

Compaq SANworks

Command Scriptor User Guide

Part Number: AA-RN6EB-TE

Second Edition (November 2001)

Product Version: Version 1.0A

This publication is primarily for use by System Administrators who use SANworks Command Scriptor to configure and monitor StorageWorks subsystems with HSG60, HSG80, HSZ70, HSZ80, and HSJ80 controllers.

COMPAQ

© 2001 Compaq Information Technologies Group, L.P.

Compaq, the Compaq logo, SANworks, StorageWorks, Tru64, and OpenVMS are trademarks of Compaq Information Technologies Group, L.P. in the U.S. and/or other countries.

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

UNIX is a trademark of The Open Group.

All other product names mentioned herein may be trademarks of their respective companies.

Confidential computer software. Valid license from Compaq required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Compaq shall not be liable for technical or editorial errors or omissions contained herein. The information is provided "as is" without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.

Compaq service tool software, including associated documentation, is the property of and contains confidential technology of Compaq Computer Corporation or its affiliates. Service customer is hereby licensed to use the software only for activities directly relating to the delivery of, and only during the term of, the applicable services delivered by Compaq or its authorized service provider. Customer may not modify or reverse engineer, remove, or transfer the software or make the software or any resultant diagnosis or system management data available to other parties without Compaq's or its authorized service provider's consent. Upon termination of the services, customer will, at Compaq's or its service provider's option, destroy or return the software and associated documentation in its possession.

Printed in the U.S.A

Command Scriptor User Guide
Second Edition (November 2001)
Part Number: AA-RN6EB-TE

Contents

About this Guide

Text Conventions	vii
Symbols in Text	ix
Getting Help	ix
Compaq Technical Support	ix
Compaq Website	x
Compaq Authorized Reseller	x

1 Command Scripter Overview

Introduction to Command Scripter	1-1
Command Scripter Browser Interface	1-2
Command Scripter Command Line Interface	1-2
Controllers Supported	1-2
Operating Systems Supported	1-2
Command Scripter Configuration Requirements	1-3
Command Scripter Host Preparation	1-3
SWCC Host Preparation	1-4
HSG Element Manager Configuration	1-5
Command Scripter Commands	1-5
Command Scripter Commands - Definition, Examples, and Output	1-6
Agentver	1-7
Hostdata	1-7
Subsysdata	1-7
Status	1-7
Devlist	1-9
Lock	1-9
Unlock	1-10
CLI	1-10
Waitnormal <Storageset> <Poll> <Timeout>	1-11
Delay x	1-12

Quit	1-12
Print	1-12
Version.....	1-12
Max Wait x	1-12

2 Using Command Scripter Browser Interface

Introduction to the Browser Interface.....	2-1
Starting the Browser Interface	2-1
Accessing Browser Interface	2-1
Supported Browsers	2-2
Browser Interface Usage.....	2-2
Browser Interface Functionality.....	2-2
Select Agent Host.....	2-3
Select Subsystem	2-4
Run Commands and Scripts.....	2-5
Script Editor.....	2-7

3 Using Command Scripter Command Line Interface

Introduction to Command Line Interface	3-1
Terminology	3-3
General Guidelines for Command Scripter Usage	3-4
Command Scripter Connection Methods	3-4
Command Format.....	3-4
Using Command Scripter with the SWCC Agent or HSG Element Manager.....	3-5
Direct Access to Storage Subsystem Using Device Name	3-6
Direct Access to Storage Subsystem Using Controller Serial Number	3-7
Direct Access to Storage Subsystem Using a Worldwide Unique Identifier.....	3-7
Direct Access to Storage Subsystem Using Bus/Target/LUN on Tru64 UNIX ...	3-7
Direct Access to Storage Subsystem Using a SCS Node Name	3-8
Interactive Command Execution	3-8
Displaying a List of Available Controllers	3-9
Exit Status Indicators.....	3-9
Command Scripter Operating System Support.....	3-9
OpenVMS	3-10
Using Command Scripter Without an Agent.....	3-10
Using Command Scripter with an Agent.....	3-10
Windows NT and Windows 2000	3-11
Using Command Scripter Without an Agent.....	3-11

Using Command Scripter with an Agent	3-12
Compaq Tru64 UNIX	3-13
Using Command Scripter without an Agent	3-13
Using Command Scripter with an Agent	3-14
Sun Solaris	3-15
Using Command Scripter Without an Agent	3-15
Using Command Scripter with an Agent	3-16
IBM AIX	3-16
Using Command Scripter Without an Agent	3-16
Using Command Scripter with an Agent	3-17
HP-UX	3-18
Using Command Scripter Without an Agent	3-18
Using Command Scripter with an Agent	3-18

Index

Figures

2-1	Title Bar	2-3
2-2	Select Agent Host Page	2-4
2-3	Select Subsystem Page	2-5
2-4	Run Commands and Scripts Page	2-7
2-5	Script Editor Page	2-8
3-1	Using Command Scripter Without an Agent	3-2
3-2	Using Command Scripter with an Agent	3-3

Tables

1	Text Conventions	vii
1-1	Command Scripter Host Preparation	1-3
1-2	SWCC Host Server Preparation	1-4
1-3	HSG Element Manager Configuration	1-5
1-4	Command Scripter Commands	1-5

About this Guide

The following topics are covered in this section:

- Text Conventions
- Symbols in Text
- Getting Help
- Compaq Technical Support
- Compaq Website
- Compaq Authorized Reseller

Text Conventions

This document uses the conventions in Table 1 to distinguish elements of text.

Table 1: Text Conventions

Element	Convention	Examples
<ul style="list-style-type: none">• Named Keys• Key Sequences	Bold	Home, Print Screen, Num Lock, Esc, PgUp A plus sign (+) between two keys means that you should press them simultaneously: Ctrl+A, Ctrl+Home, Alt+Ctrl+Del
<ul style="list-style-type: none">• Menu Items• Directory Names• Button Names• Dialog Box Names	Initial Caps (for UNIX, AIX, HP-UX and Solaris directory names, the exact case of every character is displayed).	On the File menu, choose Save. Save the file in the C:\StorageSets\Default directory. (UNIX, AIX, Solaris): Save the file in the /home/newuser/practice directory. To back up files, click the Backup Now button. In the Save As dialog box, choose the drive, then the folder.

Table 1: Text Conventions (Continued)

Element	Convention	Examples
<ul style="list-style-type: none"> User Input and System Responses (Output and Error Messages) COMMAND NAMES Drive Names 	<p>Initial Caps and monospace font</p> <p>COMMAND NAMES appear in upper case, unless they are case sensitive (UNIX, AIX, HP-UX and Solaris command names are case sensitive and will not appear in uppercase).</p> <p>Entered <variables> are displayed in angle brackets (< >) and all lower case.</p>	<p>User Input and System Responses:</p> <ul style="list-style-type: none"> To exit from the program, type <code>Exit</code>. At the prompt, type this command: <code>SHOW THIS_CONTROLLER</code> (no variable) To see your settings, give the command: <code>SHOW <storagesets> FULL</code> (with variable) You will see the <code>Continue?</code> message. <p>Command Names</p> <ul style="list-style-type: none"> Use <code>SET THIS_CONTROLLER</code> to change parameters. To manage storage, enter <code>RUN sysmgr.exe</code> (UNIX, AIX, HP-UX, Solaris): To list files, give the <code>ls</code> command. Drive Names: Navigate to your CD-ROM drive (usually <code>D:</code> or <code>E:</code>).
<i>filenames</i>	<p>Unless case sensitive, use <i>lowercase italics</i>. If filenames are case-sensitive (UNIX, AIX, HP-UX, Solaris) or are easier to understand with some upper case letters, the exact case of each character is displayed.</p>	<p>To configure storage, edit <i>storageset.ini</i>. Changes are stored in <i>NewSystemConfigurationFile.ini</i>. (UNIX, AIX, HP-UX, Solaris): Errors are logged to <i>MixedCaseFile.txt</i>.</p>
Menu Command Sequences	<p>Initial Caps, with a right angle bracket (>) between items. Menu items are displayed as shown on screen.</p>	<p>To compare documents, choose: Tools > Documents > Compare.</p>

Table 1: Text Conventions (Continued)

Element	Convention	Examples
URLs	Sans serif font	For update notices, visit: http://www.compaq.com/products/updates

Symbols in Text

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



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



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

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

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

Getting Help

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

Compaq Technical Support

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

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

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

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

- Technical support registration number (if applicable)

- Product serial numbers
- Product model names and numbers
- Applicable error messages
- Operating system type and revision level
- Detailed, specific questions

Compaq Website

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

Compaq Authorized Reseller

For the name of your nearest Compaq Authorized Reseller:

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

Command Scripter Overview

Introduction to Command Scripter

Command Scripter is a scripting application that provides command-level control of StorageWorks systems. With Command Scripter, you can create, edit, and run script files that contain StorageWorks Command Line Interpreter (CLI) commands. This scripting capability allows automation of frequently performed StorageWorks operations.

Two interfaces are included in Command Scripter:

- Command line interface for local and LAN connections to StorageWorks controller
- Browser-based interface for centralized, remote connection

Command Scripter supports the HSZ70, HSZ80, HSG60, HSG80, and HSJ80 controllers, ACS version 8.5 and later, and can be used with or without an Agent.

NOTE: All instances of “the Agent” or “an Agent” used in this chapter refer to the SWCC HS Series Agent and HSG Element Manager.



CAUTION: Command Scripter enables you to use all available CLI commands to configure and operate supported StorageWorks subsystems. As such, it is a powerful tool that if not properly used, may misconfigure subsystems or cause the loss of data. Be sure to use caution and to consult StorageWorks documentation for the proper use of CLI commands.

Command Scripter Browser Interface

The Command Scripter has a browser interface option which is discussed in Chapter 2.

The browser interface consists of the following pages with the functionality indicated:

- Select Agent Host—Select HSG Element Manager or SWCC Server
- Select Subsystem—elect a subsystem on the server
- Run Commands and Scripts—Issue CLI commands or scripts of commands
- Script Editor—Create and edit CLI scripts

Command Scripter Command Line Interface

Command Scripter gives you the option to use the command line to execute commands and scripts. Command Line Interface functionality is discussed in Chapter 3.

Controllers Supported

SANworks Command Scripter supports the following StorageWorks array controllers:

- HSG80 ACS 8.6 (Fibre Channel)
- HSG60 ACS 8.6 (Fibre Channel)
- HSZ70 HSOF 7.7 (ultra SCSI)
- HSZ80 ACS 8.6 (ultra SCSI)
- HSJ80 ACS 8.6 (CI)

Operating Systems Supported

The following host operating systems are currently supported by SANworks Command Scripter:

- Windows NT Version 4.0, Service Pack 6A
- Windows 2000 Professional, Advanced Server, Data Center, and XP Professional
- Sun Solaris, versions 2.6, 7 and later
- Compaq Tru64 UNIX, versions 4.0F, 4.0G, and 5.0A and later

- OpenVMS, version 7.1 and later
- HP-UX, Version 11.0
- IBM AIX, Version 4.3.3

Command Scripter Configuration Requirements

The Command Scripter host, SWCC Agent host and HSG Element Manager require configuration actions before you use Command Scripter. Configuration requirements are discussed below.

Command Scripter Host Preparation

The host where Command Scripter resides must be configured before Command Scripter is used. The items in Table 1–1 must be accomplished for client configuration to allow Command Scripter to communicate with an Agent.

Table 1–1: Command Scripter Host Preparation

Step	Action	Example
1.	Install Command Scripter.	See Command Scripter Installation Card
2.	Ensure that the services file has the following entry: spagent 4999/tcp	N/A
3.	Ensure spagent service is registered with TCP/IP.	A typical port is 4999.
4.	For configuration commanding access, create the <i>host.ini</i> file by invoking <code>cmdscript -p host_name</code> . <i>host_name</i> is a host with an installed Agent.	<code>cmdscript -p host name</code>
5.	You are prompted to enter a password for the specific host you indicated.	NOTE: The password you entered MUST be the same as the password used by the Agent.
6.	Enter the password.	NOTE: If you entered the correct password, you will connect to the Agent that you specified.

SWCC Host Preparation

Table 1–2 outlines the actions required to configure a server for Command Scripter access to the StorageWorks Command Console (SWCC) HS Series Agent.

Table 1–2: SWCC Host Server Preparation

Step	Action	Example
1.	Add an entry for the Command Scripter Client in the Agent's <i>client.ini</i> file.	<ul style="list-style-type: none">• For configuration access to the SWCC HS Series Agent, entry is: <i>server name/0/2</i>.• For monitor access to the Agent, entry is: <i>server name/0/1</i>.• Only those Client nodes that are listed in the <i>client.ini</i> file will be allowed to communicate with the Agent.• One <i>client.ini</i> file resides on the host and contains the names of all clients that are allowed to communicate with the Agent.
2.	Restart the Agent on the designated host. NOTE: Refer to platform-specific SWCC documentation for restarting the HS Series Agent.	N/A

HSG Element Manager Configuration

The HSG Element Manager configuration is set up within the HSG Element Manager. Table 1–3 outlines the configuration actions required to set up Command Scripter access to the HSG Element Manager.

Table 1–3: HSG Element Manager Configuration

Step	Action	Result
1.	On the HSG Element Manager home page, click Options on the Session pane.	The HSG Management System Options page displays.
2.	Click the Agent Options button on the HSG Management System Options page.	The HSG Element Manager Agent Options page displays.
3.	Type your client host name in the Host Name field and select the CLI Config button.	N/A
4.	Click the Submit button on the HSG Element Manager Agent Options page.	Your client host name is added to the database.

Command Scripter Commands

Command Scripter commands are explained in Table 1–4. The table indicates commands that are used with an Agent only, commands used with or without an Agent, and commands used without an Agent.

Table 1–4: Command Scripter Commands

Command	Meaning	With Agent Only	With or Without Agent	Without Agent
agentver	Identifies the current Agent version	X		
hostdata	Provides data about the host where the Agent is running	X		
subsysdata	Provides attached StorageWorks subsystem data	X		

Table 1–4: Command Scripter Commands (Continued)

Command	Meaning	With Agent Only	With or Without Agent	Without Agent
status	Provides overall StorageWorks subsystem status	X		
devlist	Provides the number of devices and their names	X		
lock	Gives exclusive access to a StorageWorks subsystem	X		
unlock	Releases exclusive access to a StorageWorks subsystem	X		
cli	Specifies CLI command request		X	
waitnormal set [x] [y]	Wait for normalization of a storageset 'set' Check every 'x' seconds Command will time out after 'y' seconds		X	
delay x	Stalls Command execution for 'x' seconds		X	
quit	Exits script execution and Command Scripter		X	
print	Adds text to a script output string.		X	
version	Provides the current version of Command Scripter		X	
max wait x	Response timeout x mins			X
exit	Exits script execution and Command Scripter		X	

Command Scripter Commands - Definition, Examples, and Output

The Command Scripter executes commands and scripts to configure StorageWorks subsystems and obtain status information. Refer to the following sections for a description of the functionality of Command Scripter commands.

Agentver

The purpose of the *agentver* command is to request the current version of the Agent running on a host computer.

- Agentver command example—*cmdscript -h servername agentver*
- Output example—*STEAM Agent, Version 2.3, Build 77*

Hostdata

The *hostdata* command requests information concerning the host on which the Agent is running.

- Hostdata command example—*cmdscript -h servername hostdata*
- Output example—*myhost 586 Windows NT 4.0*

The output includes the host name, host system type, and host operating system version.

Subsysdata

The purpose of the *subsysdata* command is to request information about the subsystems associated with the host.

- Subsysdata command example—*cmdscript -h servername subsysdata*
- Output example—*1 subsys1,HSG80, HSG80, PhysicalDrive0*

Information provided by this command includes the following:

- Number of StorageWorks subsystems
- Name of StorageWorks subsystems
- Controller type
- Product ID
- Version of controller software
- Device used to communicate with the controller

Status

The *status* command requests StorageWorks subsystem status, which includes status information on array controllers and attached storage devices.

- Status command example—*cmdscript -h servername status*

- Output example—*10100000001*

The status information is displayed as 11 digits (ones and zeros) that represent status as indicated below:

- Byte 1 = *Overall Status*—the state of all Field Replacement Units (FRU) on the StorageWorks subsystem:
 - 0 = good
 - 1 = a failure occurred
- Byte 2 = *Disk Status*—the state of all drives on the StorageWorks subsystem:
 - 0 = good
 - 1 = at least one drive failure occurred
- Byte 3 = *Power Supply Status*—the state of all power supplies on the StorageWorks subsystem:
 - 0 = good
 - 1 = at least one power supply failure occurred
- Byte 4 = *Fan Status*—the state of all fans on the StorageWorks subsystem:
 - 0 = good
 - 1 = at least one fan failure occurred
- Byte 5 = *Cache Battery Status*—the state of all batteries on the StorageWorks subsystem:
 - 0 = good
 - 1 = battery in low or failed state
- Byte 6 = *Temperature Status*—the state of the StorageWorks subsystem's temperature:
 - 0 = good
 - 1 = high temperature has occurred
- Byte 7 = *THIS Controller Status*—the state of THIS controller:
 - 0 = good
 - 1 = THIS controller failed
- Byte 8 = *Communication Status*—the state of communication with the StorageWorks subsystem:

- 0 = good
- 1 = lost communication with the StorageWorks subsystem
- Byte 9 = *OTHER Controller Status*—the state of the OTHER controller on the StorageWorks subsystem:
 - 0 = good
 - 1 = OTHER controller failed
- Byte 10 = *External Input Status*—the state of any external signal to the Environmental Monitoring Unit (EMU):
 - 0 = good
 - 1 = something failed
- Byte 11 = *Logical Unit Number (LUN) Status*—the state of all LUNs on the StorageWorks subsystem:
 - 0 = good
 - 1 = something failed

Devlist

The *devlist* command provides the total number and names of the devices that are attached to the selected controller.

- *Devlist* command example—`cmdscript -h servername -s subsystem devlist`
- Output example—`3 E F G`

NOTE: In the output, 3 is the number of devices attached to the subsystem; E, F, G is the list of devices attached to the subsystem.

Lock

The *lock* command is a request from the Client to gain exclusive access to the StorageWorks subsystem and prevents other client connections to the controller through the Agent. This command may be useful for controlling access to the subsystem during multi-step configuration operations. When used with the HSG Element Manager, other applications are locked out.

The Agent validates that the Client has configuration access, then tries to lock the subsystem. If the lock is unsuccessful, an error is sent to the Client.

- Lock command example—`cmdscript -h servername -s subsystem lock`
`cmdscript -h tru300 -s hsg009tb -l2`
`hsg006t>lock`

Unlock

The Client requests the subsystem be unlocked by using the *unlock* command after a subsystem is configured, thus releasing exclusive access to the subsystem.

The Agent validates that the Client has configuration access, then tries to unlock the subsystem. If the unlock is unsuccessful, an error message is sent to the Client.

- Unlock command example—`cmdscript -h servername -s subsystem unlock`
`hsg006t>unlock`

CLI

CLI, followed by a valid StorageWorks CLI command, can be used to configure and monitor a StorageWorks subsystem. For more information on CLI commands, refer to the CLI Reference Guide for your subsystem.

This keyword is optional. Any keyword not executed by Command Scripter is passed on to the controller for execution. CLI must be used to execute a CLI command that is the same as a Command Scripter meta-command such as *Exit*.

- CLI command example—`cli show this`
- Output example:

```
Controller:
HSZ80 ZG80900159 Software V83Z-1, Hardware DX01
NODE_ID      = 0000-0000-0000-0000
ALLOCATION_CLASS = 1
SCSI_VERSION  = SCSI-2
Configured for dual-redundancy with ZG80900156
  In dual-redundant configuration
Device Port SCSI address 6
Time: 18-JUL-2000 14:51:19
```

```

Host PORT_1:
  SCSI target(s) (0, 2, 4, 6, 8, 10, 12, 14)
  Preferred target(s) (0, 4, 6)
  TRANSFER_RATE_REQUESTED = 20MHZ
  Host Functionality Mode = A
  Command Console LUN is target 0, lun 0
Host PORT_2:
  SCSI target(s) (1, 3, 5, 9, 11, 13, 15)
  Preferred target(s) (9, 11, 13, 15)
  TRANSFER_RATE_REQUESTED = 20MHZ
  Host Functionality Mode = A
Cache:
  64 megabyte write cache, version 0012
  Cache is GOOD
  No unflushed data in cache
  CACHE_FLUSH_TIMER = DEFAULT (10 seconds)
Mirrored Cache:
  64 megabyte write cache, version 0012
  Cache is GOOD
  No unflushed data in cache
Battery:
  FULLY CHARGED
  Expires:      18-JUL-2002
  NOCACHE_UPS
  
```

Waitnormal <Storage> <Poll> <Timeout>

This command causes Command Scripter to pause execution of the command script until a storage set becomes fully redundant.

- Waitnormal command example—*waitnormal dvgrpr0 60 600*

NOTE: Storage set - Name of the storage set to be monitored
 Poll - Polling interval in seconds
 Timeout - Timeout interval in seconds

- Output example:

Name	Storage set	Uses	Used by
DVGRPR0	raidset	DISK10000 DISK20000 DISK30000	D206

Switches:
 POLICY (for replacement) = BEST_PERFORMANCE
 RECONSTRUCT (priority) = NORMAL
 CHUNKSIZE = 256 blocks

State:
 RECONSTRUCT 11% complete
 DISK20000 (member 0) is RECONSTRUCTING 11% complete
 DISK10000 (member 1) is RECONSTRUCTING 11% complete
 DISK30000 (member 2) is RECONSTRUCTING 11% complete
 Size: 35529666 blocks

hsz70top>

Delay x

The *Delay* command stalls processing (inactivates Command Scripter) of commands for a specified number of seconds. This command may be used to allow the StorageWorks controller to complete actions from previous commands before Command Scripter continues.

- Delay x command example—*delay 8*

Quit

The *Quit* command allows you to exit a Command Scripter when actions are completed.

- Quit command example—*quit*

Print

The *Print* command lets you add text to the output stream of a script.

- Script with Print command example:
show d8
set d8 preferred_path = other_controller
print !setting d8 preferred_path from this_controller to other_controller! show d8

Version

The *version* command checks the current version of Command Scripter software.

- Version command example—*version*
- Output example—SANworks Command Scripter V1.0A Build 60

Max Wait x

The *max wait x* command sets the amount of time that you will wait for a CLI command to execute before a timeout occurs.

- Max wait x example—*max wait 3*

Using Command Scriptor Browser Interface

Introduction to the Browser Interface

Command Scriptor provides a browser interface for configuring and monitoring StorageWorks subsystems. With the interface, you communicate with subsystems through an Agent using an internet browser.

The Command Scriptor browser interface allows simultaneous access by multiple users. However, it cannot distinguish between multiple sessions by the same user.

Starting the Browser Interface

The following operating systems are supported:

- OpenVMS Alpha version 7.1-1h1 and later—Command Scriptor runs as a detached process. After starting Compaq Insight Manager, add the following command to *SYS\$MANAGER:SYSTARTUP_VMS.COM* after starting Compaq Insight Manager:
`$ @SYS$SPECIFIC:[WBEM.CPQCMDSCR]CMDSCRIPTWIN$START`
- Tru64 UNIX version 4.0F and later—Runs as a daemon process. Startup files located in the *CPQcmdscr/rc3.d* directory can be used to start the daemon process.
- Sun Solaris version 2.6 and later—Runs as a daemon process. Startup files located in the *CPQcmdscr/rc3.d* directory can be used to start the daemon process.
- Windows NT version 4.0 Service Pack 6A and later—Runs as a service under Windows. The service is created at installation and starts automatically when the system is booted.

Accessing Browser Interface

Use the following steps to access the Command Scriptor browser interface:

1. Invoke Microsoft Internet Explorer or Netscape Communicator.

2. Browse to `http://hostname:2301`. “Hostname” is the TCP/IP address or name string of the system running the Command Scripter window. If the browser and Command Scripter window are running on the same system, “localhost” can be used instead.
3. Click on the Command Scripter icon and login as “administrator”. The default password is “administrator”.

NOTE: For security reasons, change the default password as soon as possible.

Supported Browsers

Supported browsers include:

- Microsoft Internet Explorer version 5.0 and later
- Netscape Navigator version 4.7 and later

Browser Interface Usage

The Command Scripter browser interface can be used to do the following:

- Create scripts
- Edit scripts
- Delete scripts
- Run scripts
- Execute single commands

Scripts are stored on the host system running Command Scripter and are available to all user sessions.

Browser Interface Functionality

The Command Scripter Browser Interface provides the following functionality for the user:

- Select and change host system
- Select and change subsystems
- Run commands and scripts
- Edit scripts

The browser interface functionality is executed using the following pages:

- Select Agent Host
- Select Subsystem
- Commands and Scripts
- Script Editor

Each page has a title bar as shown in Figure 2–1.



Figure 2–1: Title Bar

The left side of the title bar displays the following information:

- HTTP Server—Server on which Command Scriptor is running
- Agent Host—Host on which the SWCC HS-Series Agent is running
- Subsystem—The StorageWorks subsystem selected on the Select Subsystem page
- Access Level—User access level (1—monitoring, 2—configuring)
- Version—Command Scriptor software version

You can display the Command Scriptor pages shown on the right side of the title bar with a left mouse click.

Select Agent Host

The Select Agent Host page (Figure 2–2) allows you to select the SWCC HS Series Agent or HSG Element Manager host for the StorageWorks subsystem that you want to configure or monitor. This selection can be changed at any time.

Steps for using the Select Agent Host page follow:

1. Choose an Agent Host from the Agent Host pulldown box and click the Logon button. The Select Subsystem page (Figure 2–3) is displayed.
- or
1. Instead of choosing the Agent Host (Step 1), enter the Agent Host name in the Agent Host field.

2. Enter the password in the Password field and click the Logon button.

If the password is accepted, the Select Subsystem page (Figure 2–3) is displayed.

The host names and encrypted passwords are contained in the *host.ini* file which is located in the *Command Scripter installation* directory. This file may be edited to delete old host names.

HTTP Server: PARDES
Agent Host: mac host
Subsystem:
Access Level: 2
System: 12.0 (Build 8)

Command Scripter
Select Agent Host

Home Page Help

Select Password Enabled Agent Host

Agent Host: mac host

Logon

OR

Enter New Agent Host Name and Password

Agent Host:

Password:

Logon

Password must be greater than 4 and less than 32 characters

Figure 2–2: Select Agent Host Page

Select Subsystem

The Select Subsystem page enables you to select a StorageWorks subsystem. The subsystem selection can be changed at any time.

To choose a subsystem:

1. Click the down arrow on the Choose Subsystem pulldown box.
2. Select a StorageWorks subsystem.
3. Click the Select button.

After you select a subsystem, the Run Commands and Scripts page (Figure 2–4) is displayed.

NOTE: If you are unable to select a subsystem, an error message will be displayed. If an error occurs, check the SWCC *client.ini* file for the host you selected and the correct access level.

This page also shows the Agent Host and your access level. Level 1 access allows you to only monitor the subsystem. Level 2 allows you to configure the subsystem.

The Select Subsystem page is shown in Figure 2–3.



Figure 2–3: Select Subsystem Page

Run Commands and Scripts

From the Run Commands and Scripts page (Figure 2–4), you can execute a Command Scripter command or run a script.

To execute a CLI command:

1. Enter a command in the CLI Command field.
2. Select the Run button.
3. Select the Clear button if you decide not to execute the command.

To run a CLI script:

1. Select a script from the Script pulldown box.
2. Select the Run button.

The controller's response to the command or script will appear in the text area. The most recently run command and response will be at the top of the page.

When entering a CLI command, it is optional to preface it with "cli". For example, a command may be entered as "show this" instead of "cli show this". Other Command Scripter commands may also be entered here (agentver, hostdata, subsysdata, status,

etc.). Scripts are stored in the *Command Scripter\scripts* directory. They may be created in the Script Editor page or created in another text editor and placed in the directory.

Command Scripter assumes that all files in this directory are CLI scripts, therefore, any file extension may be used and the scripts may be created with any text editor as long as they are simple text.

The available script names are in the Script pulldown box. If the script uses macro arguments, or parameters used for substitution of values, they may be entered in the Arguments field. Arguments do not require quotes. If a command or script takes more than a few seconds to complete, the page will display “Running” with no links to other pages or input fields. The page will automatically update itself until the command or script has completed.

NOTE: Only one command or script may run at a time.

See the section on the Script Editor page for more details on creating scripts.

NOTE: The CLI "run" command is only supported when run from a command script and the connection is through an HSG Element Manager. It is important to use the lock and unlock commands with the run command. Connections through an SWCC Agent does cannot support the run command.



Figure 2-4: Run Commands and Scripts Page

Script Editor

The Script Editor (Figure 2-5) page allows you to create new scripts and modify existing ones.

The Open Script pulldown box displays scripts contained in the *Command Scripter Scripter\scripts* directory. When you select a script from the pulldown box, it will appear in the editable text area. The script may then be saved with the same name or a different one. A new script may be created by typing in the editable text area and entering the new name in the Save As field.

NOTE: When using macro arguments in the scripts, the parameters should be referred to by: %1%, %2%, etc.

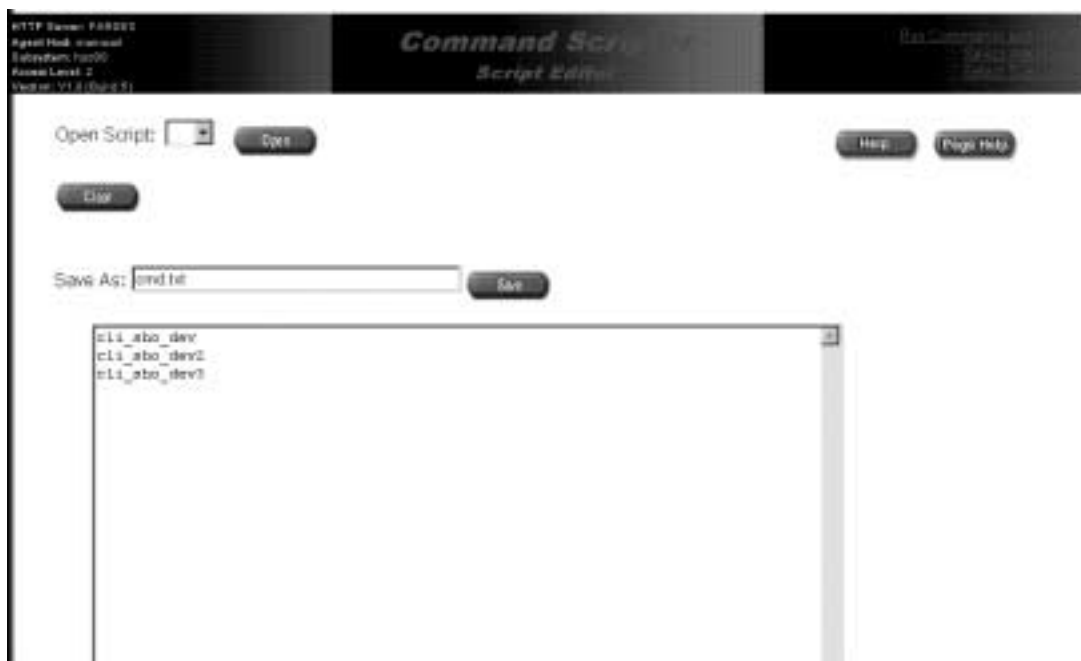


Figure 2–5: Script Editor Page

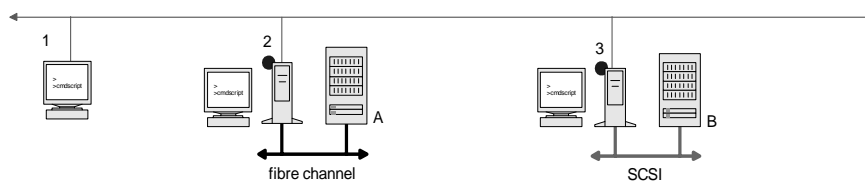
Using Command Scriptor Command Line Interface

Introduction to Command Line Interface

Command Scriptor executes CLI commands from the command line. Command Scriptor syntax can vary with the operating system. This chapter covers Command Scriptor syntax for the supported operating systems and discusses use with or without an Agent.

NOTE: All instances of “the Agent” or “an Agent” in this chapter refer to the SWCC HS Series and HSG Element Manager.

Command Scriptor commands can be executed with or without an Agent connection. The Command Scriptor and Agent can be on the same or different hosts. Ensure that the server, host, and Agent setup are accomplished as described in Tables 1-1, 1-2, and 1-3. Figures 3-1 and 3-2 show examples of how commands can be used in a StorageWorks Subsystem with or without an Agent.



Command Scriptor is installed on StorageWorks hosts - without SWCC HS-Series Agent

Computer 1

Can run CS Command Line for storage A (by login to computer 2)
Can run CS Command Line for storage B (by login to computer 3)

Legend

● Command Scriptor

Computer 2

Can run CS Command Line for storage A
Can run CS Command Line for storage B (by login to computer 3)

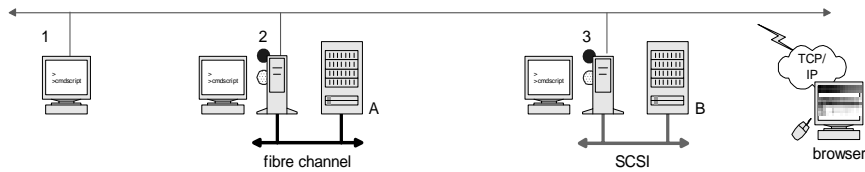
Computer 3

Can run CS Command Line for storage B
Can run CS Command Line for storage A (by login to computer 2)

Notes:

1. CS Command Line via SWCC HS-Series Agent is not available
2. CS "SWCC Commands" are not available
3. CS browser GUI is not available
4. CS = Command Scriptor

Figure 3–1: Using Command Scriptor Without an Agent



Command Scripter is installed on StorageWorks hosts - with SWCC HS-Series Agent

Computer 1

Can run CS Command Line for storage A and B (by login)
Can run CS Command Line for storage A and B (via SWCC HS-Series Agent)
Can browse to CS GUI on computers 2 and 3

Legend

● Command Scripter
○ SWCC HS-Series Agent

Computer 2

Can run CS Command Line for storage A
Can run CS Command Line for storage B (by login)
Can run CS Command Line for storage A and B (via SWCC HS-Series Agent)
Can browse to CS GUI on computers 2 and 3

Computer 3

Can run CS Command Line for storage B
Can run CS Command Line for storage A (by login)
Can run CS Command Line for storage A and B (via SWCC HS-Series Agent)
Can browse to CS GUI on computers 2 and 3

Remote via Browser

Can browse to CS GUI on computers 2 and 3

Notes:

1. CS = Command Scripter

Figure 3–2: Using Command Scripter with an Agent

Terminology

Terms used in Command Scripter syntax:

- **hostName**—Host name where the Agent resides
- **subsysName**—The storage subsystem name used by the Agent to identify a storage cabinet
- **level**—Agent command level; level 1 is used for monitoring and level 2 is for configuration
- **macro args**—Used for substitution of values into a script file
- **command**—Any CLI command
- **device name**—A physical storage device that is part of the storage subsystem

General Guidelines for Command Scripter Usage

This section addresses information and guidelines common to the operating systems supported by Command Scripter version 1.0A. Operating system specific information is covered in subsequent sections.

Command Scripter Connection Methods

Command Scripter accesses storage subsystems using the following connection methods:

- Through the SWCC Agent or HSG Element Manager
- Direct access to the controller through device drivers on the local host system using a supplied device name
- Direct access to the controller through device drivers on the local host system using a supplied controller serial number
- Direct access to controller through device drivers on the local host system using a supplied worldwide unique identifier
- Direct access to the controller through device drivers on the local host system using Bus/Target/LUN on Tru64 UNIX
- Direct access to the controller through device drivers on the local host system using a supplied SCS node name

Command Format

Each connection method has the following common elements in the command string. These elements are defined below and will not be repeated in the individual sections.

- **-m <macro args>**—Specifies a space delimited list of substitution values for insertion into a command script. This is an optional switch. If the script contains tokens in the format %number%, they will be replaced by the corresponding value from the "macro args" string. The macro arg values are shown below.
Command Scripter builds a table of tokens, and system resources determine the maximum number of tokens. Memory required increases as the number of tokens increase. Tokens can be defined until input line length is exceeded or all available memory is used. If the string contains multiple values, it must be enclosed in quotes. If tokens exist in a script and there is not a corresponding value, the token will be ignored. If there are no tokens in the script, the value string is ignored.

NOTE: The following are macro arg values:

%1% = this
%2% = other
%3% = full

Macro arg script example:

```
show %1% %3%  
show %2%  
exit
```

- **<command>**—Specifies a meta-command string or a controller CLI command string. This parameter is optional. If it is specified, Command Scripter executes the single command and exits. If it is omitted, Command Scripter runs in interactive mode. Command strings that contain spaces must be enclosed in quotes. The command parameter must be the last string on the command line.

Each connection method is discussed below.

NOTE: Each connection includes the general syntax for invoking Command Scripter. Sections discussing supported operating systems will specify additional syntax required.

Using Command Scripter with the SWCC Agent or HSG Element Manager

Command Scripter can access a storage subsystem through the SWCC Agent or the HSG Element Manager. The Agent can be running either on a local or remote host system. The HSG Element Manager runs on a SAN Appliance.

The general syntax for invoking Command Scripter when used with the Agent or HSG Element Manager is:

```
cmdscript -p <host> [-s <subsys>] [-m "<macro args>"] [-l <level>] ["<command>"]  
cmdscript -h <host> [-s <subsys>] [-m "<macro args>"] [-l <level>] ["<command>"]
```

- **-p <host>**—Specifies the TCP/IP address of the system running the Agent or the HSG Element Manager. This parameter must be used the first time a host connection is made. The user is prompted for a password used for accessing the Agent or HSG Element Manager. The password entered will be encrypted and stored in a file called *host.ini* that resides in the user's default directory. This is a required parameter and must be the first parameter in the list.
- **-h <host>**—Specifies the TCP/IP address of the system running the Agent or HSG Element Manager. This parameter is used once the access password has been entered using the "-p" switch. This switch will cause Command Scripter to

retrieve the password from the *host.ini* file before connecting to the Agent or HSG Element Manager. This is a required parameter and must be the first parameter in the list.

- **-s <subsys>**—Specifies the subsystem name for the controller to access. This is a required parameter for any command that is subsystem specific. To obtain a list of subsystems, use the "subsysdata" command.
- **-l <level>**—Specifies the access level requested for this connection. Level 1 (-l1) indicates "read-only" access which only allows you to execute CLI "show" commands. Level 2 (-l2) indicates "configuration access", which allows you to execute all CLI commands. If this parameter is omitted, -l1 is assumed.

Direct Access to Storage Subsystem Using Device Name

You can access a storage subsystem with Command Scriptor directly through the SCSI or Fibre Channel device driver using the device name or device special file name. Command Scriptor will use the supplied device name to attempt to locate a Command Console LUN (CCL) on the same controller. If a CCL is found, access to the controller will be through the CCL. Otherwise, access will be through the specified device.

The general syntax for invoking Command Scriptor using a specified device name is:

```
cmdscript -f <devicename> [-m "<macro args>"] ["<command>"]
```

- **-f <devicename>**—Specifies the device name of the device to access. On Tru64 UNIX version 5.x systems, this name is the device special file name. On other UNIX systems, this name must also include the complete directory path for the device special file. This is a required parameter and must be the first parameter in the list. To obtain a list of available devices, use the "-f subsysdata" parameter.

On Windows systems, Command Scriptor will accept a string in the format "Scsi3:1:124:0" as a device name for a controller CCL. Windows systems do not assign device names to the CCL. Therefore, this pseudo device name was implemented in Command Scriptor. The string identifies the Adapter:Bus:Target:LUN of the CCL. The "-f subsysdata" switch will generate device names in this format for each CCL.

Direct Access to Storage Subsystem Using Controller Serial Number

Command Scriptor can access a storage subsystem directly through the SCSI or Fibre Channel device driver using the controller serial number. Access will be attempted through the first available CCL. If none are available, the first available disk device is used.

The general syntax for invoking Command Scriptor using the controller serial number is:

```
cmdscript -n <serialnumber> [-m "<macro args>"] ["<command>"]
```

- **-n <serialnumber>**—Specifies the serial number of the controller. This is a required parameter and must be the first parameter in the list. To obtain a list of available devices, use the **-f subsysdata** switch.

Direct Access to Storage Subsystem Using a Worldwide Unique Identifier

Command Scriptor can access a storage subsystem directly through the SCSI or Fibre Channel device driver using the controller World Wide Unique Identifier. Access will be attempted through the first available Command Console LUN. If none are available, the first available disk device is used.

The general syntax for invoking Command Scriptor using a worldwide unique identifier is:

```
cmdscript -w <wwid> [-m "<macro args>"] ["<command>"]
```

- **-w <wwid>**—Specifies the World Wide Unique Identifier of the device to access. This is a required parameter and must be the first parameter in the list. To obtain a list of available devices, use the **-f subsysdata** switch.

Direct Access to Storage Subsystem Using Bus/Target/LUN on Tru64 UNIX

On Tru64 UNIX, Command Scriptor can access a storage subsystem directly through the SCSI or Fibre Channel device driver using the Bus/Target/LUN identifiers. Command Scriptor will attempt access through the first available Command Console LUN. If none are available, the first available disk device is used.

This feature is included for backward compatibility with previous versions. Command Scriptor does not display Bus/Target/LUN information for the available controllers. If the controller is in multibus failover mode, care should be taken to ensure the Bus/Target/LUN refers to a currently active path.

The general syntax for invoking Command Scriptor using bus/target/LUN on Tru64 UNIX is:

```
cmdsript -b <bus> -t<target> -l<lun> [-m "<macro args>"] ["<command>"]
```

- -b<bus>—Specifies the adapter identifier. This must be the first parameter.
- -t<target>—Specifies the target identifier.
- -l<lun>—Specifies the LUN identifier.

Direct Access to Storage Subsystem Using a SCS Node Name

Command Scriptor supports direct access to the controller through device drivers on the local host system using a supplied SCS node name.

This command only applies to OpenVMS systems with direct access to an HSJ controller.

Command Scriptor can access a storage subsystem directly through the SYS\$FYDRIVER driver using the controller's SCS Node Name.

The general syntax for invoking Command Scriptor using a SCS node name is:

```
cmdsript -j <scsnodename> [-m "<macro args>"] ["<command>"]
```

- -j <scsnodename>—Specifies the SCS Node Name of the controller to access. This is a required parameter and must be the first parameter in the list. The -j subsysdata switch will provide a list of available controllers.

Interactive Command Execution

If no command parameter is entered on a command line, Command Scriptor will execute commands interactively. Processing will continue until Command Scriptor encounters one of the following:

- Exit command
- Quit command
- ^C
- Fatal error

Displaying a List of Available Controllers

The -f and -j switches accept an optional keyword “subsysdata”. This keyword causes Command Scriptor to search the system for available controllers. The -f subsysdata switch will return a list of all available SCSI and Fibre Channel controllers. The -j subsysdata switch will provide a list of all available CI controllers.

Exit Status Indicators

The exit status returned by Command Scriptor indicates whether the program terminated normally or abnormally. Scripts can check the exit status to determine whether error recovery is necessary. This status does not indicate whether the controller's CLI succeeded in executing a command.

To check exit status:

- Windows systems—Zero indicates normal termination and non-zero indicates abnormal termination. To check the exit status, use the environment variable %errorlevel%.
- Unix systems—Zero indicates normal termination and non-zero indicates abnormal termination. To check the exit status, use the environment variable \$?.
- OpenVMS—SS\$_SUCCESS indicates normal termination and a low bit value indicates abnormal termination. To check the exit status, use the symbol \$STATUS.

Command Scriptor Operating System Support

Command Scriptor supports the following operating systems:

- Open VMS
- Windows NT and Windows 2000
- Compaq Tru64 UNIX
- Sun Solaris
- HP/UX
- IBM AIX

This section will discuss how CLI commands and scripts are executed both by direct access to the controller and with an Agent.

Storage subsystems can be accessed by using device name, serial number, worldwide unique identifier, SCS node name, or bus/target/LUN (for Tru64 UNIX). Device name will be used in the examples presented in the following sections addressing the operating systems supported by Command Scripter.

OpenVMS

This section outlines command syntax and procedures to follow when using Command Scripter with OpenVMS, either with or without an Agent.

Using Command Scripter Without an Agent

Invoking Command Scripter:

```
$mcr cmdscript -f "device name" "CLI command"
or
$cmdscript == "$cmdscript"
$cmdscript -f "device name" "CLI command"
```

Executing a single command:

```
$mcr cmdscript -f "access device" "command"
```

Example: `$mcr cmdscript -f "$1$gga0" "show this"`

Executing multiple commands:

```
$mcr cmdscript -f "$1$gga0" (Press Return/Enter)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show this
```

Executing a script:

```
$define sys$input dev:[scriptdir]scriptfile
$mcr cmdscript -f $1$gga0:
$deassign sys$input
```

Re-directing output to a file:

```
$define sys$output dev:[outdir] outfile
$mcr cmdscript -f "$1$gga0" "show device"
$deassign sys$output
```

Using Command Scripter with an Agent

Invoking Command Scripter:

```
$mcr cmdscript -p "host name"
```


NOTE: host name—a host with an installed Agent

Entering the password - Creates *host.ini* in the user's default directory. The encrypted password is saved in *host.ini* and is used in future invocations of Command Scripter. The password entered must be the same as the password used by the Agent.

Type "subsysdata" to show subsystem information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

```
$mcr cmdscript -h "host name" -s "subsys" -l "level" "command"
```

Example: \$mcr cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing multiple commands:

```
$mcr cmdscript -h myhost -s hsz70 -l 2 (Press Return/Enter)
cli show device (Press Return/Enter)
cli show storage (Press Return/Enter)
cli show this
exit
```

Executing a script:

```
$define sys$input cmd.txt
```

```
$mcr cmdscript -h "host name" -s "subsystem" -l "level" "command"
$deassign sys$input
```

Example: \$mcr cmdscript -h myhost -s hsz70 -l 2 <cmd.txt

Windows NT and Windows 2000

This section outlines command syntax and procedures to follow when using Command Scripter with Windows NT and Windows 2000 either with or without an Agent.

Using Command Scripter Without an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

```
cmdscript -f "device name:" "command"
```

Executing a single command:

```
cmdscript -f "e:" "show this"
```

Executing multiple commands:

```
cmdscript -f "device name" (Press Return/Enter)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show this
exit
```

Executing a script:

```
cmdscript -f "device name" < .\script
```

Re-directing output to a file:

```
cmdscript -f "device name" > .\outFile
```

Using Command Scripter with an Agent

Invoking Command Scripter:

```
cmdscript -p "host name"
```

NOTE: host name—a host with an installed Agent

Entering the password—Creates *host.ini* in the directory where Command Scripter executable resides. The password entered must be the same as the password used by the Agent.

Type "subsysdata" to show subsystems information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

```
cmdscript -h "host name" -s "subsystem" -l "level" "command"
```

Example: cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing multiple commands:

```
cmdscript -h recdr -s hsz70 -l 2 (Press Return/Enter)
cli show device (Press Return/Enter)
cli show storage (Press Return/Enter)
cli show this
```

Executing a script:

```
cmdscript -h "host name" -s "subsystem" -l "level" "command"
```

Example: cmdscript -h recdr -s hsz70 -l 2 < script.txt

Compaq Tru64 UNIX

This section outlines command syntax and procedures to follow when using Command Scripter with Compaq Tru64 UNIX either with or without an Agent.

Using Command Scripter without an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

- **Version 4.0 F and G**

```
# /var/opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: # /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rrz9c "show this"

- **Version 5x**

```
# /var/opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: # /usr/opt/CPQcmdscr/bin/cmdscript -f scp0 "show this"

Executing a single command:

```
# /var/opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: # /usr/opt/CPQcmdscr/bin/cmdscript -f dsk216c show device

Executing multiple commands:

```
cmdscript -f dsk216c (Press Return/Enter)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show other
```

Executing a script:

- **Version 4.0 F and G**

```
# /var/opt/CPQcmdscr/bin/cmdscript -f "dev name" < scriptFilePath/scriptFileName
```

Example: # /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rr219c < script.txt

- **Version 5.0 x**

/var/opt/CPQcmdscr/bin/ cmdscript -f "dev name"< scriptFilePath/scriptFileName

Example: # /var/opt/CPQcmdscr/bin/ cmdscript -f scp0 < script.txtRe-directing output to a file:

- **Version 4.0 F and G**

/var/opt/CPQcmdscr/bin/cmdscript -f /dev/rr219c show this>
outputFilePath/outputFileName

Example: # /var/opt/CPQcmdscr/bin/cmdscript -f /dev/rr219c show this > /usr/tmp/logfile

- **Version 5.0 x**

/var/opt/CPQcmdscr/bin/cmdscript -f scp0 show this > outputFilePath/outputFileName

Example: # /var/opt/CPQcmdscr/bin/cmdscript -f scp0 show this> /usr/tmp/logfile"

Using Command Scripter with an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

cmdscript -p "host name"

NOTE: host name—a host with an installed Agent

Entering the password—Creates *host.ini* in the directory where Command Scripter executable resides. The password entered must be the same as the password used by the Agent.

Type "subsysdata" to show subsystems information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

cmdscript -h "host name" -s "subsystem" -l "level" "command"

Example: cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing multiple commands:

cmdscript -h recdr -s hsz70 -l 2 (Press Return/Enter)
cli show device (Press Return/Enter)
cli show storage (Press Return/Enter)
cli show other

Executing a script:

```
cmdscript -h myhost -s hsg80 -l 2 < scriptFilePath/scriptFileName
```

Example: `cmdscript -h myhost -s hsg80 -l 2 <script.txt`

Sun Solaris

This section outlines command syntax and procedures to follow when using Command Scripter from the command line with Sun Solaris either with or without an Agent.

Using Command Scripter Without an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter

```
# /opt/CPQcmdscr/bin/cmdscript -f "device name" "command"
```

Example: `# /opt/CPQcmdscr/bin/cmdscript -f /rdsk/cltd65d052 "show this"`

Executing a single command:

```
cmdscript -f "device name" "command"
```

Example: `# /opt/CPQcmdscr/bin/cmdscript -f /rdsk/cltd65d052 show this`

Executing multiple commands:

```
cmdscript -f /rdsk/cltd65d052 (Press Return/Enter)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show other
```

Executing a script:

```
cmdscript -f /rdsk/cltd65d052 < scriptFilePath/scriptFileName
```

Example: `# /opt/CPQcmdscr/bin/cmdscript -f /rdsk/cltd65d052 < script.txt`

Redirecting output to a file:

```
# /opt/CPQcmdscr/bin/cmdscript -f /dev/rdsk/c1t2d0s2 "show this" >
outPath/outputFileName
```

Example: `# /opt/CPQcmdscr/bin/cmdscript -f /dev/rdsk/c1t2d0s2 show this > /usr/tmp/logfile`

Using Command Scripter with an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

```
cmdscript -p "host name"
```

NOTE: host name—a host with an installed Agent

Entering the password—Creates *host.ini* in the directory where the Command Scripter executable resides. The password entered must be the same as the password used by the Agent.

Type “subsysdata” to show subsystems information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

```
cmdscript -h "host name" -s "subsystem" -l "level" "command"
```

Example: cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing a script:

```
cmdscript -h myhost -s hsz70 -l 2 < scriptFilePath/script/FileName
```

Example: cmdscript -h myhost -s hsz70 -l 2 < script.txt

IBM AIX

This section outlines command syntax and procedures to follow when using Command Scripter from the command line with IBM AIX either with or without an Agent.

Using Command Scripter Without an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

```
# /opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: # /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 show this

Executing a single command:

```
# /opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: # /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 show this

Executing multiple commands:

```
cmdscript -f /dev/hdisk1 (Press Return/Enter)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show other
```

Executing a script:

```
# /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 < scriptFilePath/scriptFileName
```

Example: # /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 < script.txt

Redirecting output to a file:

```
# /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 show this > outputPath/outputFileName
```

Example: # /opt/CPQcmdscr/bin/cmdscript -f /dev/hdisk1 show this > /usr/tmp/logfile

Using Command Scripter with an Agent

From the appropriate drive and directory, invoke Command Scripter as indicated below.

Invoking Command Scripter:

```
cmdscript -p "host name"
```

Example: -p myhost

NOTE: host name—a host with an installed Agent

Entering the password—Creates *host.ini* in the directory where the Command Scripter executable resides. The password entered must be the same as the password used by the Agent.

Type "subsysdata" to show subsystems information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

```
cmdscript -h "host name" -s "subsystem" -l "level" "command"
```

Example: cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing a script:

```
cmdscript -h myhost -s hsz70 -l 2 < scriptFilePath/script/FileName
```

Example: `cmdscript -h myhost -s hsz70 -l 2 < script.txt`

HP-UX

This section outlines command syntax and procedures to follow when using Command Scriptor from the command line with HP-UX either with or without an Agent.

Using Command Scriptor Without an Agent

From the appropriate drive and directory, invoke Command Scriptor as indicated below.

Invoking Command Scriptor:

```
# /usr/opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: `# /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rdisk/c11t0d0 "show this"`

Executing a single command:

```
# /usr/opt/CPQcmdscr/bin/cmdscript -f "dev name" "command"
```

Example: `# /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rdisk/c11t0d0 "show this"`

Executing multiple commands:

```
# /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rdisk/c11t0d0)
show device (Press Return/Enter)
show storage (Press Return/Enter)
show other
```

Executing a script:

```
# /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rdisk/c11t0d0 < scriptFilePath/scriptFileName
```

Example: `# /usr/opt/CPQcmdscr/bin/cmdscript -f /dev/rdisk/c11t0d0 < script.txt`

Using Command Scriptor with an Agent

From the appropriate drive and directory, invoke Command Scriptor as indicated below.

Invoking Command Scriptor:

```
cmdscript -p "host name"
```

Example: cmdscript -p "myhost"

NOTE: host name—a host with an installed Agent

Entering the password—Creates *host.ini* in the directory where the Command Scriptor executable resides.

NOTE: The password entered must be the same as the password used by the Agent.

Type "subsysdata" to show subsystems information. For example, output from "subsysdata" would be: hsz70,HSZ70,HSZ70CCL,V77Z,\$2\$DKA0. Use the subsystem name (hsz70) in the command string.

Executing a single command:

```
cmdscript -h "host name" -s "subsystem" -l "level" "command"
```

Example: cmdscript -h myhost -s hsz70 -l 2 cli show this

NOTE: All commands with an Agent are prefaced with CLI, for example, cli show this.

Executing a script:

```
cmdscript -h myhost -s hsz70 -l 2 < scriptFilePath/script/FileName
```

Example: cmdscript -h myhost -s hsz70 -l 2 < script.txt

Index

A

- Accessing 2-1
- Agent Host 2-3
- Agent host, selecting 2-3
- Agents
 - SWCC
 - HSG Element Manager 3-1
- Agentver command 1-7
- authorized reseller, Compaq x

B

- Browser Interface 1-2
- Browser interface 2-2
 - Select Agent Host
 - Select Subsystem
 - Run Commands and Scripts
 - Script Editor 1-2
- button names, convention defined vii

C

- Caution, symbol and definition ix
- CLI command 1-10
- CLI Commands 1-5
- CLI commands
 - definition
 - examples
 - output 1-6
- CLI usage 2-5
- CLI Win Host 2-3
- Command 1-1
- Command format 3-4
- command names, convention defined viii
- Command Scriptor 1-1
- Command Scriptor Command Line 1-2

- Command Scriptor execution 3-1
- Command Scriptor interfaces 1-1
- Command scriptor with agent 3-1
- Commands and scripts, running 2-5
- Commands using command line 3-10
- Compaq
 - authorized reseller x
 - technical support ix
 - website x
- Compaq Tru64 UNIX 3-13
- Configuration requirements 1-3
- Configuring commands 1-10
- Connection methods 3-4
- Controllers Supported 1-2
- Controllers supported 1-1
- conventions
 - button names, defined vii
 - command names, defined viii
 - dialog box names, defined vii
 - directory names, defined vii
 - drive names, defined viii
 - filenames, defined viii
 - key sequences, defined vii
 - keyboard keys, defined vii
 - menu command sequences, defined viii
 - menu items, defined vii
 - named keys, defined vii
 - system responses, defined viii
 - URLs, defined ix
 - user input, defined viii

D

- Delay x command 1-12
- Devlist command 1-9

dialog box names, convention defined vii
directory names, convention defined vii
drive names, convention defined viii

E

equipment symbols ix
Execute a Cli command 2–5

F

filenames, convention defined viii

G

getting help ix
 Compaq technical support ix
 Compaq website x

H

Help, obtaining ix
Host Preparation 1–3
Hostdata command 1–7
HP/UX 3–16
HSG Element Manager Agent configuration 1–5

I

Important, defined ix
Introduction to browser interface 2–1
Introduction to command line interface 3–1
Introduction to Command Scripter 1–1

K

key sequences, convention defined vii
keyboard keys, convention defined vii

L

Lock command 1–9

M

Macro arguments 2–6, 2–7
Max Wait command 1–12
menu command sequences, convention defined viii
menu items, convention defined vii

N

named keys, convention defined vii

Note, defined ix

O

OpenVMS 3–10
Operating systems Support 1–2

P

Print command 1–12

Q

Quit command 1–12

R

Run a CLI script 2–5
Run commands and scripts 2–5
Run Commands and Scripts Page 2–7

S

Script Editor 2–7
Script Editor Page 2–8
Script Editor Window 2–7
Select Agent Host 2–3
Select Subsystem 2–4
Server, host, and agent setup 3–1
Status command 1–7
Subsysdata command 1–7
Subsystem 2–3
Subsystem, selecting 2–4
Sun Solaris 3–15
Supported controllers 1–1
symbols
 in text ix
 on equipment ix
system responses, convention defined viii

T

technical support, Compaq ix
Text editor 2–6
text symbols ix
Title Bar 2–3

U

Unlock command 1–10
URLs, convention defined ix
user input, convention defined viii

Using command scripter with Compaq Tru64
 UNIX 3–13
Using command scripter with HP/UX 3–16
Using command scripter with OpenVMS 3–10
Using command scripter with Windows NT and
 2000 3–11
Using Command Scripter without Agents 1–2
Using the browser interface 2–2

V

Version command 1–12

W

Waitnormal 1–11
Warning
 symbol and definition ix
Web interface pages 1–2
websites
 Compaq storage x
 Compaq technical support ix
Windows NT and Windows 2000 3–11

