in Figure 3–13) and click *Apply*. (Refer to the on-line **Help** menu (*Adding a System*) if you need more information about entering a system name
3. After you click the *Apply* button, once the system address is resolved, click *Close* on the Add System dialog box; Client adds an icon for *seneker.shr.dec.com* in the Navigation Window (Figure 3-14).

**Figure 3–13 Adding a Host System**

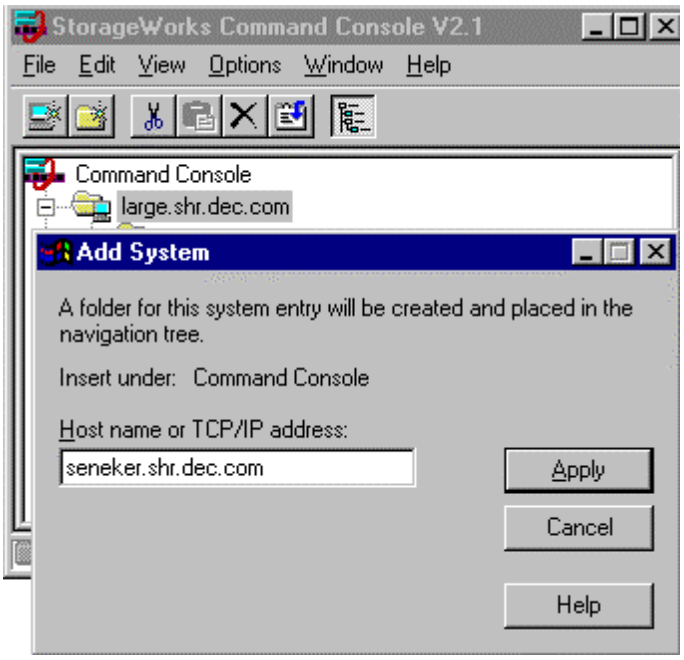


Figure 3-14 is an example of a 2-host network. The system names are *bithead.shr.dec.com* and *seneker.shr.com*. The folders are shown expanded. The switch management application or *Fibre Channel Network* and the Graphical User Interface, *Fabric Window* are configured from the host system, *seneker* .

**Figure 3–14 Seneker Added to the Navigation Window**


### 3.6.2 Deleting a Host System from the Navigation Window

To delete a system:

1. In the Navigation Window, click on and highlight the system that you want to delete.
2. From the **Edit** menu, select *Delete*.
3. System icon should disappear from Navigation Window.

### 3.6.3 Creating and Using New General Folders

Use folders to help organize your storage. The Client will automatically create folders when you add a subsystem. You may also create your own folders to help organize your subsystem. Folders can be dragged and dropped within the Navigation Window. In the following example we have created the folder *My Servers.*

1. From the File menu, choose *New Folde*r.
2. Name the new folder in the Navigation Window by double-clicking its label and entering a new name.

**Figure 3–15 Adding a Folder**

### 3.6.4 Renaming a Folder

1. Double-click on the folder name in the Navigation Window.
2. Enter the new name. Only general folders that you create can be renamed.

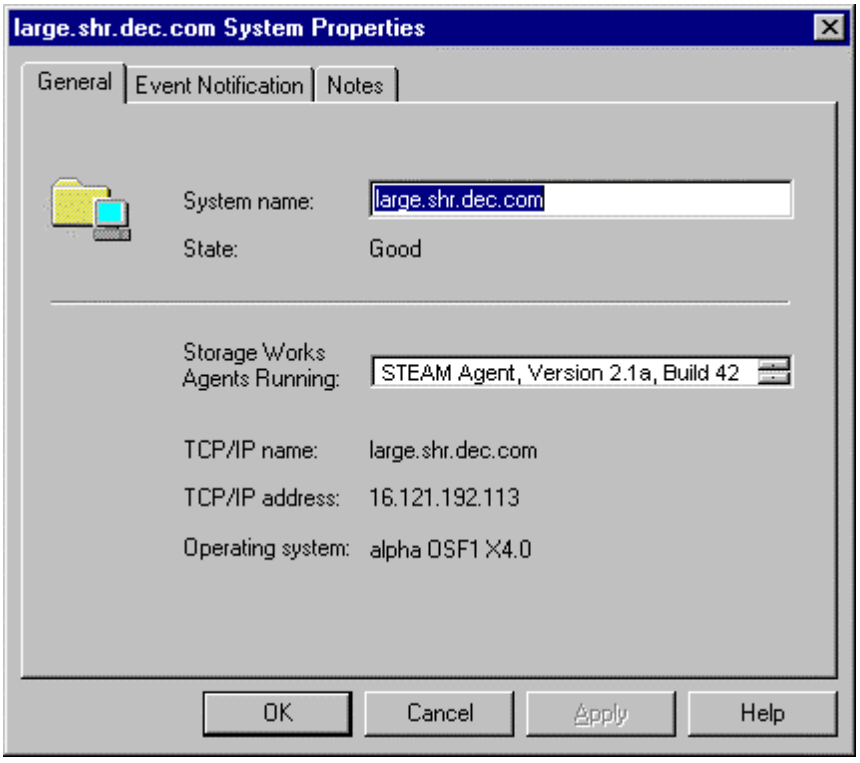
### 32.6.5 Viewing the Hierarchy in the Navigation Window

1. Click on a folder on the Navigation Window to display its contents.
2. Click the plus signs (+) to display more folders, systems, and controllers. To open a folder, double-click on it.

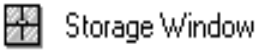
### 3.6.6 Viewing and Modifying System Folder Properties

1. Right-click on a system icon in the Navigation Window.
2. Click Properties from the shortcut menu to view the system folder properties.

**Figure 3–16 System Properties**



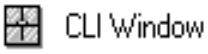
### 3.6.7 Opening a Storage Window

1. Connect to a storage subsystem. 

2. In the Navigation Window, double-click on a system folder.
3. Double-click the Storage Window icon to open a Storage Window.

### 3.6.8 Opening a CLI Window

Not all controllers support the Command Line Interpreter (CLI) window. If you are familiar with CLI syntax, you can enter commands into the command line area directly underneath the CLI Window Menu Bar. Error messages and system responses are displayed in the message area directly beneath the command line area.

1. Connect to a storage subsystem. 

2. In the Navigation Window, double-click on a system folder.
3. Double-click the CLI Window icon to open a CLI Window.

## 3.7 Choosing a Connection Method

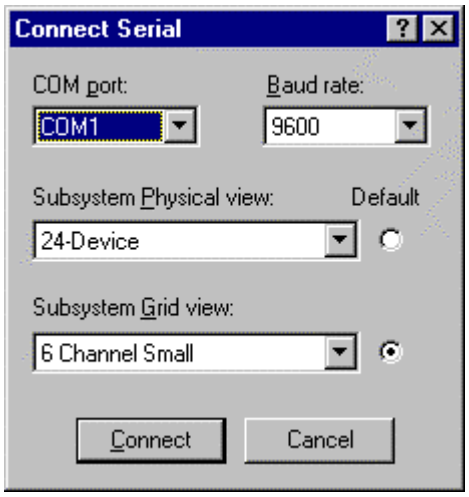
The Client offers three ways to connect to your storage subsystem: over the controller's serial port, the SCSI bus (Windows NT Only) or over a TCP/IP network. The serial port and SCSI bus connections are local connections and allow you to connect to only one storage subsystem at a time.

### 3.7.1 Establishing a Serial Connection

The simplest connection to a storage subsystem is a direct, cable connection from the Client's host system to one of the storage subsystem's serial ports. To establish a serial connection, connect a serial cable from one of the PC's COM ports to the storage subsystem's serial configuration port and proceed as follows:

1. From the Start menu, select Programs, Command Console 2.1, and then the *HSG80 Storage Window* applet.
2. When the Connection Selection dialog box displays, select the *Serial* option, then click OK to display the Connect Serial dialog box (Figure 3–17).
3. On the Serial Connection dialog box, from the drop-down menu, select the PC COM port your controller is connected to, a subsystem physical and grid view, and a baud rate. Next click the Connect button to display the Storage Window. When the Storage Window displays, you are connected to your storage subsystem.

**Figure 3–17 Connect Serial Dialog Box**




### 3.7.2 Establishing a SCSI Connection (Windows NT Hosts Only)

**NOTE**

If you are using the Solaris operating system, disregard this section.

If Client's Host Operating System is Windows NT, you can connect to, monitor and manage a storage subsystem over the SCSI bus.

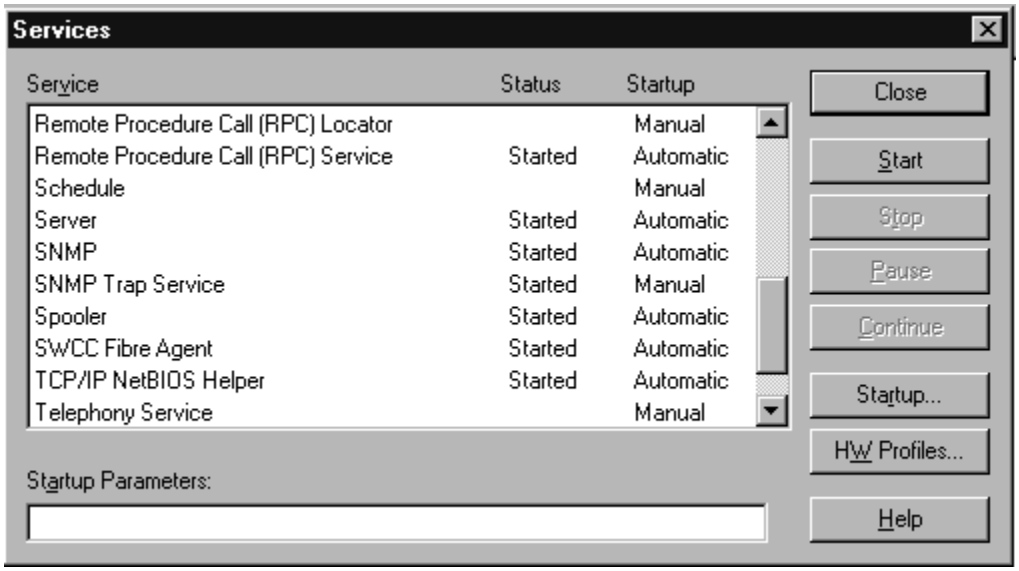
Before you can connect to your storage subsystem over the SCSI bus, at least one virtual disk must be configured on the storage subsystem. To create the first virtual disk you must use a serial connection to the subsystem. The procedure to create the first virtual disk is summarized as follows:

- Establish a serial connection to your storage subsystem using the *Establishing a Serial Connection* procedure provided in this User's Guide.
- Create a virtual disk. Refer to Section 3.8, *Creating a Virtual Disk*.
- Reboot your system.
- Start the Disk Administrator, create the partition and assign the drive letter for the virtual disk.

To connect over the SCSI bus proceed as follows:

1. If Command Console Agent is running, stop the Agent. On the Windows NT host computer, from the *Start* menu select *Settings*, *Control Panel*; then double-click on the *Services* Icon. Select *Steam* from the *Services* list and click the *Stop* button.

**Figure 3–18 Services Dialog Box**




2. Select Command Console from the Start menu, then HSZ StorageWindow.
3. When the connection dialog box displays, select the SCSI option.
4. When the SCSI Setup dialog box appears (Figure 3–19), select a subsystem drive; then click Connect. After a brief period of time, Command Console will connect to the storage system and display the Storage Window.

**Figure 3–19 SCSI Setup Dialog Box**


## 3.8 Creating a Virtual Disk

Before you can create a virtual disk, you must select a host system, open a Storage Window, set the Controller parameters, and make your physical devices known to your Controller.

### 3.8.1 Add Physical Devices to Your Configuration

You must make the physical devices known to the Controller before you can create virtual disks from them. To add physical devices to the configuration:

1. In the **Storage** menu, select *Device*, then *Add*.
2. Enter your password if required.
3. When Client prompts you to insert the physical devices (Figure 3–20), insert them in the storage subsystem; then click the **OK** button. Client adds your devices to your configuration and refreshes the Storage Window.

**Figure 3–20 Insert Devices Prompt**

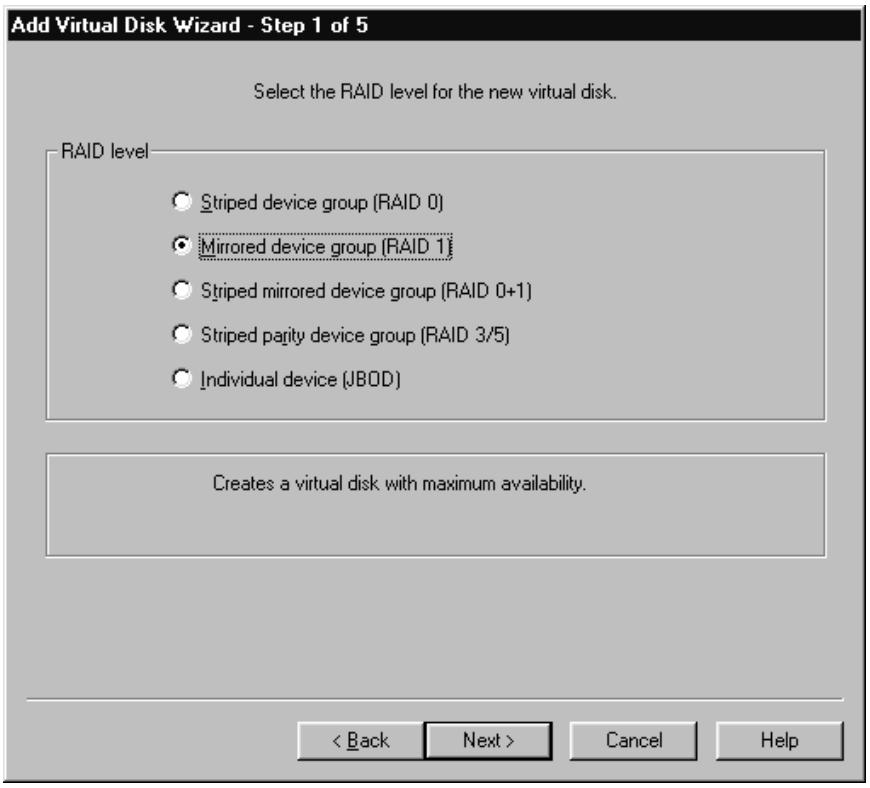

Once you have added the physical devices to the storage configuration you can use them to create a number of different types of logical storage units called virtual disks. You can create:

- Single-device virtual disks JBODs (Just a Bunch of Disks)
- Striped virtual disks (RAID 0)
- Mirrored virtual disks (RAID 1)
- Striped mirrored virtual disks (RAID 0+ 1)
- Striped virtual disks with parity (RAID 3/5)

### 3.8.2 Create a Virtual Disk

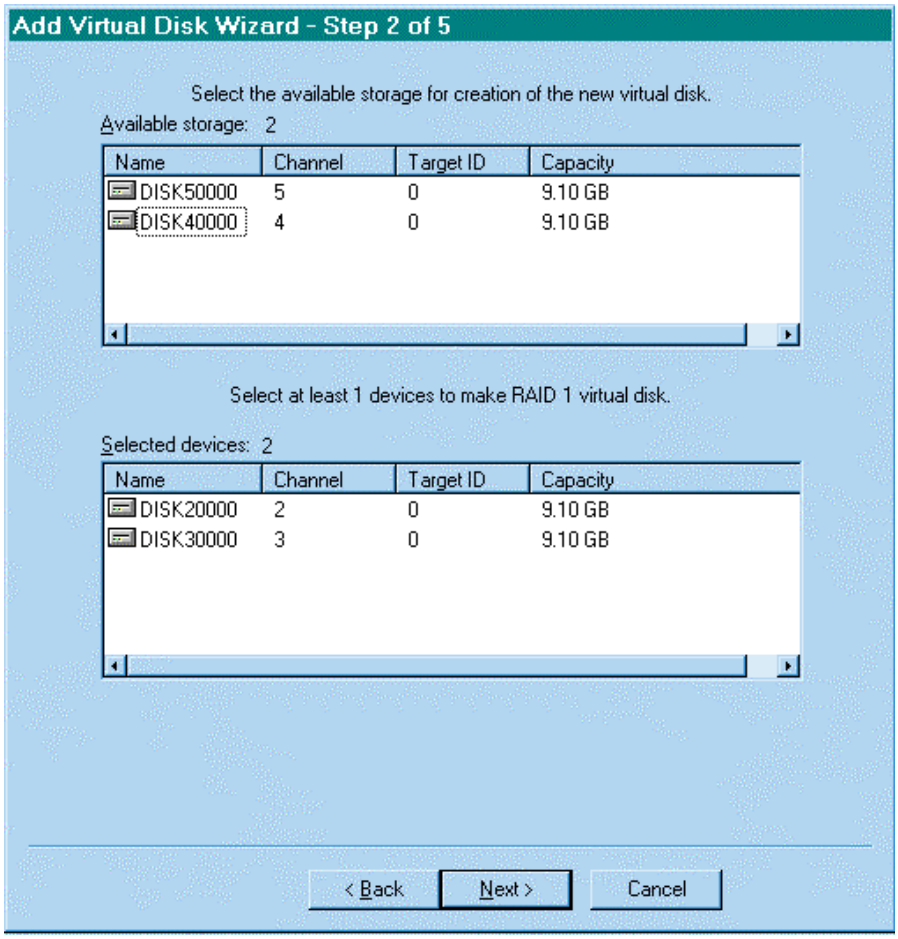
1. In the **Storage** menu, select *Add Virtual Disk*. Virtual Disk Wizard's Step 1 (Figure 3-21) appears. There are five steps in total.
2. Select the RAID level you want to create, then click *Next*.

**Figure 3–21 Add Virtual Disk Wizard Step 1**



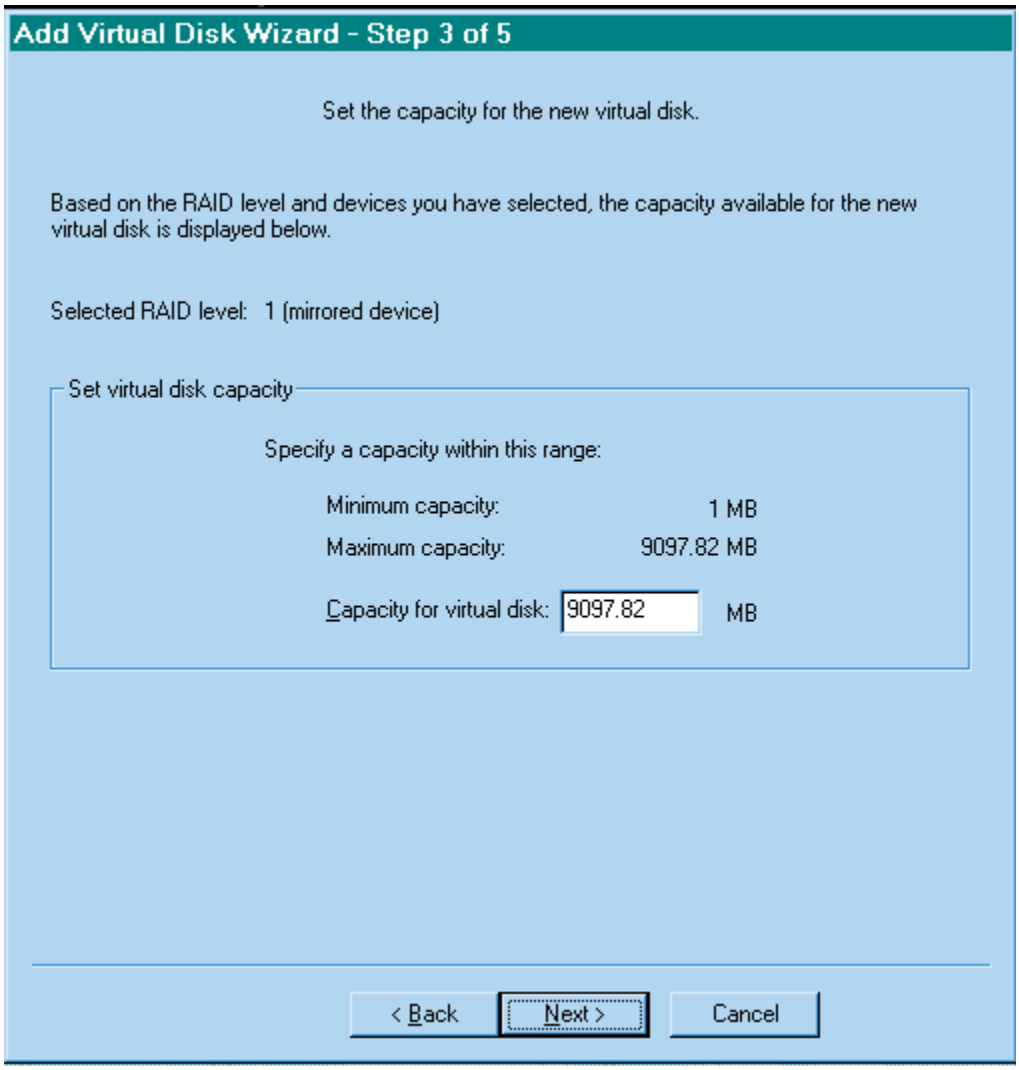

3. Select the devices you want to include in the virtual disk from a list of available storage devices. You select devices by double-clicking them in the *Available Storage* window in the dialog box. As you click them, they move to the *Selected Devices* window in the dialog box. They are also shown as selected in the Device Window pane of the Storage Window.

**Figure 3–22 Add Virtual Disk Wizard Step 2**



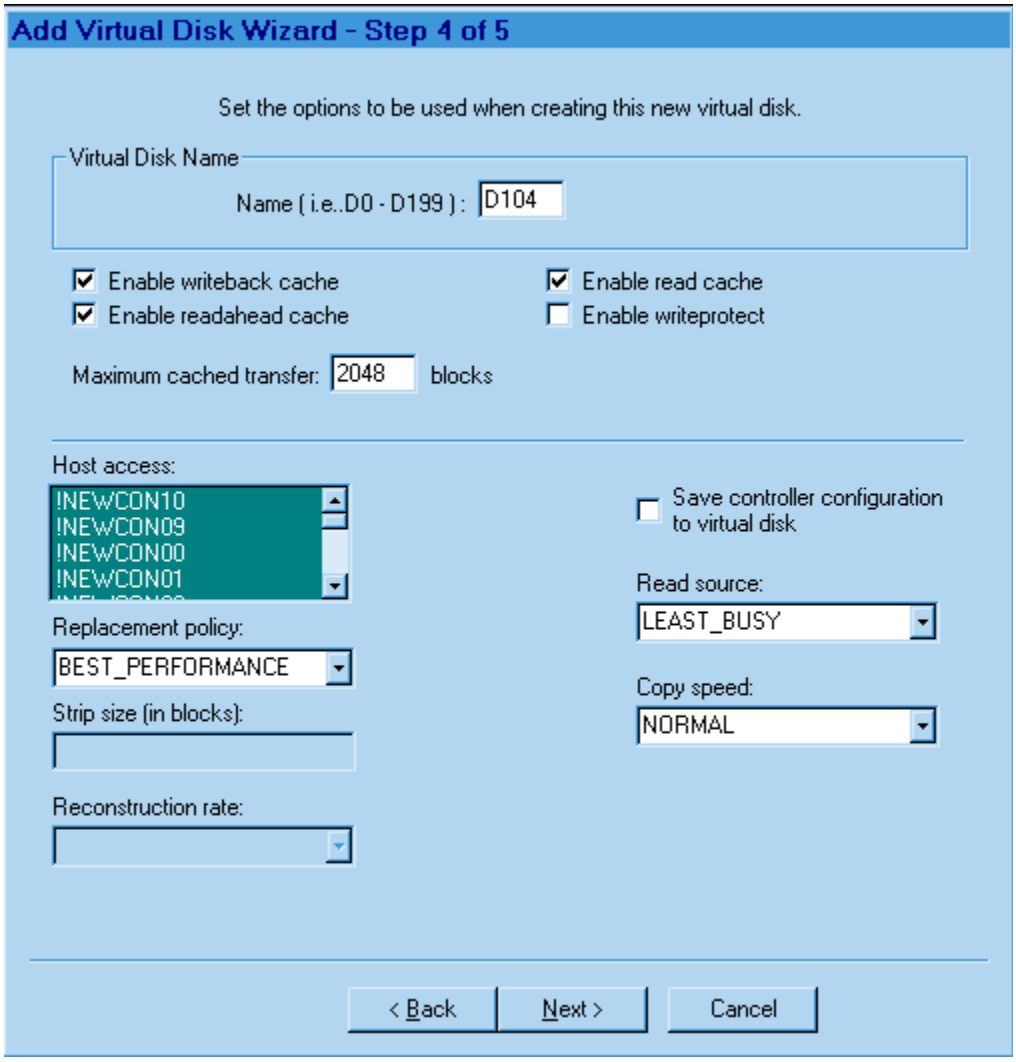

4. Select the capacity of the virtual disk, then click *Next*. The wizard offers you the option of using only a portion of the capacity of the devices you have selected for your new virtual disk. It displays the total, available capacity of the devices you have selected in the capacity box. Enter the size of the virtual disk you want to create in the box.

**Figure 3–23 Add Virtual Disk Wizard Step 3**



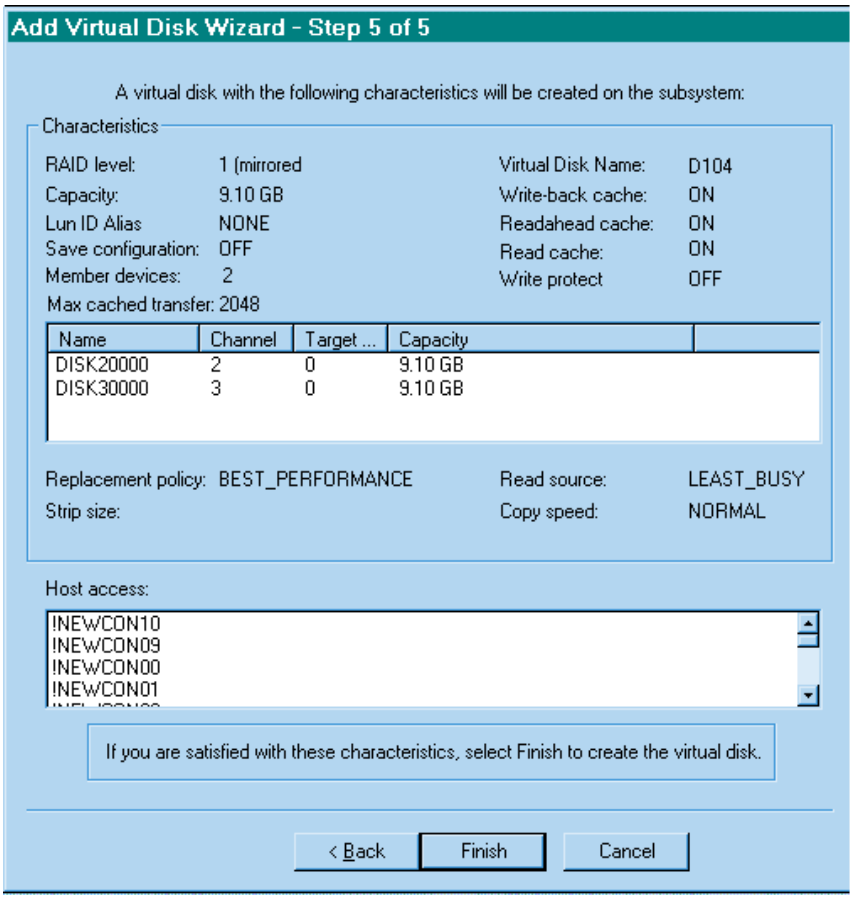

5. Specify the target ID and Logical Unit Number (LUN), operating parameters, and options of your new virtual disk, then click *Next.*

**Figure 3–24 Add Virtual Disk Wizard Step 4**




6. The final Virtual Disk Wizard window, Step 5, recaps the choices you have made in steps 1 through 4. If you are satisfied with your choices, click *Finish*. Otherwise, use the *Back* button to go return to the previous step and make the necessary changes.

**Figure 3–25 Add Virtual Disk Wizard Step 5**




### 3.8.3 Delete a Virtual Disk

To delete a virtual disk:

1. Select its *icon* by clicking on it in the Virtual Disk Window.
2. Choose *Storage* from the Menu Bar; then *Virtual Disk*, then *Delete.*

### 3.8.4 Saving the Configuration

1. From the **Storage** menu, click *Controller Configuration,* then *Save.*
2. The Save dialog box appears, enter the desired location and file name. Your configuration is saved.

# 4

# *Creating Your Storage Configuration with the CLI*

*This chapter contains instructions for creating an initial storage configuration using the Command Line Interpreter (CLI). It briefly describes the CLI and how to access it. The configuration steps include: adding devices; creating and initializing RAIDsets, stripesets, mirrorsets, and striped mirrorsets; identifying a storageset as a unit to the host; and verifying and recording the final configuration.*

**NOTE**

To create your storage configuration using the StorageWorks Command Console, refer to The Command Console User's Guide.

Once you complete the physical setup of the RAID Array, configure the devices in your subsystem into storagesets.

To configure the devices in your subsystem into storagesets, you need to:

- Plan your configuration (Refer to Appendix A)
- Add disks to the controller
- Create storagesets
- Save the configuration
- Record the configuration

## 4.1 Configuration Guidelines

Use the following guidelines to configure the HSG80 controller and your host system to optimize system performance.

### 4.1.1 Controller Device Configuration Guidelines

- The enclosure has six device ports (SCSI buses). Evenly distribute disk devices across the six separate device ports. This permits parallel activities on the controller's available device ports to the attached drives.
- Avoid configuring multiple mirrorsets with the first member being on the same device port. Configure multiple mirrorsets similar to the following example:

```
ADD MIRRORSET MIRR_1 DISK12000
ADD MIRRORSET MIRR_2 DISK20100 DISK10100 DISK20000
```

### 4.1.2 Host System Configuration Guidelines

You need to assign a host logical unit number to each storageset or single disk unit that you want your host to know about in your subsystem. The host uses these numbers to indicate the source or destination for every I/O request it sends to the controller.

Each logical unit number contains the following:

- A letter that indicates the kind of devices in the storage unit. For example, D for disk drives
- A number from 0-99 or 100-199

Each HSG80 controller has two host ports, Port 1 and Port 2, as shown in the following figures. Unit numbers D0-D99 are assigned to Host Port 1, unit numbers D100-D199 are assigned to Host Port 2. You can specify a maximum of 64 host logical units per host port, for a total of 128 host units when access is from two Fibre Channel host adapter ports.

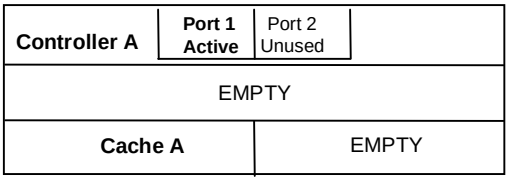
**NOTE**

Tru64 UNIX Version 4.x limits the total 2 targets with 8 LUNs per target. This will give you a total of 16 LUNs.

**NOTE**

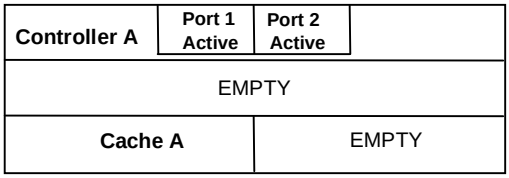
Always assign all partitions of a storageset to the same host port (do not split partitioned storagesets across host ports).

**Figure 4–1 Single Controller/Single Host**



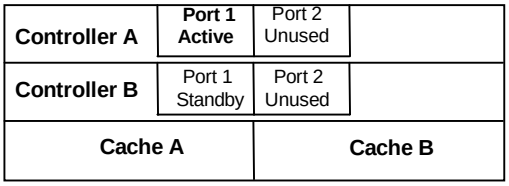

- For single HSG80 controller configurations connected to a single host you can configure up to 64 host logical units on Controller A - Host Port 1. Valid unit numbers are D0-D99. Controller A - Host Port 2 is unused.

**Figure 4–2 Single Controller/Two Hosts**



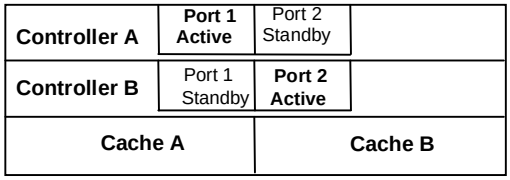

- For single HSG80 controller configurations connected to two host systems, for one host you can configure up to 64 host logical units on Controller A - Host Port 1. Valid unit numbers are D0-D99. For the other host, you can configure up to 64 host logical units on Controller A - Host Port 2. Valid unit numbers are D100-D199.

**Figure 4–3 Dual Controller/Single Host**




- For dual-redundant HSG80 controller configurations connected to a single host you can configure up to 64 host logical units on Controller A - Host Port 1. Valid unit numbers are D0-D99. Controller B - Host Port 1 is automatically configured as a standby port for these same 64 units. Controller A - Host Port 2 and Controller B - Host Port 2 are unused.

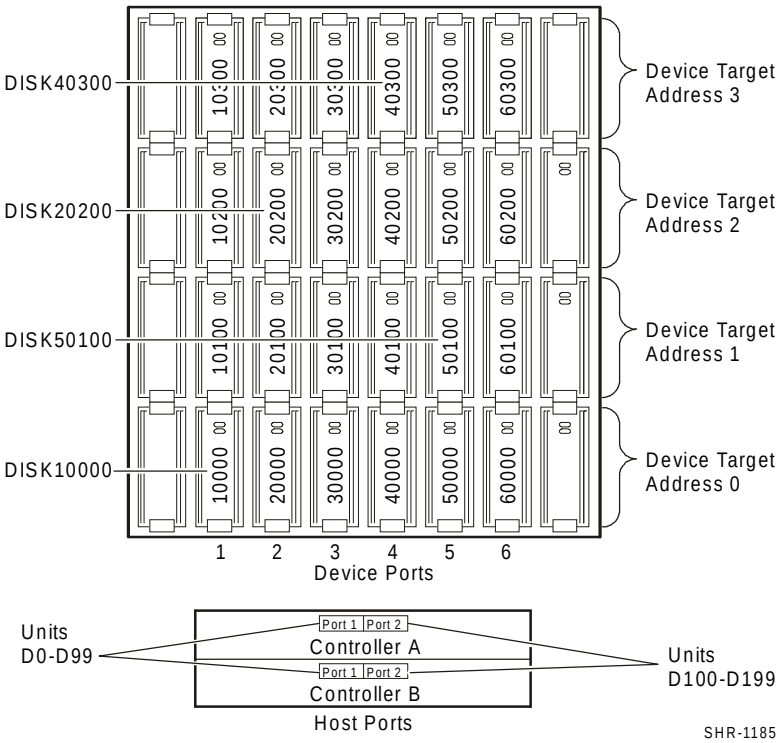
**Figure 4–4 Dual Controllers/Two Hosts**




- For dual-redundant HSG80 controller configurations connected to two host systems, for one host you can configure up to 64 host logical units on Controller A-Host Port 1. Valid unit numbers are D0-D99. Controller B - Host Port 1 is automatically configured as a standby port for these same 64 units.

  For the other host you can configure up to 64 host logical units on Controller B-Host Port 2. Valid unit numbers are D100-D199. Controller A - Host Port 2 is automatically configured as a standby port for these same 64 units.

**Figure 4–5 Mapping of Device Ports/Targets and Host Ports**



## 4.2 Accessing the Command Line Interpreter (CLI)

The CLI is a command line user interface to the HSG80 controller. It provides a series of commands for you to create a configuration for the subsystem through the controller's firmware.

This chapter describes only the CLI commands required to create an initial configuration on the controller.

See the *COMPAQ StorageWorks HSG80 Array Controller ACS Version 8.4 User's Guide* for detailed descriptions of all CLI commands.

You must make a serial connection to the HSG80 controller to access the CLI.

The three methods of accomplishing the connection are:

- Using an ASCII terminal
- Using a PC running a communications program
- Using a SUN system with the "tip" command

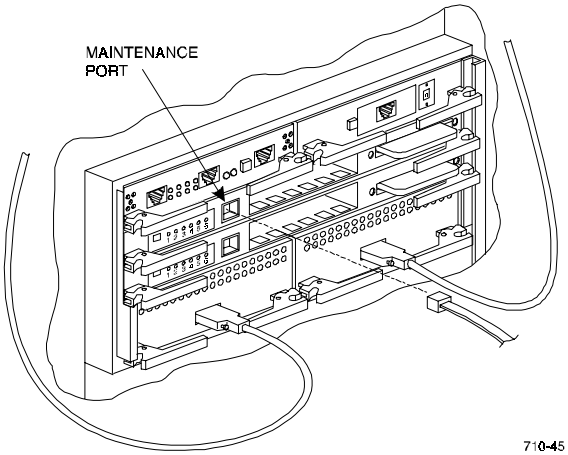
See *Command Line Interpreter* in the *HSG80 RAID Array Controller CLI Reference Manual* for detailed descriptions of all CLI commands.

### 4.2.1 Connecting the Cable to a PC or ASCII Terminal

To connect a maintenance terminal or PC to a HSG80 Controller follow these steps:

1. Locate the connecting cable that came with the RA8000/ESA12000 subsystem. It has an RJ12 connector (similar to standard telephone plug) on one end and a 9-pin serial connector on the other end.
2. Plug the serial connector into the 9-pin serial port/com port 1 of the PC. If a 9-pin serial port is not available on a PC, use the 9-pin to 25-pin adapter (P/N: 12-45238-01) supplied with your subsystem.
3. Plug the RJ12 connector from the PC or maintenance terminal into the maintenance port on the HSG80 Controller (see Figure 3–6).
4. Note which serial port on the PC you use; you will need that information if using a communications program.

**Figure 4–6 Making a Serial Connection to the HSG80 Controller**




### 4.2.2 Establishing Connection with a Maintenance Terminal

To establish a connection between a maintenance terminal and the controller, follow these steps:

1. After connecting the maintenance terminal cable to the controller, press the Enter key. The CLI prompt appears in the window similar to the following:

   ```
   HSG80 >
   ```

2. To view the status of the controller, type:

   ```
   HSG80 > SHOW THIS_CONTROLLER FULL
   ```

   The controller displays information similar to the following example: (dual-redundant configuration shown)

```
Controller:
        HSG80 ZG74100091 Software V84G-0, Hardware E01
        NODE_ID          = 5000-1FE1-FF00-0170
         Allocation_Class=0
         SCSI_Version=2
        Configured for dual-redundancy with ZG74100121
            In dual-redundant configuration
        Device Port SCSI address 7
        Time: 28-SEP-1998 21:03:32
Host PORT_1:
        Reported PORT_ID = 5000-1FE1-FF00-0171
        PORT_1_PROFILE   = PLDA
        PORT_1_TOPOLOGY  = Fabric (Link up)
        PORT_1_AL_PA     = 71 (71 negotiated)
Host PORT_2:
        Reported PORT_ID = 5000-1FE1-FF00-0172
        PORT_2_PROFILE   = PLDA
        PORT_2_TOPOLOGY  = Fabric (Standby)
        PORT_2_AL_PA     = 72 (72 negotiated)
```

```
Cache:
        128 megabyte write cache, version 0012
        Cache is GOOD
        Battery is GOOD
        No unflushed data in cache
        CACHE_FLUSH_TIMER = DEFAULT (10 seconds)
Mirrored Cache:
         Not enabled
Battery:
         FULLY CHARGED
         Expires:
         NOCACHE_UPS
Extended information:
        Terminal speed 9600 baud, eight bit, no parity, 1 stop bit
        Operation control: 00000000  Security state code: 21429
        Configuration backup disabled
```

**NOTE**

Verify that the output of the "SHOW THIS" command from your subsystem is similar to that shown. If the controller presents a NODE_ID of all zeros (0000-0000-0000-0000), or the appropriate host port does not report a LOOP_UP condition, refer to the HSG80 User's Guide controller configuration chapter for more information.

## 4.3 Adding Disks to the Configuration

The CONFIG utility locates and adds disks to the controller. Run the CONFIG utility whenever you add new disks to the controller. Enter the following command to start the configuration utility. The disk numbers will correspond to the disk locations for your subsystem.

```
HSG80 > RUN CONFIG
```

The controller responds with a display similar to that shown below:

```
CONFIG LOCAL PROGRAM INVOKED
CONFIG IS BUILDING ITS TABLES AND DETERMINING WHAT DEVICES EXIST ON
THE SUBSYSTEM. PLEASE BE PATIENT.
ADD DISK12000 1 0 0
ADD DISK10100 1 1 0
ADD DISK10200 1 2 0
ADD DISK20000 2 0 0
ADD DISK20100 2 1 0
ADD DISK20200 2 2 0
ADD DISK30000 3 0 0
ADD DISK30100 3 1 0
ADD DISK30200 3 2 0
ADD DISK40000 4 0 0
ADD DISK40100 4 1 0
ADD DISK40200 4 2 0
ADD DISK40300 4 3 0
ADD DISK50000 5 0 0
ADD DISK50100 5 1 0
ADD DISK50200 5 2 0
ADD DISK50300 5 3 0
ADD DISK60000 6 0 0
ADD DISK60100 6 1 0
ADD DISK60200 6 2 0
ADD DISK60300 6 3 0
CONFIG – NORMAL TERMINATION
```

In this example, the controller has located 21 new disks. The 5 digit number associated with each disk corresponds to a one-digit Device Port Number, a two-digit Target Number and Controller Logical Unit Number. The Controller Logical Unit Number will always be 00. DISK40000, in this example, corresponds to the disk located on Device Port 4, on controller Target 0, and Controller Logical Unit 0. DISK50100 corresponds to the disk located on Device Port 5, controller Target 1, and Controller Logical Unit 0. Figure 3-5 shows the mapping of Device Ports, Targets and Host Ports.

## 4.4 Creating a RAIDset

RAIDsets stripe user data over multiple drives and calculate parity information for data redundancy. Create RAIDsets to use redundant stripesets in your array. RAIDsets must have at least three members and can have as many as fourteen. This example creates two three member RAIDsets using the ADD RAIDSET command.

```
HSG80 > ADD RAIDSET DVGRPR0 DISK12000 DISK20000 DISK30000
HSG80 > ADD RAIDSET DVGRPR1 DISK40000 DISK50000 DISK60000
```

In this example, "DVGRPR0" and "DVGRPR1" are the names of the RAIDsets, and they are followed by a list of disks to be included in each RAIDset. The names of the RAID sets are user selectable. Performance of your RAIDsets will be optimized if each RAIDset includes disks from different ports as shown in the example.

### 4.4.1 Initializing a RAIDset

Prior to putting a RAIDset(s) into service as a logical unit, you must initialize it. The INITIALIZE command copies controller metadata onto a small amount of disk space available on the RAIDset and makes this space inaccessible to the host.

When you initialize a RAIDset, you can specify a chunksize. A chunksize is the number of blocks of data that is transferred at one time. By using the default chunksize, the controller will optimize the chunksize by selecting a number equal to the number of blocks in one track of disk data. We recommend using the default chunksize.

```
HSG80 > INITIALIZE DVGRPR0 CHUNKSIZE=DEFAULT
HSG80 > INITIALIZE DVGRPR1 CHUNKSIZE=DEFAULT
```

### 4.4.2 Adding a RAIDset as a Logical Unit

To make a RAIDset available to the host computer, you must identify it as a host logical unit. For single or dual controllers on a single host, the unit numbers may range from D0 through D99 with a maximum of 64 units. For dual controllers/two hosts, the unit numbers may range from D0 through D99 for the first host and from D100 through D199 for the second host with a maximum of 64 units per host adapter. Add units by using the ADD UNIT command.

```
HSG80 > ADD UNIT D1 DVGRPR0
HSG80 > ADD UNIT D2 DVGRPR1
```

This example uses D1 and D2, as the first and second units identified on the controller.

### 4.4.3 Setting Writeback Cache

This feature is enabled by default; but if it is necessary, a single CLI command enables that feature for the entire RAIDset:

```
HSG80 > SET D1 WRITEBACK_CACHE
HSG80 > SET D2 WRITEBACK_CACHE
```

Where D1 and D2 represent the host logical units of the RAIDsets described above.

### 4.4.4 Setting Read Ahead Cache

This feature is enabled by default; but if it is necessary, a single CLI command enables that feature for the entire RAIDset:

```
HSG80 > SET D1 READAHEAD_CACHE
```

Where D1 represent the host logical unit of the RAIDsets described above.

## 4.5 Creating a Stripeset

Use stripesets to stripe data across multiple disks. Striping data across multiple disks increases I/O performance compared with the performance of a single disk. Stripesets must have at least two members and can have as many as fourteen. All members must be single disks. This example creates a three member stripeset using the ADD STRIPESET command.

```
HSG80 > ADD STRIPESET DVGRPS0 DISK10100 DISK20100 DISK30100
```

In this example, "DVGRPS0" is the name of the stripeset, and it is followed by a list of the disks to be included in the stripeset. The names of the stripesets are user selectable. Performance of your stripesets will be optimized if each stripeset includes disks from different device ports as shown in the example.

### 4.5.1 Initializing a Stripeset

Prior to putting a stripeset into service as a logical unit, you must initialize it. The INITIALIZE command copies controller metadata onto a small amount of disk space available on the stripeset and makes this space inaccessible to the host.

When you initialize a stripeset, you can specify a chunksize. A chunksize is the number of blocks of data that is transferred at one time. By using the default chunksize, the controller will optimize the chunksize by selecting a number equal to the number of blocks in one track of disk data. We recommend using the default chunksize.

```
HSG80 > INITIALIZE DVGRPS0 CHUNKSIZE=DEFAULT
```

### 4.5.2 Adding a Stripeset as a Logical Unit

To make a stripeset available to the host computer, you must identify it as a host logical unit. For single or dual controllers on a single host, the unit numbers may range from D0 through D99 with a maximum of 64 units. For dual controllers/two hosts, the unit numbers may range from D0 through D99 for the first host and from D100 through D199 for the second host with a maximum of 64 units per host adapter. Add units by using the ADD UNIT command.

```
HSG80 > ADD UNIT D3 DVGRPS0
```

This example uses D3, since the stripeset is the third unit identified on the controller.

### 4.5.3 Setting Writeback Cache

This feature is enabled by default; but if it is necessary, a single CLI command enables that feature for the entire stripeset:

```
HSG80 > SET D3 WRITEBACK_CACHE
```

Where D3 represents the host logical unit of the stripeset described above.

## 4.6 Creating a Mirrorset

Create mirrorsets to increase data availability and achieve data redundancy by maintaining at least two drives that have exactly the same data. Mirrorsets must have at least two members, and can have as many as six. This example creates a two member mirrorset using the ADD MIRRORSET command.

```
HSG80 > ADD MIRRORSET DVGRPM0 DISK10200 DISK20200
```

In this example, DVGRPMO is the name of the mirrorset, and it is followed by a list of the disks to be included in the mirrorset. The names of the mirrorsets are user selectable. Performance of your mirror sets will be optimized if each mirrorset includes disks from different ports as shown in the example.

### 4.6.1 Initializing a Mirrorset

Prior to putting a mirrorset into service as a logical unit, you must initialize it. The INITIALIZE command copies controller metadata onto a small amount of disk space available on the mirrorset and makes this space inaccessible to the host.

```
HSG80 > INITIALIZE DVGRPM0
```

### 4.6.2 Adding a Mirrorset as a Logical Unit

To make a mirrorset available to the host computer, you must identify it as a host logical unit. For single or dual controllers on a single host, the unit numbers may range from D0 through D99 with a maximum of 64 units. For dual controllers/two hosts, the unit numbers may range from D0 through D99 for the first host and from D100 through D199 for the second host with a maximum of 64 units per host adapter. Add units by using the ADD UNIT command.

```
HSG80 > ADD UNIT D4 DVGRPM0
```

### 4.6.3 Setting Writeback Cache

This feature is enabled by default; but if it is necessary, a single CLI command enables that feature for the entire mirrorset:

```
HSG80 > SET D4 WRITEBACK_CACHE
```

Where D4 represents the host logical unit of the mirrorset described above.

## 4.7 Creating a Striped Mirrorset

Create a striped mirrorset to achieve high I/O performance and maximum data availability. striped mirrorsets must have at least two mirrorset members, and can have as many as fourteen. All members must be mirrorsets. To create striped mirrorsets, you first create mirrorsets and then you create stripesets with those mirrorsets.

### 4.7.1 Creating Mirrorsets

These examples create 2, two member mirrorsets for the striped mirrorset.

```
HSG80 > ADD MIRRORSET MIRR_0 DISK30200 DISK40200
HSG80 > ADD MIRRORSET MIRR_1 DISK50200 DISK60200
```

In these examples, MIRR_0 and MIRR_1 are the names of the mirrorsets. Each is followed by the list of disks to be included in it.

### 4.7.2 Striping the Mirrorsets

Striped mirrorsets must have at least two members, and can have as many as fourteen. This example uses the ADD STRIPESET command to create a two member stripeset with the mirrorsets that you just created.

```
HSG80 > ADD STRIPESET DVGRPSM0 MIRR_0 MIRR_1
```

In this example, DVGRPSM0 is the name of the striped mirrorset, and it is followed by a list of mirrorsets to include in it. The name of the stripeset is user selectable. Performance of your striped mirrorset will be optimized if each mirrorset includes disks from different device ports as shown the example.

### 4.7.3 Initializing the Striped Mirrorset

Prior to putting a striped mirrorset(s) into service as a logical unit, you must initialize it. The INITIALIZE command copies controller metadata onto a small amount of disk space available on the striped mirrorset and makes this space inaccessible to the host.

When you initialize a striped mirrorset you can specify a chunksize. A chunksize is the number of blocks of data that is transferred at one time. By using the default chunksize, the controller will optimize the chunksize by selecting a number equal to the number of blocks in one track of disk data. We recommend using the default chunksize.

```
HSG80 > INITIALIZE DVGRPSM0 CHUNKSIZE=DEFAULT
```

### 4.7.4 Adding a Striped Mirrorset as a Logical Unit

To make a striped mirrorset available to the host computer, you must identify it as a host logical unit. For single or dual controllers on a single host, the unit numbers may range from D0 through D99 with a maximum of 64 units. For dual controllers/two hosts, the unit numbers may range from D0 through D99 for the first host and from D100 through D199 for the second host with a maximum of 64 units per host adapter. Add units by using the ADD UNIT command.

```
HSG80 > ADD UNIT D5 DVGRPSM0
```

### 4.7.5 Setting Writeback Cache

This feature is enabled by default; but if it is necessary, a single CLI command enables that feature for the entire striped mirrorset:

```
HSG80 > SET D5 WRITEBACK_CACHE
```

Where D5 represents the host logical units of the striped mirrorset described above.

## 4.8 Adding Individual Disks as Logical Units

To use an individual disk in a RA8000 Fibre Channel subsystem, you must initialize it and then add it as a logical unit.

### 4.8.1 Initializing Individual Disks

Prior to putting an individual disk into service as a logical unit, you must initialize it. The INITIALIZE command copies controller metadata onto a small amount of disk space available on the disk and makes this space inaccessible to the host.

```
HSG80 > INITIALIZE DISK40100
HSG80 > INITIALIZE DISK50100
```

### 4.8.2 Adding as Logical Units

To make an individual disk available to the host computer, you must identify it as a host logical unit. For single or dual controllers on a single host, the unit numbers may range from D0 through D99 with a maximum of 64 units. For dual controllers/two hosts, the unit numbers may range from D0 through D99 for the first host and from D100 through D199 for the second host with a maximum of 64 units per host adapter. Add units by using the ADD UNIT command.

```
HSG80 > ADD UNIT D4 DISK40100
HSG80 > ADD UNIT D5 DISK50100
```

## 4.9 Adding Devices to the Spareset

It is advisable to add devices to the spareset to create a pool of devices for the controller to use as replacements for devices in a RAIDset, mirrorset or striped mirrorset that fail. If no spareset exists, these redundant types of storagesets will run “reduced,” and you should replace the disabled disk as soon as possible. To create the spareset, identify the drive(s) using the ADD SPARESET command.

```
HSG80 > ADD SPARESET DISK60100
```

In this example, DISK60100 was identified to the controller as a spareset.

**NOTE**

Please keep in mind that disks in the spareset must have at least the same storage capacity as those disks that they might replace.

## 4.10 Saving Copies of the Configuration

Use the following INITIALIZE command to save a copy of the entire controller configuration on a device or storageset in the subsystem. Save a copy of the controller configuration on a device or storageset so that in the event of a controller failure, you will not need to create a new controller configuration.

The controller automatically updates the saved copy of the configuration whenever the configuration changes.

We recommend keeping a copy of the configuration on at least two devices or storagesets.

To save a copy of the configuration on disk, use the INITIALIZE command as follows:

```
HSG80 > INITIALIZE DISK12000 SAVE_CONFIGURATION
```

The controller places a copy of the configuration onto the specified device or storageset and automatically updates this saved copy whenever the configuration changes. To ensure availability of a copy of the configuration, save the configuration on at least two devices.

**NOTE**

The save configuration option extends the metadata on the storageset by approximately 500 blocks. The remaining user data area can be used by the host operating system as it would any other storageset after a logical unit has been created. ALL disks in the storageset will receive a copy of the configuration data.

**CAUTION**

If user data already exists on a storageset, do NOT reinitialize it with the *save configuration* option, as this will change the site and position of the user data on the storageset. Compaq recommends backing up user data prior to reinitializing any storageset.

## 4.11 Recording your Configuration

You have now completed all the steps required to create an initial configuration for your controller. In the following steps, you should verify and record your configuration for future reference. Additional worksheets are provided in this chapter for recording future new or modified configurations.

First, verify the Logical Units you have configured:

```
HSG80 > SHOW UNITS
```

The controller responds with a display similar to that shown below:

```
LUN         USES
D1          DVGRPR0
D2          DVGRPR1
D3          DVGRPS0
D4          DISK40100
D5          DISK50100
```

Record the information in the following table:

Date_____

| LUN | Uses |
|---|---|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

Next, verify the storagesets you have configured:

```
HSG80 > SHOW STORAGESETS
```

The controller responds with a display similar to that shown below:

```
Name            StorageSet        Uses          Used by
DVGRPS0         stripeset         DISK10100     D3
                                  DISK20100
                                  DISK30100
DVGRPSM0        stripeset         MIRR_0        D5
                                  MIRR_1
DVGRPM0         mirrorset         DISK10200     D4
                                  DISK20200
MIRR_0          mirrorset         DISK30200     DVGRPSM0
                                  DISK40200
MIRR_1          mirrorset         DISK50200     DVGRPSM0
                                  DISK60200
DVGRPR0         raidset           DISK12000     D1
                                  DISK20000
                                  DISK30000
DVGRPR1         raidset           DISK40000     D2
                                  DISK50000
                                  DISK60000
SPARESET        spareset          DISK60100
FAILEDSET       failedset
```

Individual devices are not displayed in this report. To display individual devices, enter the following:

```
HSG80 > SHOW DEVICES
```

Record the above information in the following table. In the event of a controller failure, the information that is recorded here will assist you in reconstruction of the storageset on your RA8000 Fibre Channel subsystem.

| Date_____ | | | |
|---|---|---|---|
| Name | StorageSet | Uses | Used by |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# A

# *Planning Your Storage Configuration*

*This appendix describes the RAID configuration options and RAID concepts which you need to know to create your storage configuration.*

## A.1 Planning Your Configuration

RAID stands for Redundant Array of Independent Disks. It is a way of configuring multiple physical disk drives to achieve high data availability and/or larger virtual disk devices. RAID is implemented as a set of multiple storage devices (disks, tapes, and solid-state disks), called an array, and a specialized array controller, that manages the distribution of data across the array.

A RAID array, whether it contains two, five, or seven physical drives, can be configured to look like one or more large virtual disk drives. Use a RAID array virtual drive just as you would a physical drive. You can partition it if you want, and you do not need to make any application changes to realize the benefits of RAID. A RAID array provides higher levels of data availability and performance than a single physical disk drive of similar capacity.

Data for a given file is divided into chunks that is then be written across multiple drives. A *chunk* is a group of contiguous data blocks that are stored on a single physical disk drive. By using more than one physical drive, the data is transferred in chunks to multiple physical devices simultaneously, achieving transfer rates greater than each physical disk. Depending on the RAID level used, arrays also provide redundancy to protect the data availability. Arrays provide redundancy in two main ways: by mirroring and by generating parity.

The storage configuration options available depend upon your storage needs and the number of disks that you purchased for your RAID array. Table A–1 describes the storage options available and the minimum number of physical disks required to implement each.

You can use a variety of storageset type containers within a single subsystem, providing you have the disk device resources to support them.

**Table A–1 Storage Configuration Options**

| Storage Method | Storageset Type | Number of Devices | Offers |
|---|---|---|---|
| RAID 3/5<br>A redundant-stripeset combining the optimized data transfers of RAID 3 with the striping of parity of RAID 5. | RAIDset | 3 - 14 | Good throughput and read bandwidth for a high request rate of small to medium transfers.<br>High Data Availability. |
| RAID 0 | Stripeset | 2 - 14 | Good performance for both read and write requests. Provides load balancing with each request requiring a single data operation.<br>Data availability equivalent to that of an individual disk device. |
| RAID 1 | Mirrorset | 2 - 6 devices per mirrorset, up to 20 mirrorsets per RAID array | Good performance for read requests.<br>High Data Availability |
| RAID 0 + 1 | Striped mirrorsets | 2 - 14 mirrorsets | Performance for read requests surpassing that of an unstriped mirrorset since it can achieve load balancing.<br>High Data Availability. |
| Individual Devices (JBOD) | Disk Drive | 1 | Provides the storage capacity and access speed of the disk used.<br>If device fails, data is lost. |

Once you select the type of storagesets that you want to use in your subsystem, you must create them using an appropriate configuration manager.

# B

# *Command Console LUN*

*This appendix discusses the Command Console LUN.*

## B.1 Command Console LUN

The StorageWorks RAID Array is configured by creating various types of storagesets and associating them with specific IDs called Logical Unit Numbers (LUNs). The Host uses these LUNs to access the underlying storageset. The RAID Array is preconfigured with a virtual LUN, located on Controller A, LUN 0. This device, called the Command Console LUN (CCL), allows the RAID Array to be recognized by the Host System as soon as it is attached to the SCSI bus and configured into the operating system. The CCL also serves as a communications device for the StorageWorks Command Console (SWCC) Agent. The CCL identifies itself to the Host via a unique identification string. This string, **HSG80CCL**, is returned in response to the Inquiry command.

When the CCL is enabled, as it is upon delivery, its address can be determined using the CLI command:

```
HSG80 > SHOW THIS_CONTROLLER
```

The CCL can be disabled using the CLI command:

```
HSG80 > SET THIS_CONTROLLER NOCOMMAND_CONSOLE_LUN
```

The CCL is enabled using the CLI command:

```
HSG80 > SET THIS_CONTROLLER COMMAND_CONSOLE_LUN
```

In Dual-Redundant Controller configurations, these commands alter the setting of the CCL on both Controllers. The CCL is enabled only on Host Port 1. At least one storage device must be configured on Host Port 2 before installing the Agent on a Host System connected to Host Port 2. This can be any type of storage device.

Select a storageset that you plan to configure and that is not likely to change. This storageset can be used by the agent to communicate with the RAID Array. Deleting this storageset (LUN) later will break the connection between the agent and the RAID Array.

# C

# *Understanding RAID Manager's Mail Messages*

*This appendix provides information to help you understand the contents of the automatic mail message(s) RaidManager sends upon the occurrence of an event, if so enabled.*

## C.1 Mail Message Information

If you have enabled *email notification* via the Agent installation program, (Chapter 2), the Agent sends email notification to the specified network email addresses when a storage subsystem event occurs. The storage subsystem event can be a state change of a physical storage device, logical storage unit or a component of the physical enclosure.

You will always receive this message as being from the RaidManager. The message will specify the name of the Host to which the Raid Array (reporting the event) is connected. The message also tells the severity of the problem, which will be one of three levels:

- Critical
- Warning
- Informational

The mail message appears in the following form:

```
-----------------
From RaidManager Tue Oct  6 15:59:59 1998
Date: Tue, 6 Oct 1998 15:59:58 -0500 (EST)
From: RaidManager

This is an automatic message from your StorageWorks
RAID Agent, steamd. The following message was just received.
Please check your syslog files and RAID box!

Hostname: Suncity

CRITICAL: Validation failed - Unauthorized client (shr-dhcp-24-
188.shr.dec.com, access level: -1); connection refused  (SP_TCP:
ClientConnect)


End Of Automatic Message.
--------------------
```

## C.2 Event Information Fields

Often the message is self-explanatory, as shown in the previous example. Other messages reflect the state of the RAID subsystem by displaying event information fields in a line following the problem severity level.

*Example:*

```
WARNING:-
Suncity Hsg80 12000000000 HSG80 disks(disk21100:2) (SP_MONITOR: MonitorSubsys)
      \       \            \
       \       \            \
       (1)     (2)          (3)
```

The event information fields can be deciphered as follows:

(1) **Host Name -** to which the RAID experiencing the event is connected.

(2) **Storage Subsystem -** where the problem occurred.

(3) **State Change -** of a particular component of the storage subsystem. This is an eleven digit field that provides the crucial information about the status of your RAID Array.

### C.2.1 Mapping State Change Digits to RAID Subsystem Components

Each of the eleven digits in the state change field can be mapped to a RAID subsystem component for deciphering event reports. Table C–1 lists the relationship.

**Table C–1 State Change Digit Position and Corresponding Subsystem Component**

| "State Change" Digit Position | Corresponding RAID Subsystem Component |
|---|---|
| 1 | Overall Subsystem |
| 2 | Disks |
| 3 | Power Supply |
| 4 | Fans |
| 5 | Battery |
| 6 | Temperature |
| 7 | This Controller |
| 8 | Communications LUN |
| 9 | Other Controller |
| 10 | External Factors |
| 11 | Logical Units |

### C.2.1.1 The First Digit of the State Change Field - (Overall Subsystem)

The **first digit** of the state change field tells about the state change of the **overall subsystem**.

*Example:*

```
WARNING: -
Suncity Hsg80 12000000000 HSG80 disks(disk21100:2) (SP_MONITOR: MonitorSubsys)
```

This digit can take 2 values:

0 - Everything is fine

1 - Something has changed state

### C.2.1.2 The Second Digit of the State Change Field - (Disks)

The **second digit** of the state change field tells about the state change of **disks**.

*Example:*

```
WARNING: -
Suncity Hsg80 12000000000 HSG80 disks(disk21100:2) (SP_MONITOR: MonitorSubsys)
```

This digit can take three values.

0 - Everything is fine

1 - Drive went from bad to good

2 - Drive went from good to bad

In this example, a value of 2 is displayed in the disk digit position, which indicates that a drive went from good to bad.

The disknames follow in the message line, (in parenthesis), in the format:

*disk_name:state*

where *disk_name* is the name of the disk and *state* can be either of the following:

1 - disk went from bad to good

2 - disk went from good to bad

In this example, the failed drive is disk21100.

### C.2.1.3 The Third Digit of the State Change Field - (Power Supply)

The **third digit** of the state change field tells about state change of **power supply**.

*Example:*

```
WARNING: -
Suncity Hsg80 1022000000000 HSG80 pwr(0:1:2)  (SP_MONITOR: MonitorSubsys)
```

The third digit can take three values:

0 - Everything is fine

1 - Power Supply went from bad to good

2 - Power Supply went from good to bad

The position of the failed power supply usually follows in the message line, in parenthesis, as three numbers that indicate the position in the format:

*cabinet_number:power_position:state*

where *cabinet_number* is the cabinet ID from 0 to 3, the *power-position* is the power supply location from 1 to 8 and *state* can be any of the following:

1 - Power Supply went from bad to good

2 - Power Supply went from good to bad

3 - Power Supply is not present.

In this example, the Power Supply in position 1of Cabinet 0 (main cabinet) went from good to bad.

### C.2.1.4 The Fourth Digit of the State Change Field - (Fans)

The **fourth digit** of the state change field tells about the state change of **fans**.

*Example:*

```
WARNING: -
Suncity Hsg80 1002000000000 HSG80 fans(0:A:2)  (SP_MONITOR: MonitorSubsys)
```

The fourth digit can take three values:

0 - Everything is fine

1 - Fan state went from bad to good

2 - Fan state went from good to bad

The position of the fan follows in the message line, in parenthesis, in the format:

*cabinet_number:fan_position:state*

where *cabinet_number* is the cabinet ID from 0 to 3, *fan_position* is the position of the fan and *state* is either of the following:

1 - Fan went from bad to good

2 - Fan went from good to bad

In this example, a fan failure occurred in cabinet 0, the main cabinet.

### C.2.1.5 The Fifth Digit of the State Change Field - (Battery)

The **fifth digit** tells about the **battery** state change.

*Example:*

```
WARNING: -
Suncity Hsg80 10002000000 HSG80 batt(6:fail)  (SP_MONITOR: MonitorSubsys)
```

This digit can take three values:

0 - Everything is fine

1 - Battery state went from bad to good

2 - Battery state went from good to bad

The details of the battery failure follow in the message line, in parenthesis, in the format:

*controller_id:state*

where *controller_id* is the SCSI ID of the **reporting** controller (in other words, the cache battery failed for the other controller), and *state* is any of the following:

good - Battery is good

low - Battery voltage is low

fail - Battery has failed

In this example, the cache battery failed for the controller with SCSI ID 7.

### C.2.1.6 The Sixth Digit of the State Change Field - (Temperature)

The **sixth digit** of the state change field is about **temperature** state changes.

*Example:*

WARNING: -
Suncity Hsg80 10000200000 HSG80 temp(0:2:2) (SP_MONITOR: Monitor_Subsys)

The sixth digit can take three values:

0 - Everything is fine

1 - Temperature state went from bad to good

2 - Temperature state went from good to bad

The details regarding the temperature change follows in the message, in parenthesis, in the format:

*cabinet_number:sensor_type:state*

where *cabinet_number* is the cabinet ID from 0 to 3, *sensor_type* is the EMU sensor, (sensor 1 or sensor 2), and *state* is either of the following:

1 - Temperature went from bad to good

2 - Temperature went from good to bad

In this example, an adverse temperature change is detected by sensor 2 of the main cabinet (0).

### C.2.1.7 The Seventh Digit of the State Change Field - (This_Controller)

The **seventh digit** tells about the changes in **This_Controller.**

**NOTE**

This value is always zero for a dual redundant configuration.

The seventh digit will change to 1 in case of the failure of the controller in a single controller configuration.

### C.2.1.8 The Eighth Digit of the State Change Field - (Communications LUN))

The **eighth digit** tells about the state changes of the **Communications LUN**.

*Example:*

CRITICAL: - Suncity Hsg80 1000000**2**000 HSG80 (SP_MONITOR: MonitorSubsys)

CRITICAL: Unable to open device - hdisk1 (SP_MONITOR: MonitorSubsys)

This digit can take three values:

0 - Everything is fine

1 - Communication LUN is available to the host machine

2 - Communication LUN is not available to the host machine

In this example, two mail messages appear. The first message indicates a Communications LUN change of state from good to bad. The second message indicates that the Agent is not able to open the Communications LUN for monitoring the subsystem.

### C.2.1.9 The Ninth Digit of the State Change Field - (Other_Controller)

The **ninth digit** indicates the state change of the **Other_Controller**.

*Example:*

WARNING: -
Suncity Hsg80 10000000**2**00 HSG80 (SP_MONITOR: MonitorSubsys)

This value will never change for a single controller configuration. It will always change when either of the controllers fail in a dual redundant configuration.

This digit can take three values:

0 - Everything is fine

1 - Both controllers are functioning

2 - One of the two controllers failed

Physically inspect the controllers to verify which one has failed, as indicated by a solid green indicator light (not blinking).

### C.2.1.10 The Tenth Digit of the State Change Field - (External Factors)

The **tenth digit** indicates the a state change detected as caused by **external factors**.

*Example:*

```
WARNING: -
Suncity Hsg80 1000000002 0 HSG80  (SP_MONITOR: MonitorSubsys)
```

This digit can take three values:

0 - Everything is fine

1 - State changed from bad to good

2 - State changed from good to bad

### C.2.1.11 The Eleventh Digit of the State Change Field (Logical Units)

The **eleventh digit** indicates a state change of Logical Units.

*Example:*

```
WARNING: -
Suncity Hsg80 1000000000 2 HSG80  lun(d100:4) (SP_MONITOR: MonitorSubsys)
```

This digit can take 3 values:

0 - Everything is fine

1 - A LUN state changed from bad to good

2 - A LUN state changed from good to bad

The details about the failure of the LUN follow in the message line, in parenthesis, in the format:

*virtual_disk:state*

where *virtual_disk* is the unit number of the virtual disk and *state* is any of the following:

0 - good

1 - reduced

2 - reconstructing

4 - failed

In this example, Logical Unit D100 failed.

In most cases, the RaidManager will send at least two consecutive mail messages. The first one is always of the form:

```
---------------
From RaidManager Tue Oct  6 16:09:37 1998
Date: Tue, 6 Oct 1998 16:09:37 -0500 (EST)
From: RaidManager

This is an automatic message from your StorageWorks
RAID Agent, steamd. The following message was just received.
Please check your syslog files and RAID box!

Hostname: Suncity

WARNING:  - A subsystem change has been detected: Suncity HSG80 OVRL=1
(SP_MONITOR: MonitorSubsys)

End Of Automatic Message.
-------------
```

It is the second mail message that deciphering is required by applying the data provided in this appendix. The second message will be similar to the following (depending on the error that is occurring):

```
---------
From RaidManager Tue Oct  6 16:09:37 1998
Date: Tue, 6 Oct 1998 16:09:37 -0500 (EST)
From: RaidManager

This is an automatic message from your StorageWorks
RAID Agent, steamd. The following message was just received.
Please check your syslog files and RAID box!

Hostname: Suncity

WARNING:  - Suncity Hsg80 12000000000 HSG80 disks(disk10200:1 disk60300:2)
(SP_MONITOR: MonitorSubsys)

End Of Automatic Message.
-------------
```

# D

# *SNMP MIB Information*

*This appendix provides information about the SNMP traps that may be generated by the Agent.*

## D.1 SNMP Trap Information

When a physical storage device, a logical storage unit, or a physical enclosure component changes state, the Agent sends out an SNMP trap (that is, an asynchronous event notification) to any hosts that are listed in the CLIENT.INI file with this option enabled. (See Appendix E, *For Advanced Users*, for more information on client.ini settings.)

This section describes the format of the SNMP traps that may be generated. There are 21 trap types, one for each FRU type and state.

### D.1.1 Disk Device Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 1
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.28
Value 3 = <disk device name>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 4 = <cabinet>
Object 5 = 1.3.6.1.4.1.36.2.15.21.3.2.1.35
Value 5 = <channel>
Object 6 = 1.3.6.1.4.1.36.2.15.21.3.2.1.36
Value 6 = <target ID>
```

### D.1.2 Disk Device Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 2
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.28
Value 3 = <disk device name>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 4 = <cabinet>
Object 5 = 1.3.6.1.4.1.36.2.15.21.3.2.1.35
Value 5 = <channel>
Object 6 = 1.3.6.1.4.1.36.2.15.21.3.2.1.36
Value 6 = <target ID>
```

### D.1.3 Power Supply Device Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 3
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.29
Value 4 = <power supply location>
```

### E.1.4 Power Supply Device Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 4
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.29
Value 4 = <power supply location>
```

### D.1.5 Fan Device Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 5
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.30
Value 4 = <fan location>
```

### D.1.6 Fan Device Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 6
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.30
Value 4 = <fan location>
```

### D.1.7 Cache Battery Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 7
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.31
Value 3 = <cache battery location>
```

### D.1.8 Cache Battery Low Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 8
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.31
Value 3 = <cache battery location>
```

### D.1.9 Cache Battery Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 9
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.31
Value 3 = <cache battery location>
```

### D.1.10 Temperature Sensor Over Threshold Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 10
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.32
Value 4 = <temperature sensor location>
```

### D.1.11 Temperature Sensor Below Threshold Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 11
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
Object 4 = 1.3.6.1.4.1.36.2.15.21.3.2.1.32
Value 4 = <temperature sensor location>
```

### D.1.12 Communication Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 12
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
```

### D.1.13 Communication Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 13
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
```

### D.1.14 Controller Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 14
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
```

### D.1.15 Controller Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 15
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
```

### D.1.16 LUN Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 16
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.34
Value 3 = <lun name>
```

### D.1.17 LUN Reconstructing Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 17
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.34
Value 3 = <lun name>
```

### D.1.18 LUN Reduced Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 18
```

```
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.34
Value 3 = <lun name>
```

### D.1.19 LUN Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 19
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.34
Value 3 = <lun name>
```

### D.1.20 EMU External Input Failure Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 20
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
```

### D.1.21 EMU External Input Recovery Trap

```
Enterprise = 1.3.6.1.4.1.36.2.15.21
Source address = <ip address>
Generic = 6
Specific = 21
Variable Binding List :
Object 1 = 1.3.6.1.4.1.36.2.15.21.1.4
Value 1 = <host name>
Object 2 = 1.3.6.1.4.1.36.2.15.21.3.2.1.2
Value 2 = <subsystem name>
Object 3 = 1.3.6.1.4.1.36.2.15.21.3.2.1.33
Value 3 = <cabinet>
```

# E

# *Fibre Channel Switch Installation and Basic Set-up*

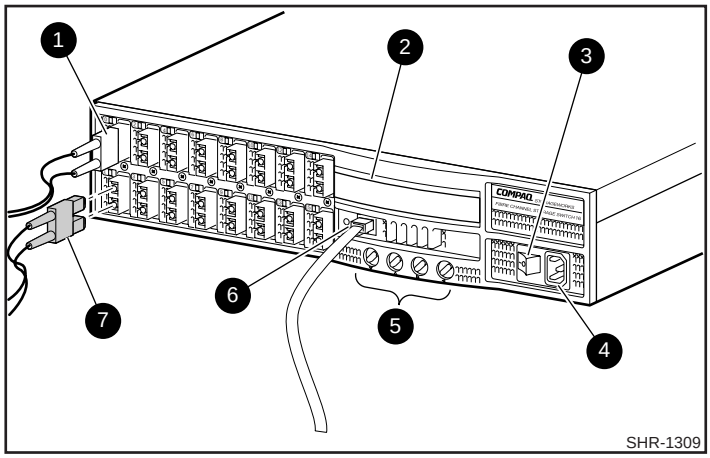
*This Appendix is designed to give you basic instructions for installing the Fibre Channel switch and for making the necessary connections. For complete details of this procedure refer to Compaq StorageWorks Quick Setup Guide StorageWorks Fibre Channel Switch, AA-RHCOA-TE and the Quick Setup Guide RAID Array 8000/ESA12000 Fibre Channel Storage Subsystem for Compaq Tru64 UNIX (ACS V8.4 for Fibre Channel Switch), AA-RFASA-TE/387388-001.*

## E.1 Procedure

The following steps are to be used in conjunction with Figure E–1.

1. Select a location and mounting method (refer to the Installation Card or User's Guide for details).
2. Connect the StorageWorks Subsystem:
   - Install a GBIC module in switch port 0.
   - Connect the fibre-optic cable from the StorageWorks subsystem to the GBIC (1).
3. Connect the Host:
   - Install a GBIC module in switch port 1.
   - Connect the fibre-optic cable from the host to the GBIC (7).
4. Connect the Ethernet Network:
   - Connect the Ethernet cable to the front panel RJ-45 connector (6).
5. Connect the AC Power:
   - Connect the power cord to the AC connector (4).
   - Turn on the powers switch (3).

**Figure E–1 Connecting to the Fibre Channel Switch**




6. Verify successful POST:
   - Ensure power LED shows steady green.
   - Ensure Port 0 and 1 LEDs show steady green (providing the ports are connected).
7. Assign an IP Address to the Switch:
   - Press the front panel *Down* button (5).
   - When *Select Menu: Configuration Menu* appears on the display (2), use the down button to select *Ethernet IP Address*.
   - Press the *Enter* button to display the switch Ethernet IP Address.
   - Use the *Up* and *Down* buttons to increment or decrement the displayed value; use the *Tab/Esc* button to move the cursor between fields.
   - When you have set the address, press the *Enter* button to store the address.

# F

# *Connections, Topologies, and other Fibre Related Settings for Fibre Channel on Compaq Tru64® UNIX*

*This appendix is a discussion with intent to get you started with some of the new Fibre Channel (FC) terms as they apply to the HSG80 connected to an Alpha server running the Compaq Tru64 UNIX. This material is not intended to be a course in FC technology but will, out of necessity, cover (at least in part) various aspects of it.*

## F.1 Introduction

Your new HSG80 controller is the first in a series that uses the Fibre Channel technology. Using FC as the connection protocol will allow you greater configuration flexibility both now and in the future.

Getting the most out of your HSG80 will require that you become familiar with some new Fibre Channel terms. You may have already discovered some of the new FC settings if you have connected a terminal to the HSG80's CLI port.

The areas of focus within this appendix deal with the initial setup of fairly simple configurations. If you are familiar with StorageWorks products in general (previous HSZn products), but are new to FC, then this material is meant for you.

## F.2 Background

### F.2.1 Connection Basics

The HSZn series RAID systems were relatively easy to connect to your Alpha server. In most cases, you would install the adapter, plug in the SCSI cable, and that was it for physical connections.

Things are a bit more complicated with Fibre Channel (FC). The HSG80 is not directly connected to the FC adapter in the server. Instead, both the FC adapter and the HSG80 RAID system are connected to a FC switch. The advantage of this is in the flexibility and accessibility that the switch affords you. With this setup, you can connect multiple servers and/or RAID arrays to have access to each other in a variety of ways.

Another difference is the cabling. Gone are the thick and sometimes inflexible SCSI cables. A thin and very flexible fiber optic cable has replaced them, and problems that use to exist with SCSI cables (skew, reflections, and signal strength) have been virtually eliminated. You may recall that the length limit on "Fast Wide Differential" SCSI cables was 25 meters. Compaq's fiber optic implementation of FC will give you the ability of cable lengths up to 10 km; that's about six miles! It is also worth noting that the fiber optic implementation can yield higher data bandwidths when compared to a copper cable implementation.

### F.2.2 Fibre Channel Basics

Fibre Channel is a transport mechanism that is similar to IP networking in many respects. It is important to note that FC is not a replacement for SCSI, but rather a method for transporting SCSI protocol, as well as other protocols. In the past, SCSI was implemented over a physical bus that was directly attached to the host. Fibre Channel transports SCSI protocol over a FC network to a remote SCSI bus or device.

Are you beginning to see the advantages of FC? Many servers and devices, like the HSG80 can be connected over great distances to form highly accessible networks of devices.

**Terms:**

The following are some FC terms that you will need to understand for the rest of the discussion:

- Node - The source or destination of information.
- Node Port (N_Port) - Device or mechanism needed to transport information between nodes.
- Domain - Two or more nodes connected together by means of an interconnect system.
- Link - A physical connection.
- Topology - A connection scheme or the way N_Ports are connected together.

**Topologies:**

As in IP networking, your FC network will have a topology. The topologies available to you on the HSG80 are:

- Direct Attach
- Arbitrated Loop
- Fabric

*Direct Attach* (also called *Point to Point*) means simply that, it is the simplest of the topologies but also the most limiting.

In an *Arbitrated Loop* topology, NL_Ports are connected to form a closed loop. This is more flexible than Direct Attach.

*Fabric* topology, also referred to as *Switched Fabric*, offers the greatest amount of flexibility. Where Arbitrated Loop has a maximum of up to 126 ports, Fabric has a maximum of 16 million ports.

Finally, on FC basics using the terms described above, the following general statements should be well noted. The HSG80, Alpha server (running Tru64 UNIX), and switch are setup with fiber optic links in a "switch fabric" topology. The multi-level FC protocol is used via the N_Ports to transfer SCSI protocol using the switch as a router.

## F.3 Connections

### F.3.1 Implementation

#### F.3.1.1 Current Implementation of FC on Compaq Tru64 UNIX

The following are important things to know about the current implementation of FC on Tru64 UNIX (V4.0F):

- Fabric Topology Supported
- 2 Targets, 16 LUNs maximum per HSG80
- HSG80 Units do not necessarily directly correlate with "rz_n" units.

As described in Section F.2.2, there are several topologies. Tru64 UNIX supports the most versatile one, "switched fabric." Not supported at this time are "direct attach" and "arbitrated loop;" however, they may be in the future.

HSG80 units do not necessarily have a one to one correlation with "rz" numbers. For example, let's say you have defined unit D0 and D100 in your HSG80. Based on past experience, if your adapter was configured as the second SCSI bus, you would expect to see rz8 and rz9 respectively. Don't be surprised if you see rz8 and rz10, or some other combination. The effect is caused because the host adapter assigns target IDs on a first seen, first assigned basis. Thus, changing of hardware or probing with different topology will result in different target ID assignments.

#### F.3.1.1 Current Implementation of FC on the HSG80

The following are important things to know about the current implementation of FC on the HSG80:

- Topology must be set.
- Connection operating_system must be set after initial connection.
- Care in defining units.

In order for your HSG80 to function properly, you need to modify the port topology. The port topology is defaulted at the time of manufacturing to "LOOP_HARD" since that was the popular mode when shipping first began. As mentioned, for Tru64 UNIX you will need the "fabric" topology setting. To set topology on both ports type the following from the CLI port:

```
SET THIS PORT_1_TOPOLOGY = FABRIC
SET THIS PORT_2_TOPOLOGY = FABRIC
SET OTHER PORT_1_TOPOLOGY = FABRIC
SET OTHER PORT_2_TOPOLOGY = FABRIC
RESTART OTHER
RESTART THIS
```

You can check to see if the port topology is set correctly by issuing a "SHOW" "THIS_CONTROLLER" or "OTHER_CONTROLLER" command. Both controllers should have one port up on fabric and the other port in standby mode. Here is an example of a "show this_controller" command showing just the port section.

```
Host PORT_1:
      Reported PORT_ID = 5000-1FE1-0000-32A1
      PORT_1_PROFILE   = PLDA
      PORT_1_TOPOLOGY  = FABRIC (standby)
Host PORT_2:
      Reported PORT_ID = 5000-1FE1-0000-32A2
      PORT_2_PROFILE   = PLDA
      PORT_2_TOPOLOGY  = FABRIC (fabric up)
      Address          = 210213
     NOREMOTE_COPY
```

The connection operating system must be set after the initial connection to the server is made. The default value set at the factory is "WINNT." This setting replaces the "Host Functionality Mode" that was present on the HSZ series of controllers. To set the operating system do the following.

```
HSG80> SHOW CONNECTIONS
Connection                                                          Unit
   Name      Operating system    Controller  Port    Address    Status Offset

!NEWCON01     WINNT                  THIS       1      210013   OL this  0
          HOST_ID=1000-0000-C920-A6C5          ADAPTER_ID=1000-0000-C920-A6C5

!NEWCON02     WINNT                  OTHER      2      210013   OL other 100
          HOST_ID=1000-0000-C920-A6C5          ADAPTER_ID=1000-0000-C920-A6C5

HSG80> SET !NEWCON07 OPERATING_SYSTEM = DIGITAL_UNIX
HSG80> SET !NEWCON08 OPERATING_SYSTEM = DIGITAL_UNIX
HSG80> RENAME !NEWCON01 BITHEAD_0
HSG80> RENAME !NEWCON02 BITHEAD_1
```

You could also use the “ADD CONNECTION” command but it requires that you input more parameters. The RENAME command was used to change the name of the connections to something more indicative of the connected system.

Typing in the “SHOW CONNECTIONS” command will show the change and it will look like this:

```
HSG80> SHOW CONNECTIONS
Connection                                                          Unit
   Name      Operating system    Controller  Port    Address    Status Offset

BITHEAD_0     DIGITAL_UNIX           THIS       1      210013   OL this  0
          HOST_ID=1000-0000-C920-A6C5          ADAPTER_ID=1000-0000-C920-A6C5

BITHEAD_1     DIGITAL_UNIX           OTHER      2      210013   OL other 100
          HOST_ID=1000-0000-C920-A6C5          ADAPTER_ID=1000-0000-C920-A6C5
```

If you are using the CLI to define units on the HSG80, care should be taken to ensure the units are visible to Tru64 UNIX. Remember, only 2 targets with 8 LUNs each are possible for Tru64 UNIX Version 4.0F. Therefore, unit numbers like D0, D1, D2... and D100, D101, D102.. are visible. However, unit numbers like D110, and D120 represent the $10^{th}$ and $20^{th}$ LUN of target 1 respectively and will not be seen.

### F.3.1.3 Hiding LUNs from Certain Connections using "Unit Offset"

Let's say you connect one server and one RA8000 to your switch. If you go into the CLI port and type in the "SHOW CONNECTIONS" command, you would see something like this.

**NOTE**

The examples in this section do not apply to clusters.

```
HSG80> SHOW CONNECTIONS
Connection                                                          Unit
   Name      Operating system    Controller  Port    Address    Status Offset

BITHEAD_0     DIGITAL_UNIX          THIS       1      210013   OL this  0
          HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000-C920-A6C5

BITHEAD_1     DIGITAL_UNIX          OTHER      2      210013   OL other 100
          HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000-C920-A6C5
```

Now let us say you add a second server to the switch. When you type in the "show connections" command now, you will see that the second server also has access to the same LUNs your first server has.

```
HSG80> SHOW CONNECTIONS
Connection                                                          Unit
   Name      Operating system    Controller  Port    Address    Status Offset

BITHEAD_0     DIGITAL_UNIX          THIS       1      210013   OL this  0
          HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000-C920-A6C5

BITHEAD_1     DIGITAL_UNIX          OTHER      2      210013   OL other 100
          HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000-C920-A6C5

EXTRA_0     DIGITAL_UNIX          THIS       1      210013   OL this  0
          HOST_ID=1000-0000-C920-A5A1         ADAPTER_ID=1000-0000-C920-A5A1

EXTRA_1     DIGITAL_UNIX          OTHER      2      210013   OL other 100
          HOST_ID=1000-0000-C920-A5A1         ADAPTER_ID=1000-0000-C920-A5A1
```

Looking at the "show connections" output you could see that two different hosts have access to the same LUNs. For instance, the first and third connections are two different hosts (Host ID's are different) with access to the first 8 LUNs. Note that the "Unit Offset" for both of these connections is equal to 0.

You may want multiple servers accessing the same LUNs but then again, you may not. Here is one method for making LUNs available to selected servers.

Unit Offset can be adjusted on your HSG80 along with the unit definitions to obtain a range of units visible to a specific server. Here is a simple example that you should be able to expand upon to fit your particular situation.

BITHEAD_0/1 are connected to host “bithead.shr.dec.com” and EXTRA_0/1 are connected to host “extra.shr.dec.com.” We are going to make Units D100 – D105 invisible to “bithead” and Units D0 – D5 invisible to “extra.”
The current Unit status from the HSG80 is:

```
SHOW UNITS
    LUN                                             Uses             Used by
------------------------------------------------------------------------------

  D0                                                DISK10000
  D1                                                DISK10100
  D2                                                DISK20000
  D3                                                DISK20100
  D4                                                DISK30000
  D5                                                DISK30100
  D100                                              DISK40000
  D101                                              DISK40100
  D102                                              DISK50000
  D103                                              DISK50100
  D104                                              DISK60000
  D105                                              DISK60100
HSG>
```

To make Units D100–D105 invisible to “bithead” we need to map them through “Unit Offset” so that they are out of the range of LUNs available to “bithead”. Since “bithead” is running Tru64 UNIX V4.0F and only 8 LUNs are available per target, we could simply execute the following to do the job:

```
SET BITHEAD_1 UNIT_OFFSET = 110
```

Now “bithead” will continue to see D0–D5 but D100–D105 are offset out of the range of LUNs the OS supports. Since nothing was done to the offset for “extra”, it will continue to see all the units. By now you can see that if you do a similar offset on the low order units for the “extra” connection then the example is complete. “extra” will see only D100–D105 and “bithead” will see only D0–D5.

The above example is only useful for very simple configurations. If you have more then two servers, getting the same results requires more planning and work. The next example gives you an idea of what to do if four servers are connected to one RA8000.

```
SHOW UNITS
    LUN                                             Uses             Used by
------------------------------------------------------------------------

  D10                                               DISK10000
  D11                                               DISK10100
  D12                                               DISK20000
  D20                                               DISK20100
  D21                                               DISK30000
  D22                                               DISK30100
  D150                                              DISK40000
  D151                                              DISK40100
  D152                                              DISK50000
  D160                                              DISK50100
  D161                                              DISK60100
  D162                                              DISK60100
HSG>
```

Notice how the units are defined in this sample output of the "SHOW UNITS" command. There are four distinct ranges of three units each. Normally, under Tru64 UNIX V4.0F you would not be able to see any of these units because they are all out of the 8 LUNs per target range.

To prove the point about the LUNs not being accessible, look at the results of the file command done on "bithead" and "extra."

```
extra.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrzg33c:   character special (8/66946) SCSI #4 HSG80CCL disk #270 (SCSI ID #1) (SCSI LUN #6)

bithead.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrzg16c:   character special (8/33154) SCSI #2 HSG80CCL disk #134 (SCSI ID #0) (SCSI LUN #6)
```

Only the CCL (the virtual communications LUN) is visible on each system. If you are wondering why it is showing up as the 6th LUN, the answer can be found in the previous section where I stated that the mapping is not necessarily one to one. In addition, previous LUN definitions bumped the floating CCL up to the sixth LUN

Now let's use unit offsets to map the first group of units D10–D12 to the server "bithead." Let's also map the last group of units D160–D163 to the server "extra." To do this, the "set unit offset" command needs to be executed in this way:

```
SET BITHEAD_0 UNIT_OFFSET = 10
SET EXTRA_1 UNIT_OFFSET = 160
```

Executing the file command again on “bithead” and “extra” we should see the respective LUN assignments.

```
Superbithead.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrz16c:    character special (8/32770) SCSI #2 HSG80 disk #128 (SCSI ID #0) (SCSI LUN #0)
/dev/rrzb16c:   character special (8/32834) SCSI #2 HSG80 disk #129 (SCSI ID #0) (SCSI LUN #1)
/dev/rrzc16c:   character special (8/32898) SCSI #2 HSG80 disk #130 (SCSI ID #0) (SCSI LUN #2)

Superextra.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrz32c:    character special (8/65538) SCSI #4 HSG80 disk #256 (SCSI ID #0) (SCSI LUN #0)
/dev/rrz33c:    character special (8/66562) SCSI #4 HSG80CCL disk #264 (SCSI ID #1) (SCSI LUN #0)
/dev/rrzb32c:   character special (8/65602) SCSI #4 HSG80 disk #257 (SCSI ID #0) (SCSI LUN #1)
/dev/rrzc32c:   character special (8/65666) SCSI #4 HSG80 disk #258 (SCSI ID #0) (SCSI LUN #2)
```

Here is the result of the “SHOW CONNECTIONS” command:

```
HSG> show connections
Connection                                                          Unit
  Name     Operating system    Controller  Port    Address    Status Offset

BITHEAD_0     DIGITAL_UNIX          THIS      1      210013   OL other   10
         HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000C920-A6C5

BITHEAD_1     DIGITAL_UNIX          THIS      2      210013   OL this   100
         HOST_ID=1000-0000-C920-A6C5         ADAPTER_ID=1000-0000-C920-A6C5

EXTRA_0       DIGITAL_UNIX          OTHER     1      210613   OL other    0
         HOST_ID=1000-0000-C920-A5A1         ADAPTER_ID=1000-0000-C920-A5A1

EXTRA_1       DIGITAL_UNIX          THIS      2      210613   OL this   160
         HOST_ID=1000-0000-C920-A5A1         ADAPTER_ID=1000-0000-C920-A5A1
```

Notice that the CCL only appears on “extra.” This is because the floating communications lun will not be visible on Tru64 UNIX if mapped to any unit offset above 0 (See 6.0 CCL). This is because the floating communications LUN will not map to any offset above 0. The only connection with a 0 offset is going to “extra” therefore “extra” is the only system that sees the CCL. The importance of this is discussed in a following section on SWCC. In addition, another two servers were not handy to complete the example—but hopefully you get the idea. You would just map them in the same fashion as we did with the first two connections.

As far as the two unused connections are concerned, you could delete them but they would just come back again after a reboot. The best thing to do is leave them there. Since there are no units defined in these two mappings, no units will be seen by the servers.

### F.3.1.4 Hiding LUNs from connections using“ENABLE_ACCESS_PATH.”

You may have noticed in the previous section that using unit offsets assigned “groups” of units to certain connections. The example was no accident in that respect. Using unit offsets will only allow you to get down to a resolution of 8 LUNs per connection and that may be all you need.

What if you wanted finer granularity? The ability to create a unit and then make it available to one and only one connection would be a nice feature. It turns out that this feature is available with the HSG80 and it is an option under the “SET UNIT” CLI command.
Let us go through a quick example on how to use this feature. Executing the “SHOW UNITS” command, as we did in the previous example, we get the following.

```
SHOW UNITS
    LUN                                             Uses             Used by
------------------------------------------------------------------------------

  D10                                               DISK10000
  D11                                               DISK10100
  D12                                               DISK20000
  D20                                               DISK20100
  D21                                               DISK30000
  D22                                               DISK30100
  D150                                              DISK40000
  D151                                              DISK40100
  D152                                              DISK50000
  D160                                              DISK50100
  D161                                              DISK60100
  D162                                              DISK60100
HSG>
```

Let’s change unit D161 to be accessible only to the connection BITHEAD_1. If you look back at the last example in the previous section, you will notice that currently D161 is part of a group of units that is only accessible through connection EXTRA_1. Now let’s make this one unit inaccessible to EXTRA_1 and accessible to BITHEAD_1 exclusively.

Typing the “SHOW D161” command, we get the following:

```
HSG> SHOW D161
    LUN                                     Uses             Used by
------------------------------------------------------------------------------
D161                                      DISK60000
        LUN ID:      6000-1FE1-0000-32A0-0009-8170-1056-001E
        NOIDENTIFIER
        Switches:
          RUN                    NOWRITE_PROTECT        READ_CACHE
          READAHEAD_CACHE        WRITEBACK_CACHE
          MAXIMUM_CACHED_TRANSFER_SIZE = 32
        Access:
              ALL
        State:
          ONLINE to this controller
          Not reserved
        Size: 17769177 blocks
        Geometry (C/H/S): ( 5258 / 20 / 169 )
HSG>
```

If you look at the “Access” field, you will notice that it is set to “ALL.” To achieve the desired results, we will use an option to the “SET” command. First, since “Access” is set to “ALL” we need to disable the access to all.

```
SET D161 DISABLE_ACCESS_PATH = ALL
```

Now with all access disabled, we can enable access just for the desired connection, which for this example is BITHEAD_1.

```
SET ENABLE_ACCESS_PATH = BITHEAD_1
```

Executing the “SHOW D161” command again displays the following:

```
HSG> SHOW D161
    LUN                                     Uses             Used by
------------------------------------------------------------------------------
D161                                      DISK60000
        LUN ID:      6000-1FE1-0000-32A0-0009-8170-1056-001E
        NOIDENTIFIER
        Switches:
          RUN                    NOWRITE_PROTECT        READ_CACHE
          READAHEAD_CACHE        WRITEBACK_CACHE
          MAXIMUM_CACHED_TRANSFER_SIZE = 32
        Access:
          BITHEAD_1
        State:
          ONLINE to this controller
          Not reserved
        Size: 17769177 blocks
        Geometry (C/H/S): ( 5258 / 20 / 169 )
HSG>
```

Finally, we should see this unit show up as a LUN on *bithead*. We should no longer see it on *extra*. The file command from extra shows that it definitely is no longer accessible from there. However, it does not appear to be showing up on bithead! What happened?

```
Superextra.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrz32c:    character special (8/65538) SCSI #4 HSG80 disk #256 (SCSI ID #0) (SCSI LUN #0)
/dev/rrz33c:    character special (8/66562) SCSI #4 HSG80CCL disk #264 (SCSI ID #1) (SCSI LUN #0)
/dev/rrzc32c:   character special (8/65666) SCSI #4 HSG80 disk #258 (SCSI ID #0) (SCSI LUN #2)

Superbithead.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrz16c:    character special (8/32770) SCSI #2 HSG80 disk #128 (SCSI ID #0) (SCSI LUN #0)
/dev/rrzb16c:   character special (8/32834) SCSI #2 HSG80 disk #129 (SCSI ID #0) (SCSI LUN #1)
/dev/rrzc16c:   character special (8/32898) SCSI #2 HSG80 disk #130 (SCSI ID #0) (SCSI LUN #2)
```

If you look back at the connection offset for BITHEAD_1 you will see that it is set to 100. The unit is D161 so the unit is out of the LUN range for that connection.

At this point, you can do one of two things. Which one you choose will depend on your particular environment.

1. You can delete unit D161 and recreate it as say D100. The above process will then work. This is the easiest way to get what you want but you have to be careful when you do this. **Remember**: You must first delete it and then recreate it. Also, if you elect to do this you should consider changing the unit offset since any new servers would create default connections with an offset of 0 and 100.
2. You can set the access of each LUN in the 160 range (3 in this case) so that each one explicitly lists what it can connect to. You will also need to change the offset on BITHEAD_1 to 160. This requires a bit more work but may be necessary if you don't want to rename the units for some reason.

Let's take the simple route to achieving the desired results. First, we will delete the unit D161 then recreate it as unit D100. Next, we will set the unit access for exclusive use by the BITHEAD_1 connection.

```
HSG> DELETE D161
HSG> ADD UNIT D100 DISK60000
HSG> SET D100 DISABLE_ACCESS_PATH = ALL
HSG> SET D100 ENABLE_ACCESS_PATH = BITHEAD_1

HSG> SHOW D100
    LUN                                        Uses             Used by
------------------------------------------------------------------------------
D100                                        DISK60000
        LUN ID:      6000-1FE1-0000-32A0-0009-8170-1056-001E
        NOIDENTIFIER
        Switches:
          RUN                     NOWRITE_PROTECT        READ_CACHE
          READAHEAD_CACHE         WRITEBACK_CACHE
          MAXIMUM_CACHED_TRANSFER_SIZE = 32
        Access:
          BITHEAD_1
        State:
          ONLINE to this controller
          Not reserved
        Size: 17769177 blocks
        Geometry (C/H/S): ( 5258 / 20 / 169 )

Superbithead.shr.dec.com> file /dev/rrz*c | grep HSG
/dev/rrz16c:    character special (8/32770) SCSI #2 HSG80 disk #128 (SCSI ID #0) (SCSI LUN #0)
/dev/rrz17c:    character special (8/32834) SCSI #2 HSG80 disk #129 (SCSI ID #0) (SCSI LUN #1)
/dev/rrz18c:    character special (8/32898) SCSI #2 HSG80 disk #130 (SCSI ID #0) (SCSI LUN #2)
/dev/rrz19c:    character special (8/35842) SCSI #2 HSG80 disk #152 (SCSI ID #3) (SCSI LUN #0)
```

The unit is now set up for exclusive use by connection BITHEAD_1 and using the file command on bithead you can see it as rz19.

**Important**: Notice that we did not initialize the disk before redefining the unit to D100. Never will you want to initialize during a unit rename operation like this.

With proper planning, the use of unit access and unit offsets, and just about any access scheme can be satisfied. Planning is the key to success and flexibility in your future configurations.

## F.4 SWCC Notes

There are a few important things to mention about running the SWCC agent on Tru64 UNIX with the HSG80. These points center on the fact that multiple systems can access the same LUNs.

When a server connects to the HSG80 via switched fabric a zero offset connection is established by default. If you have multiple servers connected to the same fabric, it is important that the SWCC agent only run on one server per RA8000 or multiple RA8000's.

Given the nature of switched fabric Fibre Channel, it is possible to have a large number of servers connected to one RA8000. It is also possible to have one server connected to many RA8000's. You must take the necessary steps to ensure that there are no conflicts when running the SWCC agent.

- If one server on your fabric has zero (0) offset connections to all of your RA8000's on that fabric then that server should be the one to run the SWCC agent.
- You must not have the SWCC agent running on multiple hosts servicing the same RA8000.
- If one server on your fabric does not have clear access to the CCL on all RA8000's (connection offset 0) it is acceptable to run the necessary amount of agents required to service all of the RA8000's. In this case, you must adhere to the above rule, which states that you cannot have any two agents servicing the same RA8000.
- One possible plan of attack that should help you remain problem free with SWCC agent operation, is as follows:

  – Plan to keep all of your LUNs out of the "0" offset range.

  – Install the SWCC agent on one system that is always on the network.

  – Make sure the "agent" system has an active connection to all RA8000's on the fabric.

## F.5 Miscellaneous Notes

- HSZTERM should never run concurrently with SWCC. Running SWCC and HSZTERM concurrently could result in lose of data, system instability or both.
- Currently, a maximum of 32 connections is possible. Due to persistence, connections that are offline are not deleted. If you do a lot of reconfiguring then over time, the connection table may fill. If it does you will get an error message telling you that you cannot create a new connection. If this happens, you should delete stale and/or unused connections from the table.
- When renaming a connection keep in mind that the system reserves the "!NEWCONxx" syntax for naming conventions. This means that your names must be different. Furthermore, once you change a connection name from "!NEWCONxx," you cannot rename it back to its previous "!NEWCONxx" name.
- REPLACEMENT PARTS (A general note of *caution*) - Be aware that your adapter (KGPSA) has a World Wide Name (WWN). If it's replaced with another one you may to check some of your FC connection settings. For instance, if you use the HSG80 units "ENABLE_ACCESS_PATH" setting, you will have to update it.

- The Connections "Operating System" field will always default to WINNT on new connections if the following conditions exist:
  - If the connection being made is the first one on the fabric.
  - All of the current connections are not set to the same operating system.
  - If there are other connections on the fabric and they are all set to "Digital Unix" then a new connection will set itself to "Digital Unix."

If not you may have to change the operating system using the SET command.

- By default the port topologies are set to LOOP_HARD. The topology must be set to FABRIC for Tru64 UNIX. You can accomplish this with the SET command. You must restart the controllers for this setting to take effect.
- World Wide Names (NODE_ID) stay with the cabinet and not the controller. If you need to replace or move controllers you should use the SET command to enter the WWN + checksum. The WWN and checksum is printed on top of the controller shelves.

## F.6. CCL Notes

The CCL (Command Console Lun) also called the virtual communications LUN is so important it requires a section to explain a few things. The following notes on the CCL refer to its behavior when the controller is set to SCSI-2 mode. SCSI-2 mode is the supported mode for Tru64 UNIX V4.0F.

Tru64 UNIX V4.0F has an 8 LUN per target limit. This is true of the CCL as well as real disk units. This operating system restriction limits the range that the floating CCL will be visible in.

With respect To Tru64 UNIX V4.0F, if you create enough units on the HSG80 to use up the available luns then the CCL will float out of range. In other word; the operating system will not see it. If the "Unit Offset" for "Port 1" is set to something other then "0" then the CCL will not be seen by the operating system. These restrictions will not exist in V5.0 of Tru64 UNIX.

Facts about the CCL (ignoring operating system limitations) for SCSI-2 mode only:

The CCL will float starting with unit 0 up to unit 99. If unit 99 is used then it will move to Port 2 and move from unit 100 through 199 as needed.

If a real unit is created where the CCL currently is located, the CCL will always move to a higher number. It does this dynamically and in an increasing direction only.

Upon a “RESTART” the CCL will be assigned the first available number starting from 0. If you create 8 units (0 – 7) the CCL will now be out of range for Tru64 UNIX V4.0F. If you now delete unit 0 (D0) and “restart” the controller, the CCL will be assigned to the first available unit number, in this case 0.