Compaq SANworks™

Data Replication Manager HSG80 ACS Version 8.5P Scripting

User Guide

First Edition (April 2001)
Part Number: EK-DRMSC-OA. A01
Compaq Computer Corporation

Data Replication Manager HSG80 ACS Version 8.5P Scripting User Guide
First Edition (April 2001)
Part Number: EK-DRMSC-OA. A01

Contents

Chapter 2
Installation

Chapter 3
File Customization

Chapter 4
Failover and Failback with Scripts

Figures

Tables

About This Guide

This guide provides installation, configuration, and operation procedures for running failover and failback scripts in a Data Replication Manager environment.

Text Conventions

This document uses the following conventions to distinguish elements of text:

Keys	Keys appear in boldface. A plus sign (+) between two keys indicates that they should be pressed simultaneously.
USER INPUT	User input appears in a different typeface and in uppercase
Command Variables and filenames	Command variables and file names appear in italics.
Menu Options, Command Names, Dialog Box Names	These elements appear with initial capital letters.
COMMANDS, DIRECTORY NAMES, and DRIVE NAMES	These elements appear in upper case. **NOTE:** Perl commands are not case sensitive but will appear in lowercase.
Type	When you are instructed to *type* information, type the information **without** pressing the **Enter** key.
Enter	When you are instructed to enter information, type the information and then press the **Enter** key.

Symbols in Text

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

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

Compaq Technical Support

In North America, call the Compaq Technical Phone Support Center 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 the nearest Compaq Technical Support Phone Center. Telephone numbers for world wide Technical Support Centers are listed on the Compaq website. Access the Compaq website by logging on to the Internet at 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
- Add-on boards or hardware
- Third-party hardware or software

- Operating system type and revision level
- Detailed, specific questions

Compaq Website

The Compaq website has the latest information on this product. You can access the Compaq website by logging on to the Internet at:

http://www.compaq.com/storage

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.

Chapter 1

DRM Scripting Overview

Introduction

Data Replication Manager (DRM) provides a means to prevent data loss through the use of hardware redundancy and software data replication.

A DRM configuration consists of multiple storage sites. The *initiator* site carries out primary data processing. A *target* site is set up for data replication. Data processing occurs at the initiator site and data is replicated or copied to the target site. If a significant failure occurs at the initiator site, data processing can be resumed at the target site, where the data is intact.

In a DRM environment *failover* makes the data available at the target site after a failure. *Failback* moves data operations back to the initiator after the initiator site has been brought back online.

Failover and failback have normally been carried out manually by issuing a complex series of Command Line Interpreter (CLI) commands. The use of *scripts* to accomplish failover and failback greatly reduces the complexity of the procedures. The operator only needs to run the scripts, which then issue the appropriate CLI commands. However, the operator must still be able to perform a failover or failback with CLI commands if the scripts encounter an abnormal condition that prevents their satisfactory completion.

This user guide will explain:

- how to obtain and install the necessary program files
- how to customize the following files to your DRM configuration:
  - configuration generation batch files
  - target controller configuration files
  - application action list
- failover and failback planning considerations
- how to run the failover and failback batch files
- basic troubleshooting recommendations when using the scripts

Benefits

The use of scripts in a DRM environment simplifies procedures from the operator's perspective when performing failover and failback. One batch file can start an entire failover sequence. Downtime is shortened by eliminating the delay between command entries. The use of scripts also ensures that the sequence of commands has been predetermined in a calm environment, rather than during a crisis, when mistakes are more common. The result is a failover and failback process that is timely, consistent, and efficient.

Components for Scripting

Scripting requires the following components:

- The Compaq DRM Scripting Kit
- A Perl interpreter
- Compaq SANworks Command Scripter

These components are limited to the requirements listed on page 1-7. A brief description of each scripting component follows.

Compaq DRM Scripting Kit

The DRM Scripting Kit contains the batch files, Perl scripts, example files, and program files necessary for the scripts to perform failover and failback.

Perl Interpreter

Perl is the interpreted programming language in which the scripts are written. The Perl interpreter translates and processes the scripts. Every Perl script must pass through the interpreter in order to execute.

SANworks Command Scripter

The SANworks Command Scripter is application software that provides an interface to communicate the CLI commands generated by the Perl scripts to the HSG80 controllers via the Fibre Channel bus.

How the Failover and Failback Scripts Work

This section describes how the components work together to perform failover and failback by the use of scripts.

Perl Scripts

The scripts are written in the Perl programming language and reside on the host's local hard drive. For redundancy, the scripts should reside on a server on both the initiator and target sites.

User-Customized Script Support Files

The failover and failback scripts use two user-customized file types to provide variable information: a *configuration file* and an *application action list*.

- The *configuration file* tells the failover/failback scripts what devices are attached to an HSG80 controller and how the controller is configured with respect to devices and storagesets. There is one configuration file for each HSG80 controller, so there are two configuration files for a DRM initiator-target controller pair.
- The *application action list* is used by the *hsg_control.pl* Perl script to perform failover and failback actions on the specified DRM initiator-target controller pairs. An example of an action is performing the first step in a planned failover on all the listed controller subsystems.

The configuration files and the application action list are system-specific and must be tailored by the user to reflect the user's unique configuration and the user's failover and failback preferences. These files can then be used by the scripts to perform failover and failback.

Running a Script

The user invokes a failover or failback script by running a batch file from a command prompt on the system console. Figure 1–1 shows the scripting information flow after running a batch file.

1. The Perl interpreter processes the script based on the information in the configuration file and the application action list.
2. The script reads the *control table*, which controls the order of CLI commands to be issued, and sends the appropriate sequence of CLI commands (for the controller configuration specified in the configuration file) to the Command Scripter.
3. The Command Scripter then communicates the commands to the HSG80 controller over the Fibre Channel bus and relays SHOW command verification back for the scripts.

The area in Figure 1–1 within the dashed lines is further detailed in Figure 1–2 on page 1-6 to show the interaction of specific failover and failback Perl scripts.

IMPORTANT: The names of remote copy sets, stripesets, mirrorsets, raidsets, association sets, and connection names may not contain a hyphen (-). This is a Perl restriction. Underscores (_) are allowed.

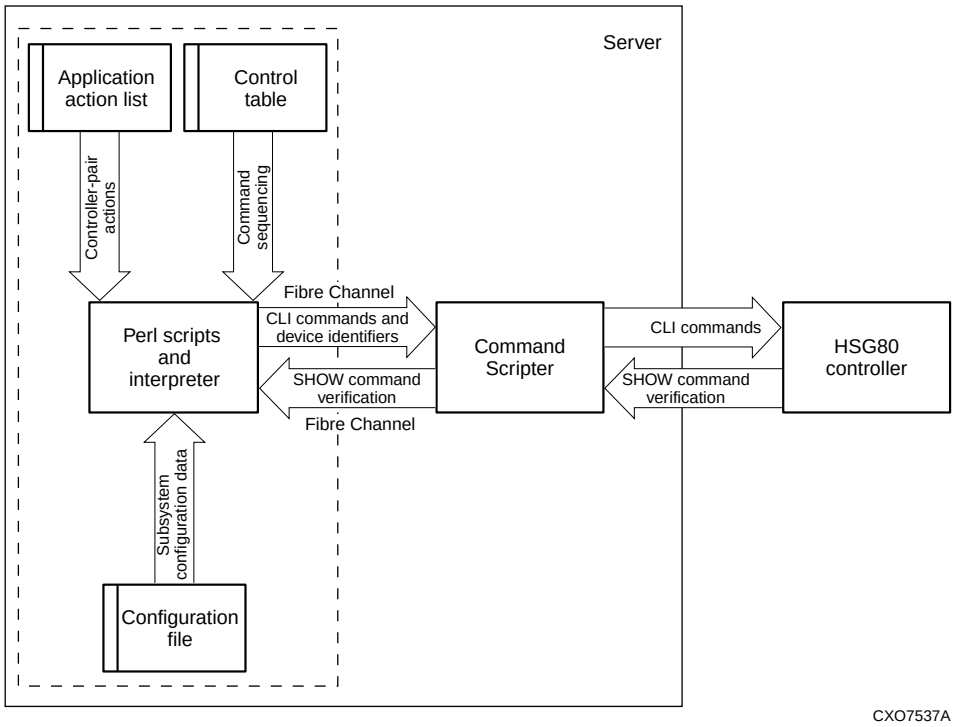

Figure 1–1 Scripting information flow

Customizing Files to a Configuration

Configuration files are created by a script that is run for each controller pair. You do not need to modify the initiator-side configuration file after it is created unless the configuration changes.

You must customize four sections in each target-side configuration file after they are created to allow the target controllers to assume the initiator role. These four sections are the association set, remote copy set, connections, and maximum cached transfer size.

You must also modify the application action list to identify those actions for controller pairs that the user wants to operate concurrently. Chapter 3, “File Customizations,” provides instructions for modifying configuration files and application action lists.

The Scripting Process Flow

Figure 1–2 shows a high-level view of the process flow for failover and failback scripts.

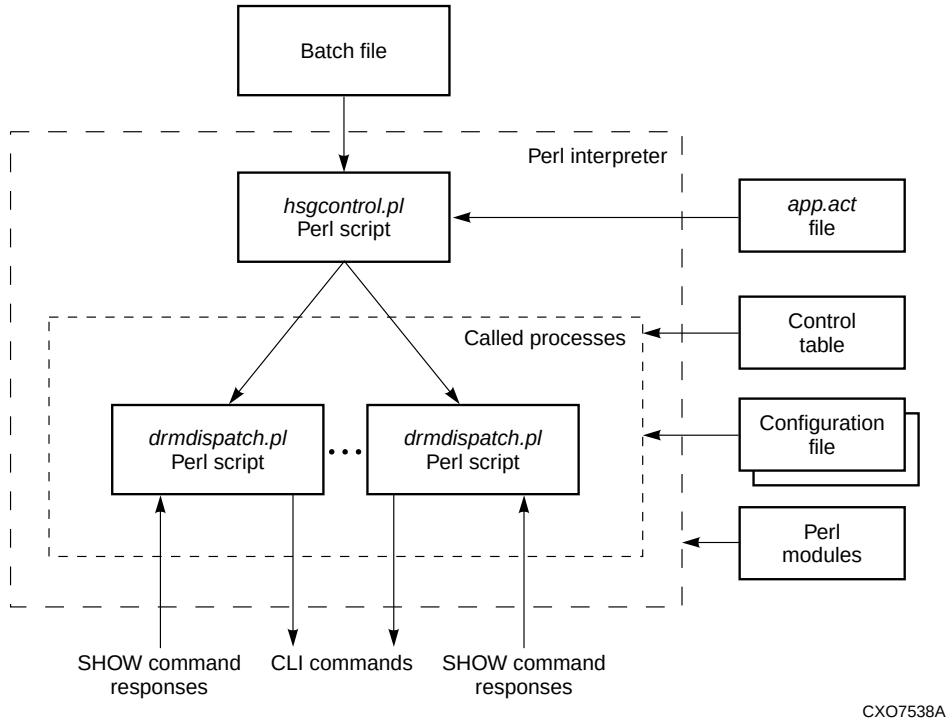

Figure 1–2 Script processing

1. The user runs a batch file to invoke the *hsgcontrol.pl* Perl script.
2. The Perl interpreter processes the scripting instructions. Parameters specified in the failover/failback batch files tell the script to read the application action list and what actions to perform. The *hsgcontrol.pl* script then calls the *drmdispatch.pl* script into action.
3. The *drmdispatch.pl* script actually performs the work of failover or failback. The script is given parameters from the application action list (*app.act* file) that specify how the actions are processed. The controller configuration files and control table are read by the *drmdispatch.pl* script and followed until all actions are performed. Perl modules, containing library routines, can be accessed by the scripts when needed.

4. The results of the *drmdispatch.pl* script are commands sent to the Command Scripter for inband transmission to the controller. SHOW command responses are then returned from the controllers and used by the scripts to verify that commands issued to the controllers were successfully executed.

Requirements

This section specifies the hardware and software required for DRM scripting.

Platforms

Supported platforms are:

- Microsoft Windows NT Server Version 4 with Service Pack 6a
- Windows 2000 Server with Service Pack 1, Advanced Server with Service Pack 1, and Datacenter Server with Service Pack 1

The scripts require a homogenous DRM environment to perform site failovers and failbacks. All servers on a DRM initiator-target pair must be running the same operating system.

Hardware

The only hardware requirement is the HSG80 controllers.

NOTE: DRM scripting cannot be used in systems with the Compaq SANworks Element Manager for HSG.

Software

The following software is required to use scripts to perform DRM failover and failback:

- Compaq DRM Scripting Kit, Version 1.0
- Perl interpreter (ActivePerl Version 5.6.0, binary kit for Win32)
- Compaq Command Scripter, Version 1.0
- Compaq Array Controller Software (ACS), Version 8.5P

Related Documentation

The documents listed below contain information relevant to failover and failback scripting.

Table 1–1 Related Documentation

Document Title	Part Number
Compaq SANworks Data Replication Manager HSG80 ACS Version 8.5P Operations Guide for Windows NT-X86	EK-DRMNT-TE. A01
Compaq SANworks Data Replication Manager HSG80 ACS Version 8.5P Operations Guide for Windows 2000 Server, Advanced Server, Datacenter Server	AA-RNV6B-TE
Compaq SANworks Command Scripter Version 1.0 User Guide	AA-RN6EA-TE
Compaq SANworks Command Scripter Version 1.0 Release Notes	AA-RN6HA-TE
Compaq SANworks Command Scripter Version 1.0 Software Product Description	AE-RN6GA-TE

Chapter 2

Installation

Introduction

This chapter discusses acquiring and installing software components necessary for the failover and failback scripting processes to run. Assuming the Data Replication Manager (DRM) platform and hardware requirements listed in Chapter 1 are already in place, the following software components are required to prepare a server attached to a Storage Area Network (SAN) for script operation:

- The Compaq DRM Scripting Kit
- A Perl interpreter
- The Compaq SANworks Command Scripter

Compaq DRM Scripting Kit

This kit contains batch files, Perl scripts, Perl modules, control tables, example files, and this user guide. The kit helps you to automate site failover and failback.

Installing the DRM Scripting Kit Files

Use the following procedure to install the DRM Scripting Kit on both the initiator server and the target server:

1. Create a directory (for example, C:\SCRIPTS) to be a default directory for the scripts.
2. Copy the scripting kit self-extracting file (*drmscript_win_v1.0.exe)* into the directory created in step 1.
3. From Windows Explorer or a command line prompt, double click or execute the *drmscript_win_v1.0.exe* file. The files will extract into the default directory.
4. Verify that the subdirectories BAT, BIN, CONFIG, LOG, and TMP were created in the default directory. See Table 2–1 for a list and description of the installed files.
5. Add an environmental variable (for example, %CLONE_HOME%) to set the default directory of the scripts.
   a. From the Windows desktop, click Start.
   b. Click Settings.
   c. Click Control Panel.
   d. Double click System.
   e. For Windows 2000 servers, click Advanced, then click Environment Variables. For Windows NT Server, click Environment.
   f. In Windows 2000, in the System Variables section, click New. In Windows NT, continue with the next step.
   g. In the dialog box, type CLONE_HOME in the Variable Name field. In the Variable Value field, enter the path to the script default directory (for example, C:\SCRIPTS).

      NOTE: When %CLONE_HOME% is used in a path name in this manual, it refers to the name you assigned to the default directory of the script files. So if you used C:\SCRIPTS as the default directory, a path name of %CLONE_HOME%\BIN would be the same as C:\SCRIPTS\BIN.

   h. In Windows 2000, click OK. In Windows NT, click Set.
   i. Click OK until you reach the Control Panel, and then close out of it.

Table 2–1 Installed DRM Scripting Kit Files

Directory	Filename	Description
Default	Hsgcs.pm Hsgcustom.pm Hsgdrm.pm Hsggen.pm Hsgwindows.pm	Perl modules containing library files used by Perl scripts. These should not be modified.
BAT	gen_ex.bat	Example file to create a configuration generation file for each controller pair.
BAT	hsg_dfb1.bat	Initiates a Perl script that performs step 1 of a disaster site failback on controllers identified in the application action list. This file should not be modified.
BAT	hsg_dfb2.bat	Initiates a Perl script that performs step 2 of a disaster site failback on controllers identified in the application action list. This file should not be modified.
BAT	hsg_dfo.bat	Initiates a Perl script that performs a two-step disaster site failover on controllers identified in the application action list. This file should not be modified.
BAT	hsg_fb1.bat	Initiates a Perl script that performs step 1 of a planned site failback on controllers identified in the application action list. This file should not be modified.
BAT	hsg_fb2.bat	Initiates a Perl script that performs step 2 of a planned site failback on controllers identified in the application action list. This file should not be modified.
BAT	hsg_fo.bat	Initiates a Perl script that performs a two-step planned site failover on controllers identified in the application action list. This file should not be modified.
BAT	hsg_nhw_dfb1.bat	Initiates a Perl script that performs step 1 of a failback on controllers identified in the application action list to a site with new hardware. This file should not be modified.
BAT	hsg_nhw_dfb2.bat	Initiates a Perl script that performs step 2 of a failback on controllers identified in the application action list to a site with new hardware. This file should not be modified.
BAT	start_perl_job.bat	Launches *drmdispatch.pl* from *hsgcontrol.pm.* This file should not be modified.

Table 2–1 Installed DRM Scripting Kit Files (Continued)

Directory	Filename	Description
BIN	drmdispatch.pl	Generates Command Line Interpreter (CLI) commands to perform failover or failback. This file should not be modified.
BIN	generate_cfg.pl	Creates the controller configuration file and saves it with a *ControllerName.cfg* filename. This file should not be modified.
BIN	hsgcontrol.pl	Reads actions from the application action list and calls the *drmdispatch.pl* Perl script to perform these named actions. This file should not be modified.
BIN	pchoice.pl	Handles user input for the batch files. This file should not be modified.
CONFIG	app_ex.act	Example file used to create an application action list.
CONFIG	failback_step1_parta.tbl	Contains step 1 failback instructions read by the *drmdispatch.pl* Perl script and performed on the initiator. This file should not be modified.
CONFIG	failback_step1_partb.tbl	Contains step 1failback instructions read by the *drmdispatch.pl* Perl script and performed on the target. This file should not be modified.
CONFIG	failback_step2.tbl	Contains step 2 failback instructions read by the *drmdispatch.pl* Perl script. This file should not be modified.
CONFIG	failover_step1.tbl	Contains step 1 failover instructions read by the *drmdispatch.pl* Perl script. This file should not be modified.
CONFIG	failover_step2.tbl	Contains step 2 failover instructions read by the *drmdispatch.pl* Perl script. This file should not be modified.
CONFIG	hsg80class.msg	Contains a translation between error codes and text messages that are displayed in the .CHK files generated by the scripts.
LOG	None	Is installed as an empty directory, but is used as a repository for .LOG and .CHK files generated by scripts. The .LOG files trace all commands and responses sent to and from the controllers. The .CHK files contain error messages.
TMP	None	Is installed as an empty directory, but is used as a repository for temporary files.

Perl Interpreter

The Perl interpreter is installed on the initiator and target hosts and is necessary to execute all Perl scripts.

Obtaining the Perl Interpreter (ActivePerl)

The Perl interpreter for the Windows platforms can be downloaded from:

http://www.activestate.com/ASPN/download/ActivePerl/

The interpreter is the ActivePerl 5.6.0.623 MSI package for Windows. Previous versions of the ActivePerl program are also available at the site.

Installing ActivePerl

Follow the Windows installation instructions located on the Activestate website listed above.

Windows NT Server users must have installed or must download Microsoft Windows Installer version 1.1 or later, and be operating with Service Pack 5 or later. No additional software is needed for Windows 2000 servers.

SANworks Command Scripter

The Command Scripter component provides the interface to communicate with the HSG80 controller via the Fibre Channel bus.

Obtaining SANworks Command Scripter

To obtain the Command Scripter, contact a reseller or Compaq account representative. Refer to “Compaq Authorized Reseller” in the “About This Guide” section for source information.

Installing SANworks Command Scripter

Use the following procedure to install the Command Scripter for Windows NT or Windows 2000 servers:

1. Insert the Command Scripter CD-ROM. The InstallShield Wizard runs automatically.

   NOTE: If the CD-ROM does not automatically run, open Windows Explorer and click the CD-ROM drive. Click the Windows folder, then click *setup.exe*.

2. From the Welcome screen, click Next.
3. The license agreement displays. Click Yes to accept the license agreement.
4. Accept the default or choose a destination for the program installation. Click Next.
5. Click Finish.
6. A SANworks Command Scripter program icon is added to the Programs menu.
7. Copy *cmdscript.exe* to %CLONE_HOME%\BIN (the subdirectory under the default directory where the DRM scripting files reside; for example, C:\SCRIPTS\BIN).
8. The Windows version of Command Scripter requires that a controller be configured on each storage subsystem for directing the inband Fibre Channel data.

   To test the connection, use the following procedure for each controller:

   a. Go to the Windows command prompt.

   b. Switch to the %CLONE_HOME%\BIN directory (where %CLONE_HOME% is the name given to the default scripts directory).

   c. Enter the following CLI command:

      cmdscript -f *ControllerDriveLetter:* "show this"

   You will see a "show this" response from the controller.

9. Take note of what drive letters correspond to each controller name. You will need these for later configuration tasks.

Chapter 3

File Customization

Introduction

After installing all the Data Replication Manager (DRM) Scripting Kit software as described in Chapter 2, you will customize the controller configuration files and application action list to work with your Storage Area Network (SAN) environment. This chapter steps you through the customizations required and explains how the script commands are constructed to perform failover and failback actions.

File Customization Steps

The following list summarizes the file configuration process that is explained in this chapter.

- Create batch files to simplify the configuration generation task (see “Creating Configuration Generation Batch Files” on page 3-2). One batch file is created for each controller pair.
- Execute the configuration batch files (created in step 1) for each controller pair. See “Running Configuration Generation Batch Files” on page 3-3.
- Use pertinent information from the initiator configuration files to copy into or otherwise modify the respective target configuration files. See “Target Controller Configuration File Customization” on page 3-4. Customizations to the target configuration files are:
  - Copying association set information from initiator configuration files to the target configuration files.
  - Copying remote copy sets from the initiator configuration files to the target configuration files.

- ❏ Identifying target-side servers that are granted access to remote copy set units following a failover.
- ❏ Modifying the maximum cached transfer size values in the target configuration files to maintain consistent values with the initiator configuration files.

- Modify the application action list (*app_ex.act* file) to specify all DRM initiator-target controller pairs in the SAN that you want to execute concurrently. See "Application Action List Customization" on page 3-7.

IMPORTANT: The names of remote copy sets, stripesets, mirrorsets, raidsets, association sets, and connection names may not contain a hyphen (-). This is a Perl restriction. Underscores (_) are allowed.

Batch File Customization

During installation, three types of batch files were extracted from the failover and failback script kit and placed in the %CLONE_HOME%\BAT directory (where %CLONE_HOME% is the default directory where the scripts reside):

- Failover and failback batch files (*hsg_fo.bat, hsg_fb1.bat,* and so on). You do not need to customize these files. Their purpose is to run the Perl script that performs a planned or disaster failover and failback.
- Subordinate batch file (*start_perl_job.bat*). This batch file is not customized. It is used by the *hsgcontrol.pl* Perl script to call the *drmdispatch.pl* Perl script.
- Configuration generation batch file (*gen_ex.bat*). You must modify this batch file to create one customized configuration generation batch file for each HSG80 controller. The resulting batch files are then run to create individual controller configuration files.

Creating Configuration Generation Batch Files

The *gen_ex.bat* file is provided as a template file to aid in creating a configuration list for each initiator and target controller pair. Edit the file by inserting the DRM controller name and device identification of the controller pair being described, and then save the file with a unique file name.

For example, we have performed these steps to create a configuration generation batch file for a controller pair named Tulsa.

1. Using a text editor, open the *gen_ex.bat* file and make the necessary modifications using the following syntax:

   Perl -I %CLONE_HOME% %CLONE_HOME%\bin\generate_cfg.pl com=cs **tulsa J:**

   where,

   tulsa identifies the DRM controller name of the subsystem.

   J: specifies the device that Command Scripter uses to communicate with controller Tulsa.

   NOTE: This must be a "sticky" drive letter, meaning that it must be persistent during restarts of the server. It also must be the drive letter of a non-RCS logical unit number (LUN). The server running the scripts uses the drive letter to direct I/O to specific HSG controllers. This means that the server running the DRM scripts must see each DRM controller it communicates with on the SAN.

2. Save the edited batch file into the %CLONE_HOME%\BAT subdirectory with a meaningful name like *tulsa_gen.bat.*

Running Configuration Generation Batch Files

After creating a configuration batch file for each controller pair, execute each of them individually.

1. In Windows Explorer, locate the configuration generation batch files in the %CLONE_HOME%\BAT directory.
2. Double click or run the batch file for the first controller.
3. Continue to run the configuration generation batch file for each initiator and target controller.

The generation batch files run a Perl script called *generate_cfg.pl*. When this script runs:

- Many SHOW commands are sent to the applicable HSG80 controller. The script creates a controller configuration file based on the received responses.
- This resulting configuration file framework is named by the script in the format *ControllerName.cfg*.

In the example above, the *tulsa_gen.bat* file would create a configuration file called *tulsa.cfg* (provided that *tulsa* is the controller name used in the generation batch file) and would place it in the %CLONE_HOME%\CONFIG directory.

Controller Configuration File Customization

The configuration files, created by running the configuration generation batch files, represent a picture in time of the controller configuration. Information in these files enable the Perl scripts to issue the correct commands. Sections in these files have names like ASSOCIATIONSET, CONNECTIONS, CONTROLLER, and so on. Appendix A shows an example of a controller configuration file.

At this time:

- The configuration files for the initiator controllers are complete when generated and do not have to be modified.
- The configuration files for the target controllers must be modified each time they are created to put them into a state that should exist after failover.

Target Controller Configuration File Customization

The target controller configuration files are built by running configuration generation batch files that execute the *generate_cfg.pl* script. However, these files require additional information to allow the target site to assume the initiator role after failover. You can copy some of this information directly from similar sections of the corresponding initiator configuration file. You will need to edit other sections manually.

The Association Set Section

This section exists in the initiator configuration file, but not in the target configuration file, because association sets do not exist on the target site when the script is executed.

To make the necessary changes:

- Using a text editor, copy the information from the initiator configuration file into the target configuration file, as shown in Figure 3-1.
- Set up a target log unit for each association set. Refer to the procedure described in the *Compaq SANworks Data Replication Manager HSG80 ACS Version 8.5P Operations Guide* for your Windows platform listed in Table 1-1.

  NOTE: Compaq recommends that the log unit on the initiator match a designated log unit on the target, and that it be set up as a write history log unit.

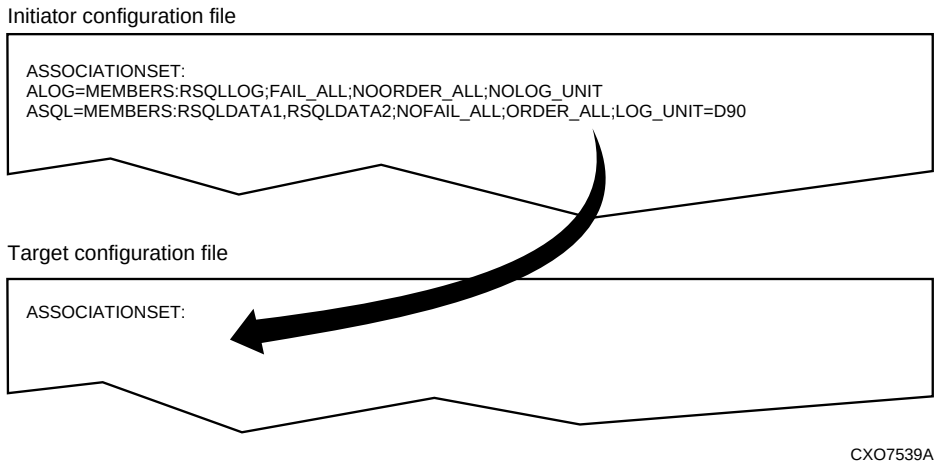

Figure 3–1 Copying association set information

The Remote Copy Set Section

This is another section that does not exist in the target configuration file because the information is not available when the script is run. Using a text editor, copy information from the corresponding initiator configuration file into the target file, as shown in Figure 3-2.

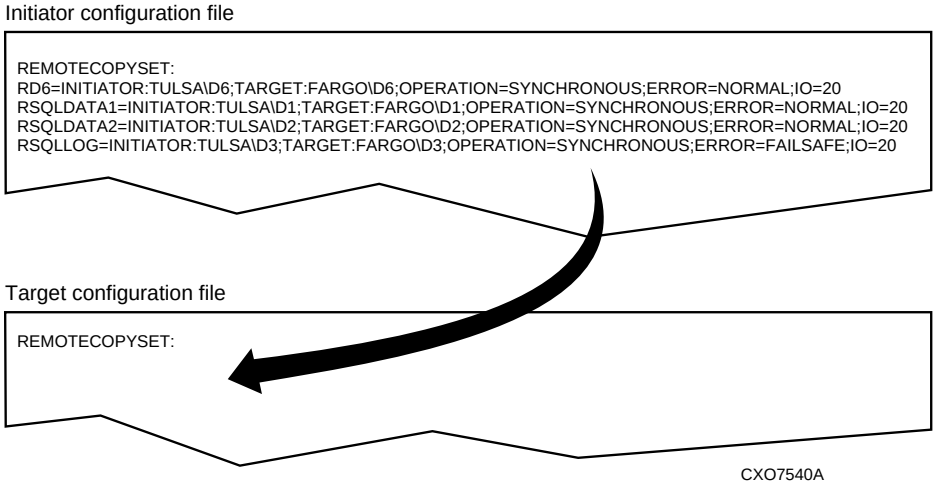

Figure 3–2 Copying remote copy set information

The Connections Section

This section in the configuration file specifies which server has access to the controllers. In a newly-created target configuration file, the information may look like the following:

```
CONNECTIONS:
D1=TULSAC,TULSAD
D2=TULSAC,TULSAD
```

Only the DRM initiator connections will have been inserted in this section. You must modify this section to enable the target-side servers access to these controllers following a site failover. This is done by modifying the target configuration file with the names of the desired server connections. Spaces between the connection names are not allowed.

Assume that SERVA_T and SERVA_B are target-side server connections to be given access to the controllers. After modifying the file with a text editor, the resulting section would look like the following:

```
CONNECTIONS:
D1=TULSAC,TULSAD,SERVA_T,SERVA_B
D2=TULSAC,TULSAD,SERVA_T,SERVA_B
```

The Maximum Cached Transfer Size Section

Default values were loaded into the created target configuration file that do not correspond to the values in the initiator configuration file. Change these values to match those of the initiator configuration file. Ensure that the units being modified are mapped to the correct remote copy sets.

For example, the newly-created initiator and target configuration files may show the following:

Initiator Configuration File	Target Configuration File
`MAXIMUM_CACHED_TRANSFER_SIZE:`	`MAXIMUM_CACHED_TRANSFER_SIZE:`
`D1=32`	`D1=1`
`D2=32`	`D2=1`
`D3=32`	`D3=1`

Continuing this example, assume that units D1, D2, and D3 on the initiator side map to units D1 through D3 on the target side. With a text editor, modify the values in the target configuration file to match the initiator as shown below:

```
MAXIMUM_CACHED_TRANSFER_SIZE:
D1=32
D2=32
D3=32
```

Application Action List Customization

During installation, the default application action list (*app_ex.act*) was extracted from the DRM Scripting Kit and placed in the %CLONE_HOME%\CONFIG directory. It provides a basic structure that you must customize using the procedures below. After customizing, you will save the file as *app.act*. For a discussion of the structure of the application action list and details of the Perl scripts that use this file, refer to Appendix B.

Steps to Customizing the Application Action List

The following steps are provided as guidelines to preparing your application action list:

1. With a text editor, open the *app_ex.act* file in the %CLONE_HOME%\CONFIG directory.
2. Identify the number of DRM initiator-target controller pairs in your DRM configuration and populate the number of actions to correspond with the number of DRM pairs. The number of pairs will match the number of entries for each action. For example, two DRM pairs will comprise two entries under the action PLANNED_FAILOVER_STEP1.
3. For each entry within an action section, modify the controller name to that of the target controller name for each DRM initiator-target pair.
4. Rename the *app_ex.act* file to be *app.act*.

Example Customization of an Application Action List

Assume that you have three DRM initiator-target controller pairs in your system called Sun-Moon, Mars-Venus, and Jupiter-Saturn.

1. You open the *app_ex.act* file with a text editor and start with the action called PLANNED_FAILOVER_STEP1. What you see is:

```
PLANNED_FAILOVER_STEP1
Background fargo drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background denver drmdispatch PLANNED failover_step1 ALL NOTFORCED
END_PLANNED_FAILOVER_STEP1
```

2. Three initiator-target controller pairs were identified in our example but the *app_ex.act* file shows only two. To add a third action, copy and paste an existing action line with a text editor. After copying another action line you would have:

```
PLANNED_FAILOVER_STEP1
Background fargo drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background denver drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background denver drmdispatch PLANNED failover_step1 ALL NOTFORCED
END_PLANNED_FAILOVER_STEP1
```

3. Change the sample DRM target controller names to the actual target controller names in all the actions, as shown below:

```
PLANNED_FAILOVER_STEP1
Background moon drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background venus drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background saturn drmdispatch PLANNED failover_step1 ALL NOTFORCED
END_PLANNED_FAILOVER_STEP1
```

4. After you have populated all the action sections with the three required action lines, save the *app_ex.act* file as *app.act.* The file customization is now complete.

Chapter 4

Failover and Failback with Scripts

Introduction

After all the required software components have been installed (Chapter 2), and the configuration files and application action lists are built and customized (Chapter 3), the Data Replication Manager (DRM) scripts can perform the automated site failover and failback process. This chapter discusses failover and failback considerations, steps to accomplish failover and failback, and offers some guidance on verifying whether the scripts worked as planned.

NOTE: Compaq strongly recommends that detailed functional testing be performed on all scripts before they are used operationally.

Power Up Data Replication Manager Systems

The procedures below outline how to power on and power off the storage subsystem after it has been configured.

CAUTION: Compaq recommends that you power up the controllers and switches at the target site before applying power to the initiator site. Powering up in the wrong sequence may cause incorrect configurations.

Power on the DRM systems in the sequence described in the following procedures.

NOTE: In this chapter, initiator site procedures appear in *shaded text*. This makes them visually different from target site procedures, which are *not shaded*.

Target Site Power Up Procedures

1. Ensure that all enclosures, switches, and cabinet power distribution units (PDUs) have their power switches in the OFF position.
2. Apply power to all PDUs.
3. Turn on the power switches for the cabinets from the target site.
4. Ensure that all controllers are on and functional.
5. Apply power to all Fibre Channel switches.

When completed, go to the Initiator Site Power Up Procedures.

Initiator Site Power Up Procedures

1. Ensure that all enclosures, switches, and cabinet PDUs have their power switches in the OFF position.
2. Apply power to all PDUs.
3. Turn on the power switches for the cabinets from the initiator site.
4. Make sure that all controllers are on and functional.
5. Apply power to all Fibre Channel switches.

Power Down Data Replication Manager Systems

Power down the DRM systems in the sequence described in the following procedures.

Initiator Site Power Down Procedures

1. Issue the following CLI commands (in this order):

   ```
   SHUTDOWN OTHER_CONTROLLER
   SHUTDOWN THIS_CONTROLLER
   ```

2. Turn off the Fibre Channel switch.
3. Turn off the power to the enclosures.
4. Turn off the PDUs.

When completed, go to the Target Site Power Down Procedures.

Target Site Power Down Procedures

1. Issue the following CLI commands (in this order):

   SHUTDOWN OTHER_CONTROLLER

   SHUTDOWN THIS_CONTROLLER

2. Turn off the Fibre Channel switch.
3. Turn off the power to the enclosures.
4. Turn off the PDUs.

Site Failover Basic Description

If the initiator site is no longer available, or if there is anticipated downtime that will prevent operation at the initiator site, you must decide whether to perform a site failover to the target site. Performing a failover enables the target site to assume the role of the initiator and access (write/read) data until the problem is resolved and a failback can be issued. Transferring control of system operation to the target site ensures that there will be minimal interruption in data access after a failure.

> **NOTE:** If you decide to perform a failover operation, keep in mind that *all* components must be failed over. Therefore, if only one component fails, fixing that single component may be preferable to performing a complete failover. Also, it is important to verify that all components at the target site are operational before you begin the site failover.

Table 4–1 outlines example scenarios that may call for a failover and those that may not.

Table 4–1 Failover Scenarios

When to Failover	When Not to Failover (recommended)
■ Both controllers fail	■ Single failed switch
■ Extended power outage at the initiator site	■ Single fiber optic cable malfunctions
■ Both host adapters fail (non-clustered hosts)	■ Single controller fails
■ Both initiator switches fail	■ Single storageset fails
■ Disaster (flooding, fire, earthquake, terrorism, etc.) that disables access to the subsystems	■ Single disk in redundant storageset fails

Table 4–1 Failover Scenarios (Continued)

When to Failover	When Not to Failover (recommended)
■ Scheduled event that will prevent computing from the initiator site for an extended period	■ Target not in normal state
■ All hosts fail	

NOTE: If one host in a multi-host environment fails, you must decide whether or not a failover is the best course of action.

When you decide that a site failover is necessary, identify which scenario best describes your situation: planned or disaster failover.

NOTE: When discussing failovers, the terms *disaster* and *unplanned* are the same. This user guide uses the term *disaster* while other documentation may use the term *unplanned*.

The planned failover procedure should be used when failover is a scheduled event. Otherwise, Compaq suggests that you use a disaster failover procedure.

CAUTION: Be sure to follow the steps outlined in the "Planned Failover Procedures" section accurately and completely, or you may incur data loss and extended downtime.

Failback Procedure Choices

During failover, the remote copy sets at the target site are in a "copy ready" state, waiting for the initiator site to become available. When a new initiator site has been established or the original one has been restored, site operation can resume after a failback procedure has been performed. This involves synchronizing data on both the initiator and target subsystems so that operation can be returned to the initiator with minimal downtime.

IMPORTANT: Verify that all components at both sites are operational before performing a failback.

The failback sequence is a scheduled event. The HSG80 Array Controller requires that a viable dual-redundant subsystem be available before a failback can take place.

IMPORTANT: Failback to a single controller configuration is not supported.

NOTE: When discussing failbacks, the terms *planned* and *simple* have the same meaning. The terms *disaster* and *full* also have the same meaning. This user guide uses the terms *planned* and *disaster* for these failbacks, while other documentation may use the terms *simple* and *full*.

Table 4–2 will help you understand which failback procedure to use in different circumstances:

Table 4–2 Types of Failover and Failback

State of the initiator controller pair	Failover type used	Failback type used	Example
Initiator site intact	Planned	Planned (or simple)	Maintenance needs to be performed at the initiator site. The site is brought back up when maintenance is complete.
Initiator site intact	Disaster	Disaster (or full)	Power goes off at initiator site. Failover is performed to the target site. Failback to the initiator is done once power is restored.
Initiator site not intact	Planned	New hardware	Routine maintenance results in a planned failover, but a problem with the equipment develops. New equipment is installed prior to failback.
Initiator site not intact	Disaster	New hardware	Lightning strike damages equipment, resulting in a disaster failover. Once new equipment is installed, a failback is performed.

Data Replication Manager Configuration Basics

The disaster-tolerant (DT) configuration that supports DRM involves two HSG80 Array Controller subsystems—one at an initiator site and one at a target site.

IMPORTANT: Because of the complexity of the configuration process, it is a good idea to have all Data Replication Manager documentation available at both sites, to eliminate confusion and to minimize the risk of error. Please follow the steps precisely in the order provided in this documentation.

Figure 4–1 depicts a basic DRM configuration.

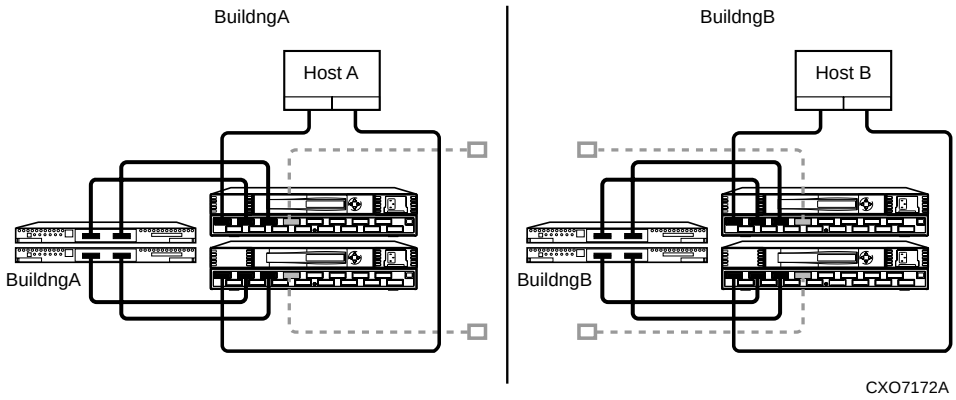

Figure 4–1 Data Replication Manager basic configuration

This figure uses fictional “Building A” as the initiator site and “Building B” as the target site. The following procedures will be described:

- Failover from Building A to Building B (planned or disaster)
- Failback step 1 from Building B back to Building A (planned, disaster, or new hardware)
- Failback step 2 from Building B back to Building A (planned, disaster, or new hardware)

NOTE: The figure refers to Building A as the initiator site and Building B as the target site. This does not change even after failover has occurred to Building B (and before failback has occurred to Building A). While in failover mode, the controllers in Building B are acting as the *initiator* for all remote copy sets and are referred to as the *target* in this document.

Notice that failback batch files are performed in two steps:

- The first step adds the initiator back into the remote copy set. It also performs a normalization if any new data was written to the target controllers while the initiator was inoperable. It is often desirable for the system to operate in this semi-failed-back state while remotely mirroring data, before reverting to the original initiator and target roles.
- The second step reverses the initiator and target roles.

Planning Considerations

The following constraints need to be considered in the initial planning of DRM failover and failback procedures.

1. If you lose intersite connections, and both the initiator and target configurations are functional, the system administrator must determine which site to use. An intersite connection could include hardware or fiber-related equipment either at the initiator or target locations.
2. If you lose all access to the target controllers for any reason, immediately remove the remote copy set targets if either of the following two conditions applies:
   - None of the remote copy sets are running with write history logging.
   - You are running with write history logging but there is a possibility the log disk may overflow.

   Use the following CLI command to remove remote copy set targets:

   SET *RemoteCopySetName* REMOVE=*TargetRemoteCopyName*\DiskName
3. If one of the initiator controllers fails, you can lose access to initiator units under the following conditions:
   - Access to target units in a remote copy set with no log disk assigned is lost for any reason (such as loss of both intersite links or loss of both target controllers),

   and

   - The target units are not removed from all of the remote copy sets.

This is because a unit will not failover between controllers in a pair (such as from a failed top controller to a functional bottom controller) if that unit is the initiator of a remote copy set that has target units assigned, and those targets are not accessible.

If this occurs, you will not be able to access or alter the unit or its remote copy set in any way from the remaining controller. Access will not be reestablished until the failed controller is either repaired or replaced. Furthermore, you will not see any apparent error message indicating that you no longer have access to the unit. To verify that units are inoperative you must check the status of all units by issuing the following command:

```
SHOW UNITS FULL
```

4. An inoperative unit will indicate the following state as part of its status display:

```
State:
                Unknown - Pending Remote Copy Set Validation
```

   This applies whether you are operating at the initiator site during normal operations or at the target site after a failover.

5. To clear this condition you must repair or replace the failed controllers, then:
   - Fix the extended link condition *or*
   - Remove the remote copy sets
6. After these conditions are met, you must restart both controllers to clear the faulted state.
7. By not removing the remote copy sets when both extended site connections are lost, you will be prohibited from moving LUNs from one HSG80 controller to the other HSG80 controller at the operating system level.

Scripting File Descriptions and Behaviors

Batch File Descriptions

The failover and failback batch files are provided in the DRM Scripting Kit and are placed in the %CLONE_HOME%\BAT directory during installation (where %CLONE_HOME% is the environmental variable indicating the default directory where the scripts reside). Each batch file contains explicit Perl commands and parameters to perform specific actions based on customizations made to the controller configuration files and application action list (refer to Chapter 3).

You should run the batch files for the desired failover or failback scenario from a command prompt window to see the status results when the script finishes. The batch files perform the following functions:

- *hsg_fo.bat* performs a planned site failover on controllers identified in the application action list.
- *hsg_fb1.bat* performs a planned site failback (step 1) on controllers identified in the application action list.
- *hsg_fb2.bat* performs a planned site failback (step 2) on controllers identified in the application action list.
- *hsg_dfo.bat* performs a disaster site failover on controllers identified in the application action list.
- *hsg_dfb1.bat* performs a disaster site failback (step 1) on controllers identified in the application action list.
- *hsg_dfb2.bat* performs a disaster site failback (step 2) on controllers identified in the application action list.
- *hsg_nhw_dfb1.bat* performs a failback (step 1) on controllers identified in the application action list to a site with new hardware.
- *hsg_nhw_dfb2.bat* performs a failback (step 2) on controllers identified in the application action list to a site with new hardware.

Verbose and Condensed Displays

You are prompted the first time a batch file is run to select either a verbose or condensed reporting display. A verbose display lists the status of the controller and all remote copy sets while the script runs (see Figure 4–2). A condensed display lists only the controller name and its status.

The verbose or condensed option is set in a .RC file, and that option will remain in effect until you change it. To change the display option, you must delete the associated .RC file. Table 4–3 lists the specific .RC files created from the failover and failback batch files. Delete the .RC file in the %CLONE_HOME%\CONFIG directory for the type of failover or failback you plan to run. You will then be prompted again for the verbose or condensed output format the next time the batch file is run. Your choice of display output will make that batch file create another .RC file to store that preference.

Try not to overburden the controllers with heavy processing (like vtdpy, FMU, and SWCC tasks) while the scripts are running.

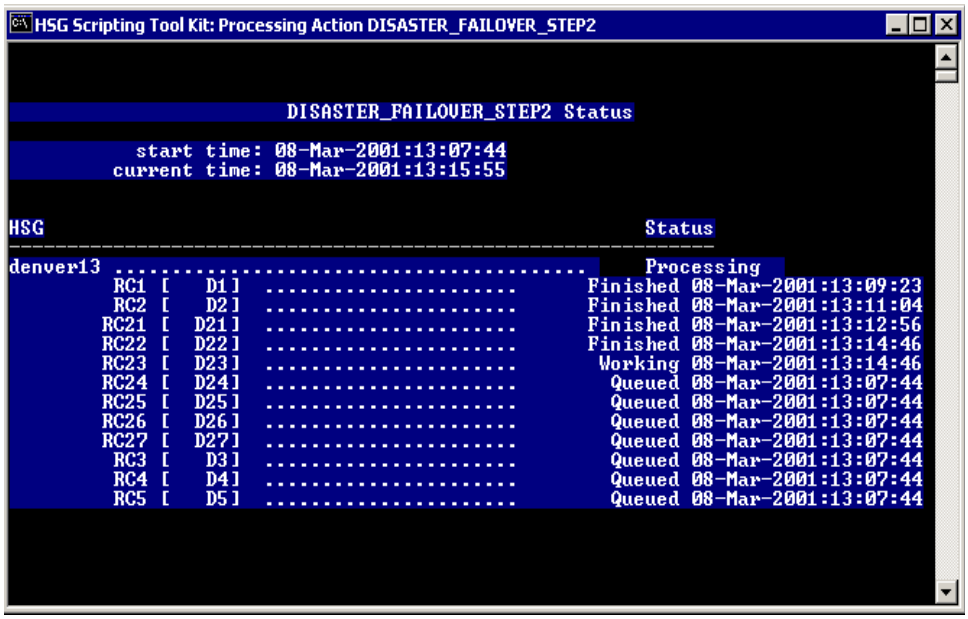

Figure 4–2 Verbose status display

Table 4–3 RC File Creation

Batch file	RC file created
hsg_fo.bat	app_fo1.rc
	app_fo2.rc
hsg_fb1.bat	app_fb1a.rc
	app_fb1b.rc
hsg_fb2.bat	app_fb2.rc
hsg_dfo.bat	app_dfo1.rc
	app_dfo2.rc
hsg_dfb1.bat	app_dfb1a.rc
	app_dfb1b.rc
hsg_dfb2.bat	app_dfb2.rc
hsg_nhw_dfb1.bat	app_dfb1b.rc
hsg_nhw_dfb2.bat	app_dfb2.rc

Terminating a Script

IMPORTANT: Compaq recommends that you do not terminate a script unless absolutely necessary. Terminating a script may leave the system in an unknown state. If this occurs, the user is responsible for putting the system in a known state.

To terminate scripts while processing, press the **Ctrl**+**C** keys. A message will appear that asks whether you want to terminate the batch job. Pressing **y** (yes) will end a script while pressing **n** (no) will cancel the request and display the command prompt. However, since some scripts are called in the background, there may be other Perl scripts running. To end these scripts:

1. Open the Windows Task Manager by pressing **Ctrl**+**Alt**+**Delete** and clicking on Task Manager.
2. Click on the Processes tab.
3. Search for all appearances of *Perl.exe* in the list. Highlight each occurrence and click on End Process.
4. Close the Windows Task Manager.

Planned Failover Procedures

The following planned failover procedures must be used in conjunction with the planned failback procedure or the new hardware failback procedure. The planned failover consists of the following three procedures:

- Initiator Site Preparation
- Running the Planned Failover Batch File
- Target Host Setup

Initiator Site Preparation Procedure

1. Before performing the failover procedure, locate your record of SHOW command output that details the current initiator configuration. Verify that your target controller configuration is the same as your initiator controller configuration.
2. If the Windows NT or Windows 2000 operating system is up and running, shut down the operating system, and power off the hosts.
3. Continue the planned failover process with the "Running the Planned Failover Batch File Procedure."

Running the Planned Failover Batch File Procedure

1. Open a command prompt window on the target host.
2. Change the directory to %CLONE_HOME%\BAT.
3. Run the *hsg_fo.bat* file.
4. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
5. If this is the first time the batch file is run, or you have deleted the .RC file, you are prompted to select a verbose or condensed reporting display. Press **v** for verbose or **c** for condensed. To change the type of display, you must delete the *app_fo1.rc* and *app_fo2.rc* files from the %CLONE_HOME%\BAT directory and run the batch file.
6. When an operation completion status result is displayed (similar to Figure 4–3), continue the failover procedure at the target site with the "Target Host Setup Procedure."

IMPORTANT: If a script encounters an abnormal condition and exits with a warning or error, abort the scripting activities and perform the remaining steps manually. These procedures are located in Chapter 5 of the applicable HSG80 ACS 8.5P operations guide listed in Table 1-1.

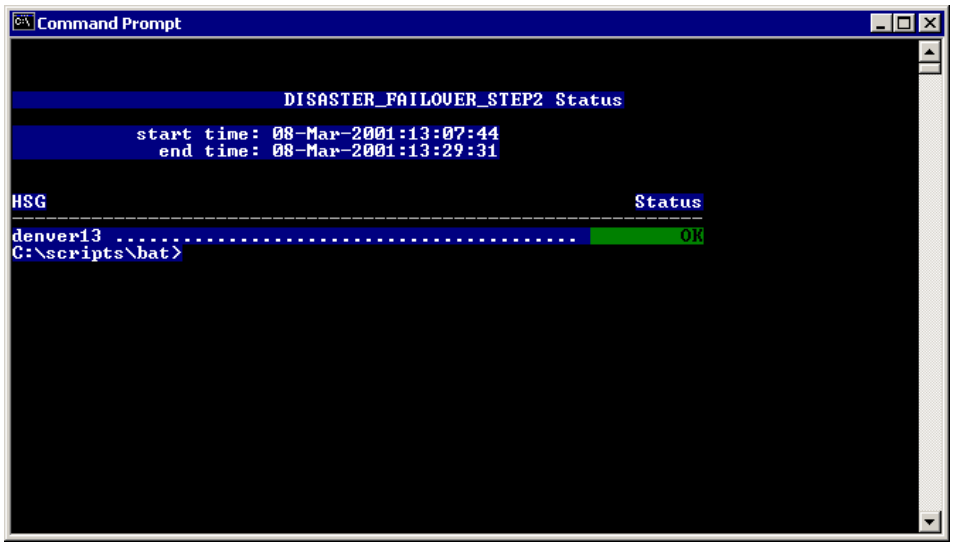

Figure 4–3 Operation completion status result display

Target Host Setup Procedure

1. To verify that failover completed successfully, issue this CLI command:

   SHOW REMOTE_COPY_SETS FULL

   The output shows you the status of remote copy sets.

   NOTE: Be sure that the units you see (listed under *Initiator State*) are at the target site.

2. To verify that the target hosts can connect to the LUNs, use this command:

   SHOW UNITS FULL

   In the Access field of the display, all units that are used by remote copy sets should show that the target hosts connections are enabled. This should also show the initiator controller connections.

3. The following step will require actions relative to each operating system being used in your configuration. For Windows 2000, proceed to step 4. For Windows NT, proceed to step 6.

4. With Windows 2000, if you **have not** changed the UNIT_OFFSET of any host connections since the hosts have been booted, you do not need to reboot the target site hosts.
   a. On each host, log in using an account that has administrative privileges.
   b. Open Computer Management and click on Disk Management.
   c. After Disk Management has initialized, go to the Action Menu and click Rescan Disks.
   d. All of the failed over units should appear in the right-hand pane. If Secure Path is not installed correctly, you will see each unit twice.
5. With Windows 2000, if you **have** changed the UNIT_OFFSET of any host connections, you must reboot that host.
   a. After the server has rebooted, log in using an account that has administrative privileges.
   b. You should be able to see all of the units in My Computer. If Secure Path is not installed correctly, you will see each drive twice.
   c. Proceed to step 7.
6. With Windows NT, allow host to recognize new units:
   a. Reboot the server(s) at the target site and log in using an account that has administrative privileges. You should be able to see all of the units in My Computer.
   b. If Secure Path is not installed correctly, you will see each drive twice.
7. This completes the planned failover procedure. The following section describes the planned failback procedure from a planned failover.

Planned Failback Procedure

The planned failback procedure is used in conjunction with the planned failover procedure. Before performing the failback procedure, locate your record of SHOW command output that details the initiator configuration. Verify that your initiator controller configuration is the same as your target controller configuration. Compare the status of the controllers, association sets, remote copy sets, units, and connections. Refer to Appendix A in the HSG80 ACS 8.5P Operations Guides (refer to Table 1-1) for the full status comparison procedure.

The planned failback consists of the following two procedures:

- Running the Planned Failback Batch Files
- Initiator Site Cleanup

Running the Planned Failback Batch Files Procedure

1. Open a command prompt window on the target host.
2. Change the directory to %CLONE_HOME%\BAT.
3. Run the *hsg_fb1.bat* file.
4. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
5. If this is the first time the batch file is run, or you have deleted the .RC file, you are prompted to select a verbose or condensed reporting display. Press **v** for verbose or **c** for condensed. To change the type of display, you must delete the *app_fb1a.rc*, *app_fb1b*, and *app_fb2.rc* files from the %CLONE_HOME%\BAT directory and run the batch file.
6. The failback script will mirror data to the initiator site to perform a normalization. The display will indicate when normalization is complete. At this time, you are disaster tolerant, and can operate in this mode until you choose to complete the failback process.

IMPORTANT: If a script encounters an abnormal condition and exits with a warning or error, abort the scripting activities and perform the remaining steps manually. These procedures are located in Chapter 5 of the applicable HSG80 ACS 8.5P operations guide listed in Table 1-1.

---

7. When you are ready to complete the failback to the original initiator site, run the *hsg_fb2.bat* file.

8. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
9. Boot all of the target hosts now, to be ready for a future failover.
10. Continue with the planned failback procedure at the initiator site with "Initiator Site Cleanup Procedure."

Initiator Site Cleanup Procedure

1. Restart the servers at the initiator site.
2. Log in using an account that has administrative privileges. You should be able to see all of the units in My Computer.
3. This completes the Planned Failback Procedure.

Disaster Failover

Use the disaster failover procedure outlined in this section in conjunction with the disaster failback or new hardware failback procedures whenever a situation occurs at the initiator site to bring it down (unable to perform its functions as an initiator).

The disaster failover consists of the following two procedures:

- Running the Disaster Failover Batch File
- Target Host Setup

Running the Disaster Failover Batch File Procedure

1. Open a command prompt window on the target host.
2. Change the directory to %CLONE_HOME%\BAT.
3. Run the *hsg_dfo.bat* file.
4. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.

5. If this is the first time the batch file is run, or you have deleted the .RC file, you are prompted to select a verbose or condensed reporting display. Press **v** for verbose or **c** for condensed. To change the type of display, you must delete the *app_dfo1.rc* and *app_dfo2.rc* files from the %CLONE_HOME%\BAT directory and run the batch file.

IMPORTANT: If a script encounters an abnormal condition and exits with a warning or error, abort the scripting activities and perform the remaining steps manually. These procedures are located in Chapter 5 of the applicable HSG80 ACS 8.5P operations guide listed in Table 1-1.

---

6. Continue with the disaster failover at the target site with "Target Host Setup Procedure."

Target Host Setup Procedure

1. To verify that all of the preceding steps have been completed successfully, issue this CLI command:

   ```
   SHOW REMOTE_COPY FULL
   ```

2. To verify that the target host can connect to the LUNs, use this command:

   ```
   SHOW UNITS FULL
   ```

3. The following steps will require actions relative to each operating system being used in your configuration. For Windows 2000, proceed to step 4. For Windows NT, proceed to step 6.
4. With Windows 2000, if you **have not** changed the UNIT_OFFSET of any host connections since the hosts have been booted, you do not need to reboot the target site hosts.
   a. On each host, log in using an account that has administrative privileges.
   b. Open Computer Management and click on Disk Management.
   c. After Disk Management has initialized, go to the Action Menu and click Rescan Disks.
   d. All of the failed over units should appear in the right-hand pane. If Secure Path is not installed correctly, you will see each unit twice.
5. With Windows 2000, if you **have** changed the UNIT_OFFSET of any host connections, you must reboot that host.
   a. After the server has rebooted, log in using an account that has administrative privileges.

b. You should be able to see all of the units in My Computer. If Secure Path is not installed correctly, you will see each drive twice.

c. Proceed to step 7.

6. With Windows NT, allow host to recognize new units:

   a. Reboot the server(s) at the target site and log in using an account that has administrative privileges. You should be able to see all of the units in My Computer.

   b. If Secure Path is not installed correctly, you will see each drive twice.

7. This completes the disaster failover procedure.

Disaster Failback Procedure

Before performing the disaster failback procedure, verify that your initiator controller configuration is the same as your target controller configuration.

Compare the status of the controllers, association sets, remote copy sets, units, and connections. Refer to Appendix A in the HSG80 ACS 8.5P Operations Guides (refer to Table 1-1) for the full status comparison procedure.

To compare status, bring up a terminal emulator session and enter a SHOW THIS command.

The disaster failback consists of the following procedures:

- Initiator Site Preparation
- Running the Disaster Failback Batch Files
- Initiator Site Target Connections Restoration

Initiator Site Preparation Procedure

1. Shut down the initiator hosts if any are still up and running.
2. If the subsystems are powered off, power on at this time.
3. Continue with the disaster failback at the target site with “Running the Disaster Failback Batch Files Procedure.”

Running the Disaster Failback Batch Files Procedure

1. Open a command prompt window on the target host.
2. Change the directory to %CLONE_HOME%\BAT.
3. Run the *hsg_dfb1.bat* file.
4. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
5. If this is the first time the batch file is run, or you have deleted the .RC file, you are prompted to select a verbose or condensed reporting display. Press **v** for verbose or **c** for condensed. To change the type of display, you must delete the *app_dfb1a.rc*, *app_dfb1b.rc*, and *app_dfb2.rc* files from the %CLONE_HOME%\BAT directory and run the batch file.
6. The failback script will mirror data to the initiator site to perform a normalization. The display will indicate when normalization is complete. At this time, you are disaster tolerant, and can operate in this mode until you choose to complete the failback process.

IMPORTANT: If a script encounters an abnormal condition and exits with a warning or error, abort the scripting activities and perform the remaining steps manually. These procedures are located in Chapter 5 of the applicable HSG80 ACS 8.5P operations guide listed in Table 1-1.

---

7. When you are ready to complete the failback to the original initiator site, run the *hsg_dfb2.bat* file.
8. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
9. Continue with the disaster failback procedure at the initiator site with “Initiator Site Target Connections Restoration Procedure.”

Initiator Site Target Connections Restoration Procedure

This section describes how to restore all target connections from the initiator site.

1. To verify that all of preceding failback steps have been completed successfully, issue this CLI command:

   ```
   SHOW REMOTE_COPY FULL
   ```

   The output shows you a list of remote copy sets. Be sure that the Initiator State field points to the initiator and the Target State field points to the target.

2. To verify that the initiator host can connect to the LUNs, use this command:

   ```
   SHOW UNITS FULL
   ```

   In the Access field of the display, all units should show that the initiator hosts are enabled.

3. The following steps will require actions relative to each operating system being used in your configuration. For Windows 2000, proceed to step 4. For Windows NT, proceed to step 6.

4. With Windows 2000, if you **have not** changed the UNIT_OFFSET of any host connections since the hosts have been booted, you do not need to reboot the target site hosts.

   a. On each host, log in using an account that has administrative privileges.

   b. Open Computer Management and click on Disk Management.

   c. After Disk Management has initialized, go to the Action Menu and click Rescan Disks.

   d. All of the failed over units should appear in the right-hand pane. If Secure Path is not installed correctly, you will see each unit twice.

5. With Windows 2000, if you **have** changed the UNIT_OFFSET of any host connections, you must reboot that host.

   a. After the server has rebooted, log in using an account that has administrative privileges.

   b. You should be able to see all of the units in My Computer. If Secure Path is not installed correctly, you will see each drive twice.

   c. Proceed to step 7.

6. With Windows NT, allow host to recognize new units:
   a. Reboot the server(s) at the target site and log in using an account that has administrative privileges. You should be able to see all of the units in My Computer.
   b. If Secure Path is not installed correctly, you will see each drive twice.
7. This completes the disaster failback procedure.

New Hardware Failback Procedure

Use the new hardware failback procedure when the initiator site is not intact, and you are working with all new hardware that is not configured.

The new hardware failback consists of the following procedures:

- Verify New Controller Communication
- Initiator Site Preparation
- Target Site Preparation (if failback is from a Planned Failover)
- Run the New Hardware Failback Batch Files

Verify New Controller Communication Procedure

1. Verify that the script server can communicate with the new controller by using the following CLI command:

   cmdscript -f *ControllerDriveLetter:* "show this"

2. When the script server and new controller demonstrate communication, continue with the "Initiator Site Preparation Procedure."

Initiator Site Preparation Procedure

1. Shut down any initiator hosts that are still up and running.
2. Manually reconfigure the controllers, but do not re-create the original remote copy sets. This procedure includes the following steps:

   NOTE: Steps b, e, and f will cause the controller pair to restart.

   a. Set node ID and checksum (this information can be found on top of the controller).

   b. Enter the following command:

      SET MULTIBUS_FAILOVER COPY = THIS

   c. Set the controller to SCSI-3 using the following CLI command:

      SET THIS_CONTROLLER SCSI_VERSION = SCSI-3

   NOTE: Do not restart the controller.

   d. Designate controller prompt names using the following CLI commands:

      SET THIS_CONTROLLER PROMPT= *"InitiatorControllerNameTop > "*

      SET OTHER_CONTROLLER PROMPT="*InitiatorControllerNameBottom > "*

   e. Set mirrored cache, using the following CLI command:

      SET THIS_CONTROLLER MIRRORED_CACHE

   f. Enable Data Replication Manager using the following CLI command:

      SET THIS CONTROLLER REMOTE_COPY = *TargetControllerName*

   g. Run the Configuration utility to assign a disk name to physical disks, using the following CLI command:

      RUN CONFIG

   h. Create and initialize storage sets and units. The units that will be part of remote copy sets must be identical to the corresponding units at the target site.

3. Disable access to all units by issuing the following CLI command:

   SET *UnitName* DISABLE=ALL

   Repeat this step for all units.

4. Set up *new units* for any additional remote copy sets that were added at the target site while failed over, by using the following CLI command:

   ADD UNIT *UnitName ContainerName* DISABLE_ALL

5. At the initiator site, make sure that the units are preferred to one controller or the other, by using the following CLI commands:

   SET *UnitName* PREFERRED_PATH = THIS_CONTROLLER

   or

   SET *UnitName* PREFERRED_PATH = OTHER_CONTROLLER

6. Set maximum cached transfer size back to 1 with the following CLI command:

   SET *UnitName* MAXIMUM_CACHED_TRANSFER_SIZE = 1

7. Set the port topology on ports 1 and 2 to fabric by using the following CLI commands:

   SET THIS (and OTHER) PORT_1_TOPOLOGY=FABRIC

   SET THIS (and OTHER) PORT_2_TOPOLOGY=FABRIC

8. Create connections to the remote target controllers. Use the CLI command:

   ADD REMOTE_COPY_SETS RCS199 D199 *InitiatorRemoteCopyName*\D199

   NOTE: This command will cause the error message "`ERROR: Initiator unit specified not found`," but it still creates and names the connections appropriately.

9. Set target access to all remote copy units by issuing the following CLI command:

   SET *UnitName* ENABLE = (*TargetRemoteCopyNameA*,*TargetRemoteCopyName*B, *TargetRemoteCopyNameC*,*TargetRemoteCopyName*D)

   Repeat this procedure for all units.

10. Compare the status of the controllers, association sets, remote copy sets, units, and connections. Refer to Appendix A in the HSG80 ACS 8.5P Operations Guides (refer to Table 1-1) for the full status comparison procedure. Make sure that any status change is reflected on the target. To make a status comparison, bring up a terminal emulator session and enter a SHOW THIS command.

11. Continue with the new hardware failback procedures at the target site (if failback is from a Planned Failover), with "Target Site Preparation Procedure from a Planned Failover."

Target Site Preparation Procedure from a Planned Failover

1. Delete the log disk by using the following CLI command:

   SET *AssociationSetName* NOLOG_UNIT

   Repeat this step for all association sets.

2. Remove the targets from the remote copy sets with the following CLI command:

   SET *RemoteCopySetName* REMOVE = *InitiatorName*\UnitName

3. Continue with the new hardware failback procedure at the initiator site with "Run the New Hardware Failback Batch Files."

Run the New Hardware Failback Batch Files

1. Open a command prompt window on the target host.
2. Change the directory to %CLONE_HOME%\BAT.
3. Run the *hsg_nhw_dfb1.bat* file.
4. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
5. If this is the first time the batch file is run, or you have deleted the .RC file, you are prompted to select a verbose or condensed reporting display. Press **v** for verbose or **c** for condensed. To change the type of display, you must delete the *app_dfb1b.rc* and *app_dfb2.rc* files from the %CLONE_HOME%\BAT directory and run the batch file.
6. The failback script will mirror data to the initiator site to perform a normalization. The display will indicate when normalization is complete. At this time, you are disaster tolerant, and can operate in this mode until you choose to complete the failback process.

   NOTE: If you are performing a new hardware failback after a planned failover, and normally use a write history log disk, Compaq recommends you add the write history log disk back into your association set if you perform normalization for longer than a day.

IMPORTANT: If a script encounters an abnormal condition and exits with a warning or error, abort the scripting activities and perform the remaining steps manually. These procedures are located in Chapter 5 of the applicable HSG80 ACS 8.5P operations guide listed in Table 1-1.

7. When you are ready to complete the failback to the original initiator site, run the *hsg_nhw_dfb2.bat* file.
8. You are presented with a message that asks you to confirm your selection. Enter a **y** (yes) to continue or an **n** (no) response to cancel the selection and return you to the command prompt.
9. If the initiator hosts are shut down, restart them at this time.
10. This completes the new hardware failback procedure.

Verifying Failover/Failback Results

It is up to the user to decide whether a failover or failback performed as desired. The following list suggests some items to check:

1. After site failover, verify that the target-side servers or hosts have proper access to any remote copy sets.
2. During site failback step 1, verify that if data is written to a target remote copy set unit, a normalization occurs to update the set.
3. After failback step 1, verify that hosts are able to continue writing to target remote copy set units, and that the write data is being mirrored to the initiator site units.
4. Following failback step 2, verify that initiator-side hosts are enabled access to remote copy set units, and that target-side hosts are disabled.

Troubleshooting Recommendations

If errors are encountered while running the scripts, some troubleshooting recommendations are provided below:

- Verify connectivity to each controller from the host by issuing the following command:

  cmdscript -f *ControllerDriveLetter:* "show this"

- Verify that all configuration files are current for each controller pair.
- Ensure that all target configuration files with manually updated sections are correctly updated with matching fields from the corresponding initiator.
- Ensure that application action files follow correct formatting and that they use correct DRM controller names.

- Inspect script-generated log files located in the %CLONE_HOME%\LOG directory. A .LOG file is created for each step in the failover/failback process, and traces the commands and responses that were sent and received from the controllers. To see error messages, look at the .CHK files that are written to this subdirectory. Refer to Appendix A for a list of all error codes, their meaning, and what actions should be taken.
- Be sure to look at the most recent .LOG and .CHK files. These files are marked with a numeric revision number with the controller name passed to *drmdispatch.pl* from the application action list.

  For example, running the DISASTER_FAILOVER_STEP1 action for controller Tulsa may result in the following files in the LOG subdirectory:

  ```
  tulsa_disaster_failover_step1.chk.1
  tulsa_disaster_failover_step1.log.1
  ```

Appendix A

Sample Controller Configuration File

This appendix provides an example of an initiator controller configuration file for a DRM initiator-target pair of Tulsa (initiator) and Fargo (target).

Example of controller configuration file

```
ASSOCIATIONSET:
ALOG=MEMBERS:RSQLLOG;FAIL_ALL;NOORDER_ALL;NOLOG_UNIT
ASQL=MEMBERS:RSQLDATA1,RSQLDATA2;NOFAIL_ALL;ORDER_ALL; LOG_UNIT=D90

CLONES:

CLONESTORAGESETCONFIG:

CONNECTIONS:
D1=TULSAC,TULSAD
D2=TULSAC,TULSAD
D3=TULSAC,TULSAD
D6=TULSAC,TULSAD
D4=None
D5=None
D90=None
D91=None
```

Example of controller configuration file (continued)

```
CONTROLLER:
CCLLUN=0
CCLid=99
Failover=MULTIBUS_FAILOVER
Firmware=V85P
SANName=FARGO
SCSI=SCSI-3
SerialNumbers=ZG83401979,ZG83401948
WWid=5000-1FE1-0000-98B0
device=J:
name=fargo

MAXIMUM_CACHED_TRANSFER_SIZE:
D1=32
D2=32
D3=32
D4=32
D5=32
D6=32
D90=32
D91=32

MIRRORSET:
HISTOTHER=DISK60300
HISTTHIS=DISK50300
LOG1=DISK20000,DISK30000
LOG2=DISK40100
SQLDATA1=DISK10000
SQLDATA2=DISK40000

PREFERRED_PATH:
D1=THIS_CONTROLLER
D2=THIS_CONTROLLER
D3=OTHER_CONTROLLER
D4=
D5=
D6=OTHER_CONTROLLER
D90=
D91=
```

Example of controller configuration file (continued)

```
RAID5SET:
R1=DISK10100,DISK20100,DISK30100

REMOTECOPYSET:
RD6=INITIATOR:TULSA\D6;TARGET:FARGO\D6;OPERATION=SYNCHRONOUS;ERROR=NORMAL;IO=20
RSQLDATA1=INITIATOR:TULSA\D1;TARGET:FARGO\D1;OPERATION=SYNCHRONOUS;ERROR=NORMAL;IO=20
RSQLDATA2=INITIATOR:TULSA\D2;TARGET:FARGO\D2;OPERATION=SYNCHRONOUS;ERROR=NORMAL;IO=20
RSQLLOG=INITIATOR:TULSA\D3;TARGET:FARGO\D3;OPERATION=SYNCHRONOUS;ERROR=FAILSAFE;IO=20

SNAPSHOTS:


STRIPESET:
S1=DISK50000,DISK60000
SQLLOG=LOG1,LOG2

TERMINALSERVER:
backup=
password=
primary=

UNIT:
D1=SQLDATA1
D2=SQLDATA2
D3=SQLLOG
D4=R1
D5=S1
D6=DISK50100
D90=HISTTHIS
D91=HISTOTHER

UNIT_IDENTIFIERS:
D1=
D2=
D3=
D4=
D5=
D6=
D90=
D91=
```

Appendix B

Structure of the Application Action List

Default Application Action List

The default application action list (*app_ex.act*) is provided in the DRM Scripting Kit and installed in the %CLONE_HOME%\CONFIG directory (where %CLONE_HOME% is the default directory where the scripts reside). It contains groupings of various failover and failback actions and lists the DRM target controllers to be included in each action. The script needs only the target controller name in an action because the script is able to determine and perform its role based on the pairing information contained in the controller configuration file.

The default application actions list is shown below. It contains actions for two separate DRM controller pairs. The two target controllers are named Fargo and Denver.

Default Application Action List

```
#
# This file specifies the actions that need to be performed on the
# HSG controllers
#
# This configuration file is used by the CLONE_HOME/bin/hsgcontrol.pl
#
# The structure of the file is as follows:
# "#"    : comment sign. Only comment signs on the 1st position of the line
#          are allowed
```

Default Application Action List (Continued)

```
# ACTION : Start of the Action specification for this application
# END_ACTION : End of Action definition for this application
#
# An action line is constructed as follows:
# Foreground|Background <ControllerName> <PerlScriptName> <Parameter1>
#<Parameter2> <Parameter3> <Parameter 4>
#
PLANNED_FAILOVER_STEP1
Background fargo drmdispatch PLANNED failover_step1 ALL NOTFORCED
Background denver drmdispatch PLANNED failover_step1 ALL NOTFORCED
END_PLANNED_FAILOVER_STEP1
#
PLANNED_FAILOVER_STEP2
Background fargo drmdispatch PLANNED failover_step2 ALL NOTFORCED
Background denver drmdispatch PLANNED failover_step2 ALL NOTFORCED
END_PLANNED_FAILOVER_STEP2
#
PLANNED_FAILBACK_STEP1_PARTA
Background fargo drmdispatch PLANNED failback_step1_parta ALL NOTFORCED
Background denver drmdispatch PLANNED failback_step1_parta ALL NOTFORCED
END_PLANNED_FAILBACK_STEP1_PARTA
#
PLANNED_FAILBACK_STEP1_PARTB
Background fargo drmdispatch PLANNED failback_step1_partb ALL NOTFORCED
Background denver drmdispatch PLANNED failback_step1_partb ALL NOTFORCED
END_PLANNED_FAILBACK_STEP1_PARTB
#
PLANNED_FAILBACK_STEP2
Background fargo drmdispatch PLANNED failback_step2 ALL NOTFORCED
Background denver drmdispatch PLANNED failback_step2 ALL NOTFORCED
END_PLANNED_FAILBACK_STEP2
```

Default Application Action List (Continued)

```
#
DISASTER_FAILOVER_STEP1
Background fargo drmdispatch DISASTER failover_step1 ALL NOTFORCED
Background denver drmdispatch DISASTER failover_step1 ALL NOTFORCED
END_DISASTER_FAILOVER_STEP1
#
DISASTER_FAILOVER_STEP2
Background fargo drmdispatch DISASTER failover_step2 ALL NOTFORCED
Background denver drmdispatch DISASTER failover_step2 ALL NOTFORCED
END_DISASTER_FAILOVER_STEP2
#
DISASTER_FAILBACK_STEP1_PARTA
Background fargo drmdispatch DISASTER failback_step1_parta ALL NOTFORCED
Background denver drmdispatch DISASTER failback_step1_parta ALL NOTFORCED
END_DISASTER_FAILBACK_STEP1_PARTA
#
DISASTER_FAILBACK_STEP1_PARTB
Background fargo drmdispatch DISASTER failback_step1_partb ALL NOTFORCED
Background denver drmdispatch DISASTER failback_step1_partb ALL NOTFORCED
END_DISASTER_FAILBACK_STEP1_PARTB
#
DISASTER_FAILBACK_STEP2
Background fargo drmdispatch DISASTER failback_step2 ALL NOTFORCED
Background denver drmdispatch DISASTER failback_step2 ALL NOTFORCED
END_DISASTER_FAILBACK_STEP2
#
```

Action Commands

Each action in the application action list begins with an action name on the first line and ends on a line with the action name preceded by the string "END_." These names must be in capital letters. For example,

```
PLANNED_FAILOVER_STEP1
(individual action lines go here)
END_PLANNED_FAILOVER_STEP1
```

The # character as the first character on a line indicates that the rest of that line is a comment. Comment symbols placed anywhere other than the first character on a line are not allowed. Empty lines are also not allowed.

An action line is constructed as follows:

```
Foreground | Background ControllerName PerlScriptName Param1 Param2 Param3 Param4
```

Refer to Table B–1 for a description of this structure.

Table B–1 Structure of an Action Command

Variable	Description
Foreground | Background	Indicates whether the action line must be executed in the foreground or background. If in the foreground, the *hsgcontrol.pl* script executes that action line and waits until the Perl script is finished before continuing with the next line. If run in the background, the *hsgcontrol.pl* script starts a background process for the action line and continues processing the next action line.
ControllerName	The name of the storage subsystem on which the action is performed. This parameter is passed, without any checking or case conversion, to the Perl script.
PerlScriptName	The name of the Perl script.
Parameters 1, 2, 3, and 4	These four parameters are used by the Perl script without any parsing or case conversion. In an action, they appear in the following order: parameter 1 = failure type parameter 2 = control table parameter 3 = remote copy sets processed parameter 4 = condition clearing

How the Perl Scripts Use the Application Action List

Two important Perl scripts are responsible for invoking failover and failback:

- The *hsgcontrol.pl* script reads the application action list (*app.act*) and passes parameters to the *drmdispatch.pl* script.
- The *drmdispatch.pl* script reads the control tables and executes the steps for failover and failback.

Since batch files initiate the failover or failback process, these Perl scripts are not visible to the user except for the reference to *drmdispatch.pl* in the application action list. The following sections discuss these two Perl scripts.

hsgcontrol.pl

The *hsgcontrol.pl* script is run from a batch file and processes the actions read from the application action list. The following is the syntax for the script. Its structure is shown in Table B–2.

```
%CLONE_HOME%\BIN\hsgcontrol.pl FileName ActionLabel RCFile
```

Table B–2 Structure of hsgcontrol.pl Script Command

Variable	Description
%CLONE_HOME%\BIN	The environmental variable pointing to the default directory of the script files and the BIN subdirectory. This is the path to the *hsgcontrol.pl* script.
hsgcontol.pl	The Perl script name.
FileName	The name of the application action list that contains the actions to be performed.
ActionLabel	The named action to be performed from the application action list. For example, PLANNED_FAILOVER_STEP1
RCFile	This file sets the user's preference to receive status reporting in either "verbose" or "condensed" mode. A discussion of these preferences are provided in Chapter 4.

An example of the *hsgcontrol.pl* Perl script command executed by a batch file is:

```
Perl -I %CLONE_HOME% %CLONE_HOME%\bin\hsgcontrol.pl app PLANNED_
FAILOVER_STEP1 app_fo1.rc
```

The script would search for a section named PLANNED_FAILOVER_STEP1 in the *app.act* file and perform all action entries between PLANNED_FAILOVER_STEP1 and END_PLANNED_FAILOVER_STEP1.

drmdispatch.pl

As *hsgcontrol.pl* performs the actions in the application list, it calls the *drmdispatch.pl* script and passes the parameters specified in the application action list. The following is the syntax for the script. Its structure is shown in Table B–3.

```
%CLONE_HOME%\BIN\drmdispatchl.pl ControllerName FailoverType ControlTable
RCSProcessed Forced_NotForced ErrorLog
```

Table B–3 Structure of drmdispatch.pl Script Command

Variable	Description
%CLONE_HOME%\BIN	The environmental variable pointing to the default directory of the script files and the BIN subdirectory. This is the path to the *drmdispatch.pl* script.
drmdispatch.pl	The Perl script name.
ControllerName	The name of the controller receiving the action.
FailoverType	Specifies either a PLANNED or DISASTER failover.
ControlTable	Identifies the control table interpreted by the script. These control tables have a *.tbl* extension and should not be modified.
RCSProcessed	A list of the remote copy sets to process. This should be set to ALL.
Forced_NotForced	Indicates whether the script must clear conditions before deleting a unit. This should be set to NOTFORCED, which requires manually clearing conditions via the storage system.
ErrorLog	An optional parameter that specifies the error log file when the script is run.

An example of the *drmdispatch.pl* Perl script command called by the *hsgcontrol.pl* script with parameters passed by the app.act files is:

```
Perl %CLONE_HOME%\bin\drmdispatch.pl fargo PLANNED failover_step1 ALL NOTFORCED Er
```

This script would start a planned failover using the steps defined in the failover_step 1 control table. All remote copy sets would be processed. Error conditions for a controller would not be cleared by the script. Error messages would be placed in a log file called *Er* in the %CLONE_HOME%\Log directory, where %CLONE_HOME% is the default directory of the scripts.

Appendix C

Scripting Error Codes

This appendix describes error codes that could be encountered by using the Perl scripts. Table C–1 lists all error codes, their meaning, and what actions should be taken.

Table C–1 Scripting Error Codes

Error Code	Meaning	Action
1120	%new - cannot read controller configuration file	Check presence, and file access to, specified configuration file.
1180	%delete_unit - cannot read unit, or unit does not exist	Check unit and underlying storage for possible problems. Correct any error conditions detected on the controller. Verify connectivity with the controller, and repair error conditions as necessary.
1181	%delete_unit - cli command error	Check unit and underlying storage for possible problems. Correct any error conditions detected on the controller.
1240	%get_subclone_list - source unit does not exist	A mismatch exists between configuration file and the actual controller configuration. Check unit section or clone section for discrepancies, and repair.
1241	%get_subclone_list - cannot read source unit	Verify connectivity with the controller and repair error conditions, as necessary. Check unit specified in the error message and verify that it is a valid unit on the controller.

Table C–1 Scripting Error Codes (Continued)

Error Code	Meaning	Action
1242	%get_subclone_list - cannot match storage type on source unit	Configuration problem on unit specified. Update configuration file.
1320	%reconstructing - cannot read mirror properties	Verify specified mirror set on controller and correct as necessary.
1422	%cli - cannot send command	Communications error. Identify communications problem and correct.
1500	%get - cannot match show output to storage type	Communications error. Identify communications problem and correct.
1520	%parse_mirror_info - unknown header information	Communications error. Identify communications problem and correct.
1521	%parse_mirror_info - cannot parse mirror name	Communications error. Identify communications problem and correct.
1522	%parse_mirror_info - cannot parse mirror membership info	Communications error. Identify communications problem and correct.
1524	%parse_mirror_info - cannot parse mirror size	Communications error. Identify communications problem and correct.
1540	%parse_unit_info - unknown header information	Communications error. Identify communications problem and correct.
1541	%parse_unit_info - cannot parse unit name	Communications error. Identify communications problem and correct.
1542	%parse_unit_info - cannot parse unit size	Communications error. Identify communications problem and correct.
1560	%parse_storage_type - unknown header information	Communications error. Identify communications problem and correct.
1561	%parse_storage_type - unknown storage information	Communications error. Identify communications problem and correct.
1580	%get_size - cannot read size of the disk	Communications error. Identify communications problem and correct.
1644	%connect - no controller prompt	Communications error. Identify communications problem and correct.
2000	%drm_site_failover - cannot read configuration entry for remote copy set	Configuration file problem. Verify correctness of the remote copy set section to the remote copy set specified.

Table C–1 Scripting Error Codes (Continued)

Error Code	Meaning	Action
2003	%drm_site_failover - cannot failover unit. SITE_FAILOVER command failed	A site failover was attempted five times. Controller problem with site failover of remote copy set. Identify and correct problem.
2010	%drm_move_initiator - cannot read configuration entry for remote copy set	Possible mismatch between configuration file and controller. Verify that the configuration file accurately reflects configuration on controller. Possible communication error. Verify communication with controller and correct problem as needed.
2011	%drm_move_initiator - cannot move initiator role to target in remote copy set	Possible communications or controller problem. Identify and correct problem.
2012	%drm_move_initiator - initiator role for remote copy set is already specified to another controller and/or unit. Mixed up initiator and target.	Possible mismatch between configuration file and controller. Verify that the configuration file accurately reflects configuration on controller.
2021	%add-rcs_to_assocset - configuration file not up to date	Mismatch between configuration file and controller configuration. Update configuration file.
2022	%add_rcs_to_assocset - cannot add association set	Attempted adding association set five times. There is a controller problem with adding the association set. Identify and correct controller problem.
2023	%add_whl_to_assocset - cannot add WHL to association set	Possible mismatch between configuration file and controller. Verify that the configuration file accurately reflects configuration of controller.
2030	%remove_rcs_from_assocset - the association set does not exist	Association set does not exist on controller but does exist in configuration file. Update configuration file.
2040	%drm_create_remotecopyset - a storageset, remote copy set, or association set already exists	Name conflict exists. Verify and correct the configuration filename.
2041	%drm_create_remotecopyset - cannot read configuration entry for remote copy set	Cannot find entry of remote copy set in configuration file. Update the configuration file.

Table C–1 Scripting Error Codes (Continued)

Error Code	Meaning	Action
2042	%drm_create_remotecopyset - cannot create the remote copy set	Cannot run add remote command. Check for possible communications errors and repair.
2050	%drm_delete_remotecopyset - the remote copy set does not exist	The remote copy set does not exist on the controller. Update the configuration file.
2051	%drm_delete_remotecopyset - the remote copy set is still part of an association set	Mismatch between the configuration file and controller configuration. Configuration file indicates remote copy set should be part of the association set, but controller indicates it is not part of the association set. Update configuration file.
2052	%drm_delete_remotecopyset - remote copy set still exists after removal	Cannot delete remote copy set. Check for communication error and repair as needed. Also possible issue with error mode on controller. If in failsafe, set the error mode to normal.
2060	%drm_add_target_unit - the remote copy set does not exist	Remote copy set does not exist on the controller. Update the configuration file.
2061	%drm_add_target_unit - the remote copy set already has a target. Delete that first.	Remote copy set is present but already has a target configured. There is a mismatch between the configuration file and controller configuration. Update the configuration file.
2062	%drm_add_target_unit - cannot add target to existing remote copy set. Check connection to remote site	Check target unit status for access problems and verify the path to remote site is functional.
2070	%drm_remove_target_unit - the remote copy set does not exist	The remote copy set does not exist on the controller. Update the configuration file.
2071	%drm_remove_target_unit - mismatch between configuration file and current controller setup	The controller and its configuration file specify a target unit, but conflict with each other. Update the configuration file.
2072	%drm_remove_target_unit - target unit still there after removal	Communication error. Identify communication problem and correct.
2080	%drm_grant_server_access - connection configured for unit does not exist in controller	Mismatch between configuration file and controller. Connection no longer exists on controller. Update configuration file.

Table C–1 Scripting Error Codes (Continued)

Error Code	Meaning	Action
2090	%drm_deny_server_access - connection configured for unit does not exist in controller	Mismatch between configuration file and controller. Connection no longer exists on controller. Update configuration file.
2100	%parse_associationset_info - unknown header information	Communication error. Identify communication problem and correct.
2101	%parse_associationset_info - cannot parse association set name	Communication error. Identify communication problem and correct.
2110	%parse_remotecopyset_info - unknown header information	Communication error. Identify communication problem and correct.
2111	%parse_remotecopyset_info - cannot parse remote copy set name	Communication error. Identify communication problem and correct.
2120	%drm_read_associationset_config - association set configuration information missing	No association set defined in configuration file. Update configuration file.
2130	%drm_read_remotecopyset_config- remote copy set configuration information missing	No remote copy set defined in configuration file. Update configuration file.
2140	%drm_change_hostport_topology - invalid port number	Failover or failback control file problem. File is possibly corrupt. Restore control file.
2141	%drm_change_hostport_topology - invalid topology	Failover or failback control file problem. File is possibly corrupt. Restore control file.
2160	%drm_change_unit_characteristic - cannot change characteristic for unit	Controller problem. Check the status of the unit and repair as needed.
2161	%drm_change_unit_characteristic - incorrect value for characteristic. Action continues, but using defaults.	Possible controller problem. Check unit status and repair as needed. Possible configuration problem. Repair file as needed.
2180	%drm_upload_saved_config	Communications error. Identify communication problem and correct.
2200	%drm_deny_remote_hsg_access - cannot remove access	Communications error. Identify communication problem and correct.
2201	%drm_deny_remote_hsg_access - cannot read configuration entry for remote copy set	Configuration file problem. Update configuration file.

Table C–1 Scripting Error Codes (Continued)

Error Code	Meaning	Action
2202	%drm_deny_remote_hsg_access - remote copy set does not contain this controller	Configuration file problem. Node specified is not an initiator or target. Update configuration file.
2210	%drm_grant_remote_hsg_access - cannot grant access	Communication error. Identify communication problem and correct.
2211	%drm_grant_remote_hsg_access - cannot read configuration entry for remote copy set	Configuration file problem. Update configuration file.
2212	%drm_grant_remote_hsg_access - remote copy set does not contain this controller	Configuration file problem. Node specified is not an initiator or target. Update configuration file.
2220	%parse_connection_info - unknown header information	Communication error. Identify communication problem and correct.
2221	%parse_connection_info - cannot parse connection information	Communication error. Identify communication problem and correct.
2231	%change_rcs_characteristic - cannot change characteristic due to incorrect value	Control table problem. Table has been possibly corrupted. Restore control table.
2232	%change_rcs_characteristic - cannot change characteristic for remote copy set on HSG controller	Problem is with remote copy set on controller. Correct error on controller as needed.

Glossary

This glossary defines terms associated with the use of scripts to perform failover and failback in a Data Replication Manager (DRM) environment. It is not a comprehensive glossary of computer terms.

ACS

See array controller software.

ActivePerl

The Perl interpreter used by the DRM Perl scripts.

application action list

A file that controls multiple instances of an action across DRM pairs (initiator-target pairs). An application action list is specific for one failover/failback application and specifies the actions that are to be performed on an initiator-target pair.

array controller

See controller.

array controller software (ACS)

Software that is contained on a removable PCMCIA program card that provides the operating system for the array controller. Also known by the acronym *ACS*.

association set

A group of remote copy sets that share common attributes. Members of an association set can be configured to transition to the same state at the same time. An association set:

- Shares the same log unit
- Has its host access removed from all members when one member fails
- Keeps I/O order across all members

asynchronous mode

A mode of operation of the remote copy set whereby the write operation provides command completion to the host after the data is safe on the initiating controller, and prior to the completion of the target command.

Asynchronous mode can provide greater performance and faster response time, but the data on all members at any one point in time cannot be assumed to be identical. *See also* synchronous mode.

batch file

A text file containing operating system commands that are used to invoke Perl scripts. The failover and failback batch files are categorized as either *planned* or *disaster.* The batch file *hsg_fo.bat*, for example, performs a planned site failover on controllers identified in the application action list.

CLI

See command line interpreter.

CLI command

CLI commands allow users to manage their subsystems by viewing and modifying the controller and its attached devices. A primary function of CLI commands is to control the failover mode of a controller pair.

CLI SHOW command

See SHOW commands.

clone

A utility that duplicates data on any unpartitioned single-disk unit, stripeset, mirrorset, or striped mirrorset in preparation for backup.

command line interpreter (CLI)

The configuration interface that operates the controller software. Also known as *command line interface*.

Command Scripter

Application software that provides an interface to communicate the CLI commands generated by the Perl scripts to the HSG80 controllers via the Fibre Channel bus. With Command Scripter, users can edit and run script files that contain CLI commands.

condensed display

A screen output displayed while the failover and failback scripts run. The display shows only the status of the controller. A user preference for this type of display is set in a resource (.RC) file created by the failover and failback batch files. *See also* verbose display.

configuration file

A file that tells the failover/failback scripts what devices are attached to a controller and how the controller is configured with respect to devices and storagesets. There is one configuration file for each controller pair, so there are two configuration files for a DRM initiator-target pair. *See also* initiator configuration file *and* target configuration file.

container

1. Any entity that is capable of storing data, whether it is a physical device or a group of physical devices. 2. A virtual internal controller structure representing either a single disk or a group of disk drives linked as a storageset. Stripesets and mirrorsets are examples of storageset containers that the controller uses to create units.

controller

A hardware device that uses software to facilitate communications between a host and one or more storage devices organized in an array. The HS-series StorageWorks family of controllers are all array controllers.

control table

A file that controls the order of CLI commands to be issued and sends the appropriate sequence of CLI commands (for the configuration specified in the configuration file) to the Command Scripter. The Command Scripter runs the script file and issues the commands to the HSG80 controller over the Fibre Channel bus.

copying member

In a mirrorset, a copying member is a container introduced to the mirrorset after it has already been in use. None of the blocks can be guaranteed to be the same as in other members of the mirrorset. Therefore, the *copying* member is made the same by copying all the data from a *normal* member. This is in contrast to *normalizing*, where all blocks written since creation are known to be the same.

When all blocks on the copying member are the same as those on the normal member, the copying member becomes a normal member. Until it becomes a normal member, the copying member contains undefined data and is not useful. *See also* normalizing member.

Data Replication Manager

Data Replication Manager provides controller-based mirroring across a Fibre Channel link. The HSG80 Array Controller storage system is used on a host port-to-host port basis, which allows data to be synchronously migrated from one storage system or physical site to another, even if they are located at different physical sites. Using the required multiple-bus failover configuration, the controller is not only able to distribute an input/output request to both the initiator and target sites, but it can also transfer the initiator's role to the target site as needed. Thus, Data Replication Manager is a distributed computing model that supports full disaster-tolerant storage.

device identifier

A unique 16-digit hexadecimal identifier of a Fibre Channel device.

disaster failover

Also referred to as an *unplanned* failover. As applied to the Data Replication Manager, an unplanned outage of the controllers. This may occur when site communication is lost or due to some other failure whereby remote copy sets cannot be implemented. The controllers do not perform an orderly shutdown. *See also* planned failover.

disaster tolerance (DT)

The ability for rapid recovery of user data from a remote location when a significant event (or disaster) occurs at the primary computing site.

disk monitoring

The recording of redundant data for fault-tolerant operation. Data is written on two partitions of the same disk or on two separate disks within the same system. Disk mirroring uses the same controller. RAID 1 provides for mirroring, which is usually accomplished with SCSI drives. *See also* RAID.

disk striping

The spreading of data over multiple disk drives to improve performance. Data is interleaved by bytes or by sectors across the drives. For example, with four drives and a controller designed to overlap reads and writes, four sectors could be read in the same time it normally takes to read one. Disk striping does not inherently provide fault tolerance or error checking. It is used in conjunction with various other methods. *See also* RAID.

DRM controller name

The identification of a controller in a DRM environment.

drmdispatch.pl

See script.

DRM Scripting Kit

A self-extracting kit that contains batch files, Perl scripts, Perl modules, control tables, and example files used to configure a DRM environment and perform failover and failback.

DT

See disaster tolerance.

dual-redundant configuration

A storage subsystem configuration consisting of two active controllers operating as a single controller. If one controller fails, the other controller assumes control of the failing controller's devices. *See also* failover, failback.

environmental variable

Environmental information, such as drive, path, or filename, associated with a symbolic name (in the format %name%) that lets batch files work in the Windows environment. With the DRM scripts, the environmental variable %CLONE_HOME% defines the path to the default directory where the scripts reside.

fabric

A network of switches containing a Fibre Channel arbitrated loop. *See also* switch fabric.

failback

In a DRM environment, after failover occurs, failback moves data operations back to the initiator after the initiator site has been brought back online. *See also* failover.

failover

The process that takes place when one controller in a dual-redundant configuration assumes the workload of a failed companion controller. Failover continues until the failed controller is repaired or replaced. *See also* failback, dual-redundant configuration, planned failover.

failsafe locked

The failsafe error mode can be enabled by the user to fail any I/O whenever the target is inaccessible or the initiator unit fails. When either of these conditions occurs, the remote copy set goes into the inoperative (offline) state and the failsafe error mode is "failsafe locked."

fast-failback

The synchronization of the initiator site with the target site during a planned failover of the initiator subsystem.

The write operations are logged to the target site write history log, and during the fast-failback, the initiator site is updated from the write history log. *See also* mini-merge, disaster failover, planned failover, write history logging.

fiber

An optical strand used in fiber optic cable. Spelled *Fibre* when used in *Fibre Channel* protocol. *See also* fiber optic cable, Fibre Channel bus.

fiber optic cable

A transmission medium that transmits digital signals in the form of pulses of light. Fiber optic cable is noted for its properties of electrical isolation and resistance to electrostatic contamination.

Fibre Channel bus

A high-speed transmission technology that can be used as a front-end communications network, a back-end storage network, or both at the same time. Fibre Channel is a driving force in the storage area network (SAN) arena for connecting multiple hosts to dedicated storage systems. With Fibre Channel, the hosts can talk not only to the storage system via SCSI, but also to each other via IP over the same network. Fibre Channel supports existing peripheral interfaces and communications protocols, including SCSI and IP. Its name is somewhat misleading, as Fibre Channel not only supports single-mode and multi-mode fiber connections, but coaxial cable, and twisted pair as well.

HSG80 Array Controller

An intelligent mass storage controller that interfaces between host computer systems using a Fibre Channel bus and Ultra Wide attached mass storage devices, using Ultra Wide Single Ended SCSI buses.

hsgcontrol.pl

See script.

initiator

The site that carries out primary data processing. If a significant failure occurs at the initiator site, data processing can be resumed at the target site, where the data is intact. *See also* target.

initiator configuration file

A configuration file at the initiator site. *See also* configuration file, initiator.

IP address

An acronym for Internet Protocol address. The IP address is a number that is used as the address specifying a particular computer or other device connected to the Internet.

local terminal

A terminal plugged into the EIA-423 maintenance port on the front bezel of the HS series array controllers. Also called a maintenance terminal.

Logical Unit Number (LUN)

A value that identifies a specific logical unit belonging to a SCSI target ID number. A number associated with a physical device during a task's I/O operations. Each task in the system must establish its own correspondence between logical unit numbers and physical devices.

LOG_UNIT

A CLI command switch that, when enabled, assigns a single, dedicated log unit for a particular association set. The association set members must all be in the normal error mode (not failsafe). *See also* write history logging.

LUN

See Logical Unit Number.

maintenance terminal

See local terminal.

mini-merge

As applied to the Data Replication Manager, the data transfers to be made whenever a target becomes inaccessible. This happens when both links or both target controllers have gone down. The transfers that would have been made are instead logged into the association set's assigned log unit to wait until the remote copy set subsystem comes back online. *See also* fast-failback, write history logging.

mirroring

Duplicating data onto another computer at another location. Mirroring is performed for backup purposes or to be in closer proximity to the user.

mirrorset

1. A group of storage devices organized as duplicate copies of each other. Mirrorsets provide the highest level of data availability at the highest cost. Another name for *RAID 1*. Also called *mirrored units* or *mirrored virtual disks*. 2. Two or more physical disks configured to present one highly reliable virtual unit to the host. 3. A virtual disk drive consisting of multiple physical disk drives, each of which contains a complete and independent copy of the entire virtual disk's data.

multiple intersite links

Each intersite link (ISL) is a fiber link between two switches. As applied to Data Replication Manager, increasing bandwidth between switches is handled by adding additional connections between the switches, with a maximum of two connections.

normalization

A state in which, block for block, data written by the host to a mirrorset member is consistent with the data on other normal and normalizing members. The normalizing state exists only after a mirrorset is initialized.

normalizing member

A mirrorset member whose contents are the same as all other normal and normalizing members for data that has been written since the mirrorset was created, or since lost cache data was cleared. A normalizing member is created by a normal member when either all of the normal members fail, or all of the normal members are removed from the mirrorset. *See also* copying member.

normal member

A mirrorset member that, block for block, contains exactly the same data as that on the other members within the mirrorset. Read requests from the host are always satisfied by normal members.

other controller

The controller in a dual-redundant pair that is not connected to the controller serving the current CLI session with a local terminal. *See also* this controller, local terminal.

peripheral device

Any unit, distinct from the CPU and physical memory, that can provide the system with input or can accept output from it. Terminals, printers, tape drives, and disks are peripheral devices.

Perl

Practical Extraction Report Language. A programming language that combines syntax from several UNIX utilities and languages. Perl is widely used to write Web server programs for such tasks as automatically updating user accounts and news group postings, processing removal requests, synchronizing databases, and generating reports.

Perl interpreter

A program through which Perl programs are passed at run time for execution. The Perl interpreter translates the program internally and then executes it immediately.

Perl module

Code written in the Perl language to perform a specific task.

planned failover

As applied to the Data Replication Manager, an orderly shutdown of the controllers for installation of new hardware, updating the software, and so on. The host applications are quiesced and all write operations are permitted to complete before the shutdown. The controllers must be in synchronous operation mode before starting a planned failover. *See also* disaster failover, synchronous mode.

port

In general terms, a port is:

- A logical channel in a communications system.
- The hardware and software that connect a host controller to a communications bus, such as a SCSI bus or serial bus.

With respect to a controller, a port is the logical route for data in and out of a controller that can contain one or more channels, all of which contain the same type of data.

With respect to a SCSI system, a port is the hardware and software that connect a controller to a SCSI device.

RAID

Redundant Array of Independent Disks. A disk subsystem that increases performance and provides fault tolerance. RAID is a set of two or more hard disks and a specialized disk controller that contains the RAID functionality.

RAID improves performance by disk striping, which interleaves bytes or groups of bytes across multiple drives, so more than one disk is reading and writing simultaneously. Fault tolerance is achieved by mirroring or parity.

RAID 0

Provides disk striping only, which interleaves data across multiple disks for better performance. It does not provide safeguards against failure.

RAID 1

Uses disk mirroring, which provides 100% duplication of data. Offers highest reliability, but doubles storage cost.

RAID 2

Bits (rather than bytes or groups of bytes) are interleaved across multiple disks.

RAID 3

Data is striped across three or more drives. Used to achieve the highest data transfer rate, because all drives operate in parallel. Parity bits are stored on separate, dedicated drives.

RAID 4

Similar to RAID 3, but manages disks independently rather than in unison. Not often used.

RAID 5

Most widely used. Data is striped across three or more drives for performance; parity bits are used for fault tolerance. The parity bits from two drives are stored on a third drive.

RAID 6

Highest reliability, but not widely used. Similar to RAID 5, but does two different parity computations or the same computation on overlapping subsets of the data.

RAID 10

Also designated *RAID 1,0*. It is a combination of RAID 1 and 0 (mirroring and striping).

RCS

See remote copy set.

redundancy

The provision of multiple interchangeable components to perform a single function to cope with failures and errors. A RAIDset is considered to be redundant when user data is recorded directly to one member, and all of the other members and associated parity also are recorded. If a member is missing from the RAIDset, its data can be regenerated as needed, but the RAIDset is no longer redundant until the missing member is replaced and reconstructed.

remote copy set (RCS)

A feature that allows data to be copied (mirrored) from the originating (initiator) site to a remote (target) site. The result is an exact copy of the data (remote copy set) at the target site. Used in disaster tolerance (DT) applications such as the Data Replication Manager. *See also* disaster tolerance (DT).

SAN

See Storage Area Network.

script

A program written in an interpreted programming language that specifies a set of actions to perform a specific task. For DRM scripting, the *hsgcontrol.pl* script reads actions from the application action list and then calls the *drmdispatch.pl* script. The *drmdispatch.pl* script interprets and executes instructions from the failover and failback control tables and initiates CLI commands to accomplish failover and failback. *See also* failover, failback.

scripting language

A high-level programming or command language that is interpreted (translated on the fly) rather than compiled ahead of time. A scripting, or script, language may be a general-purpose programming language or it may be limited to specific functions to augment the running of an application or system program. Spreadsheet macros and communications scripts are examples of limited-purpose scripting languages.

SHOW commands

A set of CLI commands that display information about controllers, storagesets, devices, partitions, and units.

Storage Area Network (SAN)

A back-end network connecting storage devices via peripheral channels such as SCSI and Fibre Channel. There are two ways of implementing SANs: centralized and decentralized. A centralized SAN ties multiple hosts into a single storage system, which is a RAID device with large amounts of cache and redundant power supplies. The cabling distances allow for local as well as campus-wide and metropolitan-wide hookups over peripheral channels, rather than over an overburdened network. SCSI distances have also been extended. This centralized storage topology is commonly employed to tie a server cluster together for failover.

Fibre Channel is a driving force in the SAN arena because it supports existing peripheral interfaces, as well as network interfaces. Fibre Channel can be configured point to point, in an arbitrated loop (FC-AL), or via a switch. With Fibre Channel, the hosts can talk not only to the storage system via SCSI, but they can also communicate with each other via IP over the same topology. If a centralized storage system is not feasible, a SAN can connect multiple hosts with multiple storage systems.

storage array

An integrated set of storage devices. Storage arrays can be manipulated as one unit, with a single command.

storageset

1. A group of devices configured with RAID techniques to operate as a single container.
2. Any collection of containers, such as stripesets, mirrorsets, striped mirrorsets, JBODs, and RAIDsets.

storage unit

The general term that refers to storagesets, single-disk units, and all other storage devices that are installed in a subsystem and accessed by the host. A storage unit can be any entity that is capable of storing data, whether it is a physical device or a group of physical devices.

StorageWorks Command Console (SWCC)

A graphical user interface (GUI) that provides local and remote management of StorageWorks controllers. It is a tool for monitoring, configuring, and troubleshooting storage subsystems. SWCC issues commands and interprets the responses sent by the controller. The user interface displays the logical and physical layout and status of a selected subsystem in graphical form.

surviving controller

The controller in a dual-redundant configuration pair that serves its companion's devices when the companion controller fails.

SWCC

See StorageWorks Command Console.

switch fabric

1. The internal interconnect architecture, used by a switching device, that redirects the data coming in on one of its ports out to another of its ports. 2. The combination of interconnected switches used throughout a campus or large geographic area, which collectively provide a routing infrastructure.

synchronous mode

A mode of operation of the remote copy set whereby the data is written simultaneously to the cache of the initiator subsystem and the cache of the target subsystem. The I/O completion status is not sent until all members of the remote copy set are updated. *See also* asynchronous mode.

target

The site that is set up for data replication. Data processing occurs at the initiator site and data is replicated or copied to the target site. If a significant failure occurs at the initiator site, data processing can be resumed at the target site, where the data is intact. *See also* initiator.

target configuration file

A configuration file at the target site. *See also* configuration file, target.

TCP/IP

Transmission Control Protocol/Internet Protocol. A communications protocol that is the protocol of the Internet and has become the global standard for communications. TCP provides transport functions, which ensure that all bytes sent are received correctly at the other end.

TCP/IP is a routable protocol, and the IP part of TCP/IP provides the routing capability. In a routable protocol, all messages contain not only the address of the destination station, but the address of a destination network. This allows TCP/IP messages to be sent to multiple networks within an organization or around the world, hence its use in the worldwide Internet. Every client and server in a TCP/IP network requires an IP address. *See also* IP address.

this controller

The controller that is serving the current CLI session through a local or remote terminal. *See also* other controller.

unit

A container made accessible to a host. A unit may be created from a single disk drive or tape drive. A unit may also be created from a more complex container, such as a RAIDset. The controller supports a maximum of eight units on each target.

unplanned failover

Also referred to as a *disaster* failover. As applied to the Data Replication Manager, an unplanned outage of the controllers. This may occur when the site communication is lost, or due to some other failure whereby remote copy sets cannot be implemented. The controllers do not perform an orderly shutdown. *See also* planned failover.

verbose display

A screen output displayed while the failover and failback scripts run. The display shows the status of the controller and remote copy sets. A user preference for this type of display is set in a resource (.RC) file created by the failover and failback batch files. *See also* condensed display.

write history logging

As applied to the Data Replication Manager, the use of a log unit to record a history of write commands and data from the host. Write history logging is used for mini-merge and fast- failback. *See also* mini-merge, fast-failback.

Index