Netra High Availability Suite Foundation Services 2.1 6/03 Cluster Administration Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 817–1768–11
September 2004

Adobe PostScript

040902@9495

Contents

Tables

Preface

This book describes how to maintain a cluster running the Netra™ High Availability (HA) Suite Foundation Services 2.1 6/03.

Who Should Use This Book

This book is for system administrators who are maintaining a cluster running the Foundation Services. This book helps administrators to do the following maintenance tasks:

- Map out a cluster network configuration
- Verify the configuration of a cluster or a node
- View and change a disk partition configuration
- Patch cluster software
- Add and replace cluster hardware

Before You Read This Book

To maintain a cluster that is running the Foundation Services you must have experience of working with a cluster environment. Knowledge of networking and system administration of the Solaris™ operating system is also required.

How This Book Is Organized

This book is divided into parts that contain related chapters.

Part I introduces the suite of tools and configuration files you use to maintain and debug the cluster environment:

- Chapter 1 describes how to use the Foundation Services tools and configuration files. It also describes the Solaris tools and configuration files that you can use.
- Chapter 2 explains how to configure and maintain your system log files.

Part II contains procedures that verify whether a cluster is working correctly:

- Chapter 3 describes how to verify whether a group of nodes form a highly available cluster. This chapter also explains how to test whether a cluster is working correctly.
- Chapter 4 shows how to examine the cluster network configuration and the external network configuration.

Part III contains procedures to use to maintain a cluster:

- Chapter 5 explains how to manage differences between files that are not shared. In addition, this chapter describes how to configure the naming services.
- Chapter 6 shows how to start a node without the Foundation Services. It also explains how to stop and restart a node or cluster.
- Chapter 7 describes how to install a patch for the Foundation Services or for the Solaris operating system.
- Chapter 8 describes how to resize a disk partition, how to add a new disk partition, and how to share a disk partition.

Part IV describes how to add nodes to a cluster and how to replace cluster hardware:

- Chapter 9 describes the options for adding a diskless node or dataless node to a cluster.
- Chapter 12 explains how to replace an Ethernet card on a peer node.
- Chapter 11 describes how to replace a board on a peer node.
- Chapter 13 shows how to replace a disk on a dataless node or the vice-master node.

Related Books

You will require some of the following books from the Foundation Services documentation set:

- *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Glossary*
- *What's New in Netra High Availability Suite Foundation Services 2.1 6/03*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Quick Start Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Hardware Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Cluster Administration Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 CMM Programming Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 NMA Programming Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Reference Manual*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Standalone CGTP Guide*
- *Netra High Availability Suite Foundation Services 2.1 6/03 Release Notes*
- *Netra High Availability Suite Foundation Services 2.1 6/03 README*

Accessing Sun Documentation Online

The docs.sun.com^{SM} Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is `http://docs.sun.com`.

Ordering Sun Documentation

Sun Microsystems offers select product documentation in print form. For a list of documents and how to order them, see "Buy printed documentation" at `http://docs.sun.com`.

Typographic Conventions

The following table describes the typographic changes that are used in this book.

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`

`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`

AaBbCc123	Command-line placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. Do *not* save the file. (Emphasis sometimes appears in bold online.)

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`

TABLE P–2 Shell Prompts *(Continued)*

Shell	Prompt
Bourne shell and Korn shell superuser prompt	#

PART I

Using Cluster Administration Tools and Files

This part introduces the tools and configuration files that you can use on a cluster. It shows you how to configure system log files, as described in the following chapters:

- Chapter 1 describes how to use the Foundation Services tools and configuration files, as well as related Solaris tools and configuration files.
- Chapter 2 shows how to configure and maintain system log files.

CHAPTER 1

Using Cluster Administration Tools and Configuration Files

The Foundation Services provide tools and configuration files to administer your cluster environment. Throughout this document, references to a cluster imply the use of a cluster running the Foundation Services. For a description of the cluster environment, see "Cluster Model" in the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

This chapter contains the following sections:

Adding Tools to Your Path

The Foundation Services tools are located in `/opt/SUNWcgha/sbin`. To include these directories in your path, add them to your `/etc/profile` file.

To use some of the Foundation Services tools, you must log in as superuser. To know whether you need to log in as superuser, see the man page of the Foundation Services tool.

To add the man pages to your path, see "To Access Man Pages on the Master-Eligible Nodes" in the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

▼ To Add the Foundation Services Tools to Your Path

1. **Log in to a peer node as superuser.**
2. **Open the `/etc/profile` file in a text editor.**
3. **Add the following lines to the file:**

   ```
   PATH=/opt/SUNWcgha/sbin:$PATH
   export PATH
   ```

4. **Save the `/etc/profile` file, and exit the text editor and the shell session.**
5. **Test the success of this procedure by running a Foundation Services tool at the prompt in a new shell session.**

   For a list of the tools and a reference to their corresponding man pages, see Table 1–1.

Using Foundation Services Tools for Cluster Administration

The following table lists the Foundation Services tools you can use for cluster administration. The table describes the purpose of the tools and provides links to their man pages. To add the man pages to your path, see "To Access Man Pages on the Master-Eligible Nodes" in the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

TABLE 1–1 Foundation Services Cluster Administration Tools

Tool	Description
`nhadm`(1M)	Verifies cluster configuration, using the following commands: ■ `nhadm check` ■ `nhadm check installation` ■ `nhadm check configuration` ■ `nhadm check starting` Creates the file required for patching shared packages, using the following command: ■ `nhadm confshare` Displays information about node configuration, using the following command: ■ `nhadm display` Copies files fom the master node to the vice-master node. Files listed can be passed as an argument or listed in the data-file file. ■ `nhadm [-d` *data-file*`] copy [`*file*`]` Lists differences between specified nonreplicated files on the master node and vice-master node, using the following command: ■ `nhadm synccheck` Stores differences between specified nonreplicated files on the master node and vice-master node, using the following command: ■ `nhadm syncgen` Lists all possible error messages returned by the `nhadm` command: ■ `nhadm -z`
`nhcmmqualif`(1M)	Qualifies a node as the master node if the node is master-eligible, and if the cluster has no master node.
`nhcmmrole`(1M)	Gets the role of the node. A node can have the following roles: ■ MASTER ■ VICEMASTER ■ IN_CLUSTER ■ OUT_OF_CLUSTER For more information about roles, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Glossary*.

TABLE 1–1 Foundation Services Cluster Administration Tools *(Continued)*

Tool	Description
`nhcmmstat`(1M)	Analyzes the state of a node, using the following commands: ■ `info` ■ `local` ■ `master` ■ `mynode` ■ `potential` ■ `vice` Analyzes the state of a group of nodes, using the following commands: ■ `all` ■ `count` Forces a reload of the `cluster_nodes_table` configuration, using the following command: ■ `reload` Triggers a switchover, using the following command: ■ `so` Qualifies a node, using the following command: ■ `squalif` Exits, using the following commands: ■ `exit` ■ `quit` Displays help information, using the following command: ■ `help`
`nhcrfsadm`(1M)	Authorizes or refuses permission to start replication of the master node to the vice-master node.
`nhenablesync`(1M)	Triggers disk synchronization.
`nhpmdadm`(1M)	Administrates the Daemon Monitor. For more information about the Daemon Monitor, see the `nhpmd`(1M) man page.

Using Solaris Tools for Cluster Administration

The following table lists Solaris tools that you can use for cluster administration. The table describes the purpose of the tools and provides links to their man pages. To add the man pages to your path, see “To Access Man Pages on the Master-Eligible Nodes” in the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

TABLE 1–2 Solaris Tools Used for Cluster Administration

Tool	Description
`boot`(1M)	Boots a specific node. For an example of using the `boot` command, see "To Restart a Cluster" on page 74.
`format`(1M)	Formats, labels, repairs, and analyzes disks on your system. For more information about using the `format` utility on a cluster, see "Using the `format` Utility to Display and Modify the Configuration of a Disk Partition" on page 87.
`ifconfig`(1M)	Checks and sets a network interface configuration, as described in "To Obtain Configuration Information About the Network Interfaces of a Node" on page 48.
`init`(1M)	Creates processes from information stored in the `/etc/inittab` file. This command can be used to reboot a node. For example, see "To Restart the Foundation Services" on page 68.
`mount`(1M)	Mounts a file system. For an example of using the `mount` command, see "To Increase the Size of a Mirrored Data Partition on a Virtual Disk" on page 93.
`netstat`(1M)	Shows the network status as described in "Examining the Routes on a Node" on page 55.
`patchadd`(1M)	Adds a patch, as described in Chapter 7.
`pgrep`(1)	Finds processes by name and other attributes. For an example of using the `pgrep` command, see "To Verify That an `nhcmmd` Daemon Is Running on Each Peer Node" on page 41.
`pkill`(1M)	Kills processes. For an example of using the `pkill` command, see "To Replace Ethernet Cards on a Diskless Node With the DHCP Static Boot Policy" on page 108.
`ps`(1)	Lists processes. For a list of monitored daemons, see the `nhpmd`(1M) man page.
`reboot`(1M)	Do not use this command. The `reboot` command does not take into account the order in which daemons are to be stopped or started. Incorrect use of the `reboot` command can cause data loss on the master node and error messages on the vice-master node. To reboot a master-eligible node running the Foundation Services, use the `init` command.
`route`(1M)	Adds or removes a route, as described in "Examining the Routes on a Node" on page 55.
`truss`(1)	Traces system calls and signals. For information about using this command, see "Stopping and Restarting Daemon Monitoring" on page 69.

TABLE 1–2 Solaris Tools Used for Cluster Administration *(Continued)*

Tool	Description
`ufsdump`(1M)	Backs up an entire file system or selected files within a file system. For an example of using this command, see "To Increase the Size of a Mirrored Data Partition on a Virtual Disk" on page 93.
`ufsrestore`(1M)	Restores files from backup media created with the `ufsdump` command. For an example of using this command, see "To Increase the Size of a Mirrored Data Partition on a Virtual Disk" on page 93.

Debugging Diskless Nodes With the `snoop` Tool

When debugging diskless nodes, use the `snoop` command, as follows.

▼ To Examine a Diskless Node From the Master Node Using `snoop`

1. **Log in to the diskless node that you want to examine.**
2. **Find the IP address of this node:**

   ```
   % ifconfig -a
   ```

   The `ifconfig` command returns the IP address, the interface type, and the Ethernet address.
3. **Log in to the master node.**
4. **From a console window, run:**

   ```
   % snoop -d NIC0 ether diskless-node-NIC0-Ethernet-address
   ```

   The `snoop` command captures packets from the network and displays their contents.
5. **From another console window, run:**

   ```
   % snoop -d NIC1 ether diskless-node-NIC1-Ethernet-address
   ```

6. **To exit the `snoop` tool, press Control-C.**

Using Foundation Services Configuration Files in the Cluster Environment

The files created during initial cluster configuration can be used during administration to determine the state of the cluster. By default the configuration files are in the `/etc/opt/SUNWcgha/` directory. The following table lists the files created during initial cluster configuration.

TABLE 1–3 Foundation Services Files Created During Initial Cluster Configuration

Configuration File	Description
`cluster_nodes_table`	Contains a list of the peer nodes in a cluster. There is a copy of the file on each master-eligible node. For information, see the `cluster_nodes_table`(4) man page.
`nhfs.conf`	Contains configuration information for the individual Foundation Services. There is a copy of this file on each peer node. For information, see the `nhfs.conf`(4) man page.
`nma.notifs.txt`	Defines types and targets for the trap notifications of the Node Management Agent (NMA) SNMP. For information, see the `nma.notifs.txt`(4) man page.
`nma.params.txt`	Defines the communication, implementation and accessibility configuration of SNMP in the NMA. For information, see the `nma.params.txt`(4) man page.
`nma.properties`	Lists the properties of the NMA on a node. There is a copy of this file on each peer node. For information, see the `nma.properties`(4) man page.
`nma.targets.txt`	Defines the targets to which the NMA sends SNMP traps. For information, see the `nma.targets.txt`(4) man page.

TABLE 1–3 Foundation Services Files Created During Initial Cluster Configuration *(Continued)*

Configuration File	Description
`nma.security`	Configures SNMP security for the NMA. For information, see the `nma.security`(4) man page.

When you reconfigure a cluster, you might need to modify configuration files. Refer to the man page of a configuration file for information about how to change the file.

Using Solaris Configuration Files in the Cluster Environment

This section describes the Solaris configuration files that can help you determine the current configuration of the cluster. For information about how and when to modify a Solaris configuration file, consult the man page for the file and the Solaris documentation set. The following table lists the Solaris configuration files that can be edited during cluster configuration.

TABLE 1–4 Solaris Files Used With the Foundation Services

Configuration File	Description
`/etc/bootparams`	A database of boot parameters. For information, see `bootparams`(4).
`/etc/coreadm.conf`	Parameters for system core file configuration.
`/etc/dfs/dfstab`	Commands for sharing resources across a cluster. Commands for partitions mirrored by Reliable NFS are not located in this file. For information, see `dfstab`(4).
`/etc/hostname.`*interface-name*	Configuration of each network interface on a node. The file contains the host name of the interface specified by *interface-name*. If you create a network interface, you must create a `/etc/hostname.`*interface-name* file for that interface. For information about network interface configuration, see "Examining the Network Configuration Files" on page 54.
`/etc/hosts`	A local database of host names. For information, see the `hosts`(4) man page. For information about network interface configuration files, see "Examining the Network Configuration Files" on page 54.

TABLE 1–4 Solaris Files Used With the Foundation Services *(Continued)*

Configuration File	Description
`/etc/inet/inetd.conf`	The list of servers that `inetd` invokes when it receives an internet request over a socket. For information, see the `inetd.conf`(4) man page.
`/etc/inittab`	The process dispatching control by the `init` command. This file contains the `rc2` and `rc3` entries. For information, see the `inittab`(4) man page. For information about the Daemon Monitor, see the `nhpmd`(1M) man page.
`/etc/netmasks`	A list of the network masks used to implement IP subnetting and their associated IP network numbers. For information, see the `netmasks`(4) man page. For information about network configuration, see Chapter 4.
`/etc/nodename`	The local source for the system name. For information, see the `nodename`(4) man page. For information about network interface configuration files, see “Examining the Network Configuration Files” on page 54.
`/etc/notrouter`	A file to specify that a node does not act as a router. This file is used to protect against route mismatch. This file must be configured on each peer node to ensure that the node is not routable.
`/etc/nsswitch.conf`	A configuration file for the name service that provides the sources of database information and their lookup order. For information, see the `nsswitch.conf`(4) man page. For information about using the naming services, see “Using the Naming Services” on page 65.
`/etc/rc2`	A script that executes commands when the system is changing to init state 2. A modified version of this script is used by the Daemon Monitor. For information about the Daemon Monitor, see the `nhpmd`(1M) man page.
`/etc/rc3`	A script that executes commands when the system is changing to init state 3. A modified version of this script is used by the Daemon Monitor. For information about the Daemon Monitor, see the `nhpmd`(1M) man page.
`/etc/services`	A list of the services and aliases available through the Internet. The `/etc/services` file can be changed to modify the Foundation Services ports if, for example, an application is already using the ports configured by default at startup. For information, see the `services`(4) man page. For information about prioritizing the Foundation Services, see “Using the Naming Services” on page 65.

TABLE 1–4 Solaris Files Used With the Foundation Services *(Continued)*

Configuration File	Description
`/etc/syslog.conf`	A file used by the system log daemon, `syslogd`, to forward a system message to the appropriate log files and users. For information, see the `syslog.conf`(4) man page. For information about configuring system log files, see Chapter 2.
`/etc/system`	A file that customizes the operation of the operating system kernel. You should not have to change the `/etc/system` file. If you do change the `/etc/system` file, you must reboot the node for the changes to take effect.
`/etc/vfstab`	A table of file system defaults. For information, see `vfstab`(4).
`.profile`	A file to set up an environment for a user at login time. For information about how to use this file, see the `profile`(4) man page. For examples of the configuration of the `.profile` file, see "To Add the Foundation Services Tools to Your Path" on page 20 and "To Set `NLSPATH` Permanently" on page 32.
`.rhosts`	A list of trusted hosts and users. The `.rhosts` file specifies remote trusted hosts and users. Modify this file to change remote access authorization between peer nodes. For information, see `rhosts`(4).

CHAPTER 2

Accessing and Maintaining Log Files

Information, error, and alert messages are logged to system log files. Use the system log files to diagnose problems or errors. By configuring the log files, you can direct different categories of messages to files or to the console window. In this way, you can make the information in logs more accessible and prevent the occurrence of large log files.

For information about using log files generated by the Solaris operating system and the Foundation Services, consult the `syslog`(3) and `syslogd`(1M) man pages.

This chapter contains the following sections:

Accessing System Log Files

All information, error, and alert messages generated by the Foundation Services are sent to the system log files. The messages can be processed by client programs or by the Node Management Agent (NMA). To configure your access to the system log files, edit the`/etc/syslog.conf` file. You can add or change the message sources, priorities, and message locations according to the syntax described in the `syslog.conf`(4) man page. See the man pages of the individual daemons for details of the error levels used by each service.

All Foundation Services messages have the `syslog` facility set to `local0`. You can configure system log files to contain selected categories of messages. The Foundation Services have the following error message categories:

- `emerg`

- `alert`
- `crit`
- `err`
- `info`
- `none`
- `notice`

For information about error message categories, see the `syslog.conf`(4) man page.

Note – The console window on the Netra CP21*xx* board is a 9600–bps serial line. If you are using a Netra CP21*xx* board, configure the `syslog.conf` file to direct messages to files.

▼ To Redirect Foundation Services Messages to a File on the Master Node

This procedure redirects specific categories of messages to log files.

1. **Log in as superuser to the node for which you want to redirect messages.**
2. **Open the `/etc/syslog.conf` file in a text editor.**
3. **To redirect the `info` and `notice` messages to a log file called *logfile*, add the following line to the `/etc/syslog.conf` file:**

   ```
   local0.info;local0.notice / ifdef /var/adm/logfile
   ```

   Note – Use tabulation in the `/etc/syslog.conf` file.

   ---

4. **Save the `/etc/syslog.conf` file and exit the text editor.**
5. **Create a `/var/adm/`*logfile* file if this file does not already exist:**

   ```
   # touch /var/adm/logfile
   ```

6. **Stop the `syslogd` daemon:**

   ```
   # /etc/init.d/syslog stop
   ```

7. **Restart the `syslogd` daemon:**

   ```
   # /etc/init.d/syslog start
   ```

▼ To Define `loghost` in `/etc/hosts`

This procedure configures messages to be sent to a specific node. This procedure can be used to prevent disruption of Network File System (NFS) traffic by high volumes of messages.

1. **Log in to the master node as superuser.**
2. **In a text editor, open the `/etc/hosts` file for the diskless node.**

   If your cluster was created by the `nhinstall` tool, the `/etc/hosts` file is located at `/export/root/`*diskless-node-name*`/etc/hosts`.
3. **Remove the `loghost` state for the `localhost` entry.**
4. **Enter the `loghost` state for the `cgtp0` address of the master node.**

   For example:

   ```
   10.x.3.y    master        loghost
   ```

   If the master node is using a default class C address, `10.`*x*`.3.`*y* is the IP address of the `cgtp0` interface. The system log messages are directed to the *logfile* specified for the master node.
5. **Save the `/etc/hosts` file and exit the text editor.**
6. **Stop the `syslogd` daemon:**

   ```
   # /etc/init.d/syslog stop
   ```
7. **Restart the `syslogd` daemon:**

   ```
   # /etc/init.d/syslog start
   ```

System log messages are sent to the node specified in the `/etc/hosts` file. The node can be the current node or any other peer node. On the specified node, the messages are logged to a file called *logfile*, specified in the `/etc/syslog.conf` file. For information about redirecting messages to a nonpeer node, see "To Redirect System Log Messages to a Nonpeer Node" on page 33.

Accessing Internationalized Message Catalogs

Messages generated by the Foundation Services are stored in message catalogs. You can set the access permissions to the internationalized catalogs in one of the following ways:

- Set the access permissions for an individual shell session, as described in "To Set `NLSPATH` for an Individual Shell Session" on page 32.

- Set the access permissions permanently, as described in "To Set `NLSPATH` Permanently" on page 32.

▼ To Set `NLSPATH` for an Individual Shell Session

1. **Log in to a peer node.**
2. **Set the language:**

   ```
   # LANG=C
   ```

   The default language is English.
3. **Export the language:**

   ```
   # export LANG
   ```
4. **Set the National Language Support (NLS) path:**

   ```
   # NLSPATH=/opt/SUNWcgha/lib/locale/%L/LC_MESSAGES/%N
   ```
5. **Export the path:**

   ```
   # export NLSPATH
   ```

▼ To Set `NLSPATH` Permanently

1. **Log in to the peer node on which you want to set up `NLSPATH`.**
2. **Open the `/etc/profile` file in a text editor.**
3. **Add the following lines to the `/etc/profile` file:**

   ```
   LANG=C
   export LANG
   NLSPATH=/opt/SUNWcgha/lib/locale/%L/LC_MESSAGES/%N
   export NLSPATH
   ```
4. **Save the `/etc/profile` file and exit the editor.**
5. **Source the `/etc/profile` file.**

Maintaining System Log Files

This section describes how to manage large system log files and how to remove log files. Use tools such as `grep` to search the system log file and identify messages for a specific node.

▼ To Redirect System Log Messages to a Nonpeer Node

This procedure redirects system log messages from peer nodes to the system log file on a nonpeer node, preventing the need to maintain large log files on peer nodes.

1. **Log in to a peer node as superuser.**

2. **In a text editor, add the name of the nonpeer node to the `/etc/hosts` file.**

3. **In a text editor, edit the `/etc/syslog.conf` file as follows:**

   a. **Remove the line:**

   ```
   local0.info;local0.notice;local0.crit;local0.warning    /var/adm/logfile
   ```

   b. **Add the line:**

   ```
   local0.info;local0.notice;local0.crit;local0.warning     @nonpeer
   ```

   nonpeer is the name of the nonpeer node, as defined in the `/etc/hosts` file.

   You can replace @*nonpeer* with `@loghost` if `loghost` is defined in `/etc/hosts`.

4. **Repeat Step 1 through Step 3 on each peer node.**

5. **Log in to the nonpeer node to which you want to redirect the messages.**

6. **Create an empty file called `/var/adm/`*logfile*:**

   ```
   # touch /var/adm/logfile
   ```

7. **Open the `/etc/syslog.conf` file in a text editor.**

8. **Specify *logfile* as the destination for system log messages by adding the following line to the `/etc/syslog.conf` file:**

   ```
   local0.info;local0.notice;local0.crit;local0.warning        /var/adm/logfile
   ```

9. **Stop the `syslogd` daemon:**

   ```
   # /etc/init.d/syslog stop
   ```

10. **Restart the `syslogd` daemon:**

```
# /etc/init.d/syslog start
```

▼ To Remove Links to Crashed Daemons

Removing old log files might not be sufficient to free space. When a daemon crashes while writing to a log file, a link to the log file still exists after the death of the daemon. Consider the impact on a cluster before shutting down a daemon.

1. **Stop the daemon.**
2. **Remove the link to the log file.**
3. **Restart the daemon.**

Logging NFS Operations

To log NFS operations, perform the following procedure.

▼ To Log NFS Operations

1. **Log in to the master node as superuser.**
2. **Open the `/etc/nfs/nfslog.conf` file in a text editor and add this line:**

```
my-tag default_dir=my-dir logformat=extended
```

my-tag	Label to identify NFS log messages
my-dir	Directory where log of the NFS operations is located

3. **Open the `/etc/dfs/dfstab` file in a text editor and add this line:**

```
share -o log=my-tag shared-nfs
```

my-tag	Label to identify NFS log messages
shared-nfs	Shared directory to which all NFS operations, such as `rm` and `mkdir`, are logged.

4. **Restart the NFS server:**

```
# /etc/init.d/nfs.server start
```

5. **Verify that the `shared-nfs` directory is being shared:**

```
# share
```

An output similar to this is displayed in the console window:

```
-               /share-nfs   log=my-tag   ""
```

6. **Verify that the `nfslogd` daemon is running:**

   ```
   # ps -ef | grep nfslogd
   ```

   - If the `nfslogd` daemon is running, output similar to this is produced:

     ```
     root    606   348   0 15:03:37 pts/2     0:00 grep nfslogd
     root    592     1   0 14:51:09 ?         0:00 /usr/lib/nfs/nfslogd
     ```

   - If the daemon is not running, repeat Step 4 and Step 5.

7. **Test that NFS operations are being logged.**

   a. **Log in to a peer node other than the master node, as superuser.**

      This node is referred to as *node A*.

   b. **Mount the *share-nfs* directory:**

      ```
      # mount master-node:/share-nfs /mnt
      ```

   c. **Log in to the master node as superuser.**

   d. **Create a file:**

      ```
      # touch myfile
      ```

      ```
      # chmod 777 myfile
      ```

   e. **Log in to *node A* and examine *myfile*:**

      ```
      # echo TEST myfile
      ```

   f. **Examine the log files on the master node:**

      ```
      # cat  /share-NFS/nfslog
      ```

      You should see entries for the NFS operations that you have performed.

8. **Log in to the vice-master node as superuser.**

9. **Repeat Step 2 through Step 7.**

PART II

Verifying the Cluster Configuration

To verify that a cluster is working and to investigate and map the cluster network configuration, see the following chapters:

- Chapter 3 describes how to verify whether a cluster is running highly available services. This chapter also describes how to test whether a cluster is functioning correctly.
- Chapter 4 shows how to obtain information about a cluster network configuration.

CHAPTER **3**

Determining Cluster Validity

This chapter describes how to verify whether a group of nodes form a cluster, and whether the cluster is functioning correctly. Before you perform maintenance tasks or change the cluster configuration, verify that the cluster is functioning correctly. When you have completed maintenance tasks, verify that the cluster is still functioning correctly.

This chapter is divided into the following sections:

Defining Minimum Criteria for a Cluster Running Highly Available Services

A Foundation Services cluster can run the following highly available services: Reliable NFS and the Reliable Boot Service. For information about highly available services, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

A highly available cluster has the following features:

- A master node and a vice-master node. The master node is the central information point for the cluster. The vice-master node backs up the master node. To verify that there is a master node and a vice-master node in the cluster, see "To Verify That the Cluster Has a Master Node and a Vice-Master Node" on page 40.
- An `nhcmmd` daemon on each peer node. The `nhcmmd` daemon on the master node manages the membership of the other peer nodes. The `nhcmmd` daemon on other peer nodes receives cluster information from the `nhcmmd` daemon on the master

node. To verify that there is an `nhcmmd` daemon on each peer node, perform the procedure described in "To Verify That an `nhcmmd` Daemon Is Running on Each Peer Node" on page 41.

- A redundant network. When the network is redundant, there is no single point of network failure. To verify that the cluster network is redundant, see "To Verify That the Cluster Has a Redundant Ethernet Network" on page 41.
- Synchronized master node disk and vice-master node disk. Synchronization ensures that the vice-master node has an up-to-date copy of the information on the master node. To verify that the master node and vice-master node are synchronized, see "To Verify That the Master Node and Vice-Master Node Are Synchronized" on page 42.

If your cluster has diskless nodes, the Reliable Boot Service must be running on the master node and the vice-master node.

Verifying Services on Peer Nodes

When performing administration tasks, regularly verify that your cluster is running correctly by performing the procedures described in this section.

▼ To Verify That the Cluster Has a Master Node and a Vice-Master Node

1. **Log in to a master-eligible node as superuser.**
2. **Type:**

```
# nhcmmstat -c all
```

The `nhcmmstat` command displays information in the console window about all of the peer nodes. The information includes the role of each node. The peer nodes must include a master node and a vice-master node. For more information, see the `nhcmmstat`(1M) man page.

- If there is a master node but no vice-master node, reboot the second master-eligible node:

  ```
  # init 6
  ```

  Verify that the second master-eligible node has become the vice-master node:

  ```
  # nhcmmstat -c all
  ```

  If the second master-eligible node does not become the vice-master node, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

- If there is neither a master node nor a vice-master node, you do not have a highly available cluster. Verify your cluster configuration by examining the `nhfs.conf` file and the `cluster_nodes_table` file for configuration errors.

  For more information, see the `nhfs.conf`(4) and `cluster_nodes_table`(4) man pages.
- If there are two master nodes, you have a *split brain* error scenario. To investigate the cause of split brain, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

▼ To Verify That an `nhcmmd` Daemon Is Running on Each Peer Node

1. **Log in to a peer node.**
2. **Verify that an `nhcmmd` daemon is running on the node:**

   ```
   # pgrep -x nhcmmd
   ```

   - If a process identifier is returned, the daemon is running.
   - If a process identifier is not returned, the daemon is not running.

   To investigate the cause of daemon failure, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.
3. **Repeat Step 1 and Step 2 on each peer node.**

▼ To Verify That the Cluster Has a Redundant Ethernet Network

1. **Log in to a peer node as superuser.**
2. **Verify that the peer nodes are communicating through a network:**

   ```
   # nhadm check starting
   ```

   If any peer node is not accessible from any other peer node, the `nhadm` command displays an error message in the console window.
3. **Search the system log files for this message:**

   ```
   [ifcheck] Interface interface-name used for cgtp has failed
   ```

   This message is created by the `nhcmmd` daemon if the peer nodes are not communicating through a redundant network.

   If the redundant network fails, examine the card, cable, and route table associated with the link. Investigate the system log files for relevant error messages.

▼ To Verify That the Master Node and Vice-Master Node Are Synchronized

1. **Log in to a master node as superuser.**
2. **Test whether the vice-master node is synchronized with the master node:**

```
# /usr/opt/SUNWesm/sbin/scmadm -S -M
```

- If the `scmadm` command reaches the replicating state, the vice-master node is synchronized with the master node.
- If the `scmadm` command does not reach the replicating state, the vice-master node is not synchronized with the master node.

3. **If the master and vice-master nodes are not synchronized, verify if the** *RNFS.EnableSync* **parameter is set in to `FALSE` in the `nhfs.conf` file.**

   If the *RNFS.EnableSync* parameter is set to `FALSE` and if you want to trigger synchronization:

   a. **Trigger synchronization:**

   ```
   # nhenablesync
   ```

   For information on `nhenableysnc`, see `nhenablesync`(1M).

   b. **Repeat Step 2.**

   If the *RNFS.EnableSync* parameter is not set to `FALSE` but the vice-master node remains unsynchronized, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

   For more information about the `scmadm` command, see the `scmadm`(1M) man page. For more information about the *RNFS.EnableSync* parameter, see the `nhfs.conf`(4) man page.

▼ To Verify That the Reliable Boot Service Is Running

1. **Log in to the master node.**
2. **Determine whether an `in.dhcpd` daemon is running on the node:**

```
# pgrep -x in.dhcpd
```

- If a process identifier is returned, the daemon is running.
- If a process identifier is not returned, the daemon is not running.

To investigate the cause of daemon failure, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

Verifying That a Cluster Is Configured Correctly

A cluster must meet the criteria outlined in "Defining Minimum Criteria for a Cluster Running Highly Available Services" on page 39. The following procedures describe how to verify that a cluster is configured correctly.

▼ To Verify That a Cluster Is Configured Correctly

1. **Log in to a peer node as superuser.**
2. **Type:**

   ```
   # nhadm check
   ```

   The `nhadm` tool tests whether the Foundation Services and its prerequisite products are installed and configured correctly.

   If the `nhadm` command encounters an error, it displays a message in the console window. If you receive an error message, perform the following steps:

   a. **Identify the problem area, diagnose, and correct the problem.**

      For an explanation of the error messages displayed by `nhadm`, type:

      ```
      # nhadm -z
      ```

   b. **Rerun the `nhadm check` command, diagnosing and correcting any further errors until all tests pass.**

For more information, see the `nhadm`(1M) man page.

Reacting to a Failover

When a master node fails over to the vice-master node, a fault has occurred. Even though your cluster has recovered, the fault that caused the failover could have serious implications for the future performance of your cluster. You must treat a failover seriously. After a failover, perform the following procedure.

▼ To React to a Failover

1. **Log in to the failed master node as superuser.**

2. **Examine the system log files for information about the cause of the failover.**

   For information about log files, see Chapter 2.

3. **Verify that the failed master node has been elected as the vice-master node:**

   ```
   # nhcmmstat -c vice
   ```

   - If there is a vice-master node in the cluster, `nhcmmstat` prints information to the console window about the vice-master role.
   - If there is no vice-master node, `nhcmmstat` sends an error code.

     If there is no vice-master node, investigate why the failed master node is not capable of taking the vice-master role. For information, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

4. **Ensure that you have a valid cluster as described in "Defining Minimum Criteria for a Cluster Running Highly Available Services" on page 39.**

5. **Run the `nhadm check` command to verify that the node is correctly configured.**

   ```
   # nhadm check
   ```

CHAPTER **4**

Examining the Cluster Networking Configuration

For information about how to obtain information about peer nodes and the network interfaces on those nodes, see the following sections. For a description of addressing and networking, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

Using Tools to Examine the Cluster Network

Use the following tools to examine the cluster network configuration:

- `nhadm`
- `ifconfig`
- `netstat`

Use these tools to ensure that your cluster network is correctly configured after maintenance, or to obtain information about cluster membership problems. For information about solving cluster membership problems, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

Verifying the Network Configuration

Before collecting data about the cluster network configuration, perform the following procedure to test whether the interfaces of a node are configured correctly.

▼ To Verify the Network Interfaces of a Node

1. **Log in as superuser to the node whose interfaces you want to examine.**
2. **Check the network configuration:**

   ```
   # nhadm check configuration
   ```

   The tests this command performs include:

   - Whether the files `/etc/hostname.`*NIC0* and `/etc/hostname.`*NIC1* exist on the current node.
   - Whether the `/etc/hosts` file contains IP addresses that correspond to the *NIC0*, *NIC1*, and `cgtp0` interfaces for each peer node.

   The `nhadm` tool displays the success or failure of each test it performs.

Creating a Network Topology Map

A network topology map contains information about each node in a cluster, and each interface on a node. A network topology map can also include information about nonpeer nodes that are communicating with the cluster.

When investigating your network topology, you can create a network topology map. The map is a helpful reference when using the cluster or changing the cluster configuration. The following figure illustrates an example of the information that you can include in a network topology map.

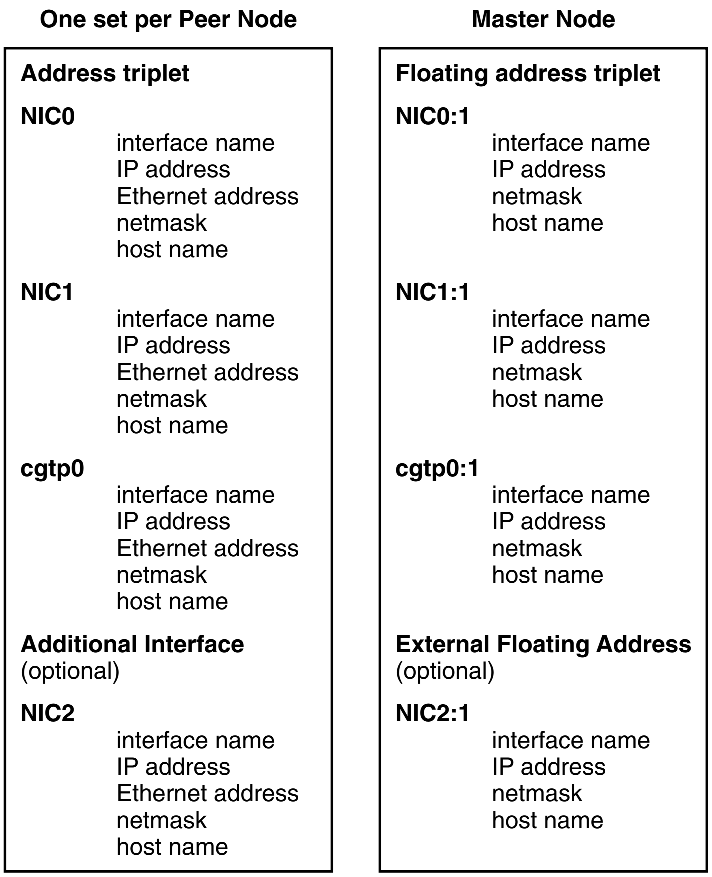

FIGURE 4–1 Template for a Network Topology Map

As you work through the procedures in this chapter, add information to the network topology map for the nodes in your cluster. You can link all the nodes in the network topology map using the routing information described in “Examining the Routes on a Node” on page 55.

Mapping the Network Configuration of a Peer Node

This section describes how to obtain network information about a peer node.

▼ To Identify the Name of a Node

1. **Log in to the node you want to examine.**

2. **Run:**

```
# hostname
```

The name of the node is displayed in the console window.

▼ To Identify the Host Name of the Network Interfaces of a Node

1. **Log in to the node whose network interfaces you want to examine.**

2. **Run:**

```
# netstat -i
```

The host name of the node's network interfaces is displayed.

For more information, see the `netstat`(1M) man page.

▼ To Obtain Configuration Information About the Network Interfaces of a Node

This procedure determines the IP addresses, netmask value, network IDs, node IDs, Ethernet address, and interface names of a peer node.

1. **Log in to the node whose network interfaces you want to examine.**

2. **Run the `ifconfig` command:**

```
# ifconfig -a
```

The `ifconfig` command displays configuration information about the network interfaces.

Each peer node has at least three configured network interfaces. The physical interfaces, *NIC0* and *NIC1*, and the CGTP interface. Additional network interfaces are displayed for the master node, the vice-master node, and for interfaces that are configured for external access.

If the node is using the class B addressing scheme, output similar to this is displayed:

```
hme0: flags=1004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4> mtu 1500 \
index 1
        inet 172.15.0.30 netmask ffff0000 broadcast 172.15.255.255
        ether 8:0:20:f9:b3:60
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 2
```

```
        inet 127.0.0.1 netmask ff000000
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
        inet 172.16.0.30 netmask ffff0000 broadcast 172.16.255.255
        ether 8:0:20:f9:b3:61
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
        inet 172.17.0.30 netmask ffff0000 broadcast 172.17.255.255
        ether 0:0:0:0:0:0
```

In this example, the *nodeid* is `30`, the netmask is `ffff0000`, and the *network ID*s are `172.15.0.0`, `172.16.0.0`, and `172.17.0.0`.

If the node is using the default class C addressing scheme, output similar to this is displayed:

```
hme0: flags=1004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4> mtu 1500 \
index 1
        inet 10.250.1.30 netmask ffffff00 broadcast 10.250.1.255
        ether 8:0:20:f9:b4:b0
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 2
        inet 127.0.0.1 netmask ff000000
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
        inet 10.250.2.30 netmask ffffff00 broadcast 10.250.2.255
        ether 8:0:20:f9:b4:b1
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
        inet 10.250.3.30 netmask ffffff00 broadcast 10.250.3.255
        ether 0:0:0:0:0:0
```

In this example, the *nodeid* is `30`, the netmask is `ffffff00`, and the *network ID*s are `10.250.1.0`, `10.250.2.0`, and `10.250.3.0`.

Mapping the External Network Configuration of a Node

This section describes how to create a map of the external network configuration of a peer node.

For further information about external addresses, see “External Addressing and Networking” in *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

▼ To Examine Interfaces Configured for External Communication

Use this procedure to determine whether a node is configured for external communication.

1. **Log in to the node whose network interfaces you want to examine.**

2. **Run the `ifconfig` command:**

   ```
   # ifconfig -a
   ```

   The `ifconfig` command displays configuration information about the network interfaces of a node. The following output is for a diskless or a dataless node using the class C addressing scheme, and configured for external communication through additional logical interfaces.

   ```
   hme0: flags=1004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4> mtu 1500 \
   index 1
           inet 10.250.1.30 netmask ffffff00 broadcast 10.250.1.255
           ether 8:0:20:f9:b4:b0
   lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 2
           inet 127.0.0.1 netmask ff000000
   hme0:100: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 \
   index 2
           inet 172.36.128.51 netmask ffffff00 broadcast 172.36.128.255
   hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
           inet 10.250.2.30 netmask ffffff00 broadcast 10.250.2.255
           ether 8:0:20:f9:b4:b1
   cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
           inet 10.250.3.30 netmask ffffff00 broadcast 10.250.3.255
           ether 0:0:0:0:0:0
   ```

3. **Search in the output of Step 2 for an interface to an external network.**

   In this example, the logical interface `hme0:100` is an additional interface configured for external addressing.

Mapping the Floating Address Triplet of the Master Node

The master node and vice-master node have three interfaces for the floating address triplet. For example, `NIC0:1`, `NIC1:1`, and `cgtp0:1`. The interfaces are assigned to the master node and vice-master node, but are configured as *up* on the master node only. If a switchover or failover occurs, the floating address triplet is configured *down* on the old master node and *up* on the new master node. For more information, see “Cluster Addressing and Networking” in the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

▼ To Map the Floating Address Triplet of the Master Node

1. **Log in to a peer node.**

2. **Identify the master node:**

```
# nhcmmstat all
```

The `nhcmmstat` command also displays information in the console window about each peer node.

3. **Log in to the master node as superuser.**

4. **Run:**

```
# ifconfig -a
```

For a node using the class C addressing scheme, output similar to this is displayed:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
         inet 127.0.0.1 netmask ff000000
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
         inet 10.98.1.11 netmask ffffff00 broadcast 10.98.1.255
         ether 8:0:20:da:7d:9c
hme0:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
mtu 1500 index 2
         inet 10.98.1.1 netmask ffffff00 broadcast 10.98.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
         inet 10.98.2.11 netmask ffffff00 broadcast 10.98.2.255
         ether 8:0:20:da:7d:9d
hme1:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
mtu 1500 index 3
         inet 10.98.2.1 netmask ffffff00 broadcast 10.98.2.255
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
         inet 10.98.3.11 netmask ffffff00 broadcast 10.98.3.255
         ether 0:0:0:0:0:0
cgtp0:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
mtu 1500 index 4
         inet 10.98.3.1 netmask ffffff00 broadcast 10.98.3.255
```

The floating address triplet has the logical interfaces `hme0:1`, `hme1:1`, and `cgtp0:1`. Note the IP addresses and netmask of the interfaces for the floating address triplet.

5. **Log in to the vice-master node as superuser.**

6. **Repeat Step 4.**

Output similar to this is displayed:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv> mtu 8232 index 1
         inet 127.0.0.1 netmask ff000000
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 \
index 2
         inet 10.98.1.10 netmask ffffff00 broadcast 10.98.1.255
         ether 8:0:20:da:8f:c6
hme0:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
1500 index 2
         inet 10.98.1.1 netmask ffffff00 broadcast 10.98.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
         inet 10.98.2.10 netmask ffffff00 broadcast 10.98.2.255
         ether 8:0:20:da:8f:c7
hme1:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
1500 index 3
        inet 10.98.2.1 netmask ffffff00 broadcast 10.98.2.255
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
        inet 10.98.3.10 netmask ffffff00 broadcast 10.98.3.255
        ether 0:0:0:0:0:0
cgtp0:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
1500 index 4
        inet 10.98.3.1 netmask ffffff00 broadcast 10.98.3.255
```

Note – Note that the `hme0:1`, `hme1:1`, and `cgtp0:1` interfaces of the vice-master are configured but are not marked `UP`.

Mapping the Floating External Address of the Master Node

The Node State Manager (NSM) assigns an external IP address to one interface on the master node. This address is called the *floating external address*. To examine the floating external address, perform the following procedure.

▼ To Map the Floating External Address of the Master Node

1. **Log in to a peer node.**

2. **Identify the master node:**

   ```
   # nhcmmstat all
   ```

   The `nhcmmstat` command also displays information about each peer node to the console window.

3. **Log in to the master node.**

4. **Verify that the NSM has been configured in the `nhfs.conf` file.**

   For a node using the class C addressing scheme, the NSM configuration is like this:

   ```
   NSM.Exec.Masterdir=/opt/SUNWcgha/actions/master
   NSM.Exec.Vicemasterdir=/opt/SUNWcgha/actions/vicemaster
   NSM.Log.Masterdir=/var/adm/log/SUNWcgha/nsm_master
   NSM.Log.Vicemasterdir=/var/adm/log/SUNWcgha/nsm_vicemaster
   NSM.External.Master.Address=172.36.128.5
   NSM.External.Master.Nic=hme0:101
   ```

In this example, a floating external address is configured on the master node. The floating external address is assigned to the logical interface `hme0:101`, and has the external IP address `172.36.128.5`. After failover or switchover, the floating external address is configured on the new master node.

5. **Run the `ifconfig` command on the master node:**

   ```
   # ifconfig -a
   ```

   Output similar to this is displayed:

   ```
   lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
           inet 127.0.0.1 netmask ff000000
   hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
           inet 10.250.1.20 netmask ffffff00 broadcast 10.250.1.255
           ether 8:0:20:f9:b3:6a
   hme0:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
   mtu 1500 index 2
           inet 10.250.1.1 netmask ffffff00 broadcast 10.250.1.255
   hme0:101: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> \
   mtu 1500 index 2
               inet 172.36.128.5 netmask ffffff00 broadcast 172.36.128.255
   hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
           inet 10.250.2.20 netmask ffffff00 broadcast 10.250.2.255
           ether 8:0:20:f9:b3:6b
   hme1:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
   mtu 1500 index 3
           inet 10.250.2.1 netmask ffffff00 broadcast 10.250.2.255
   cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
           inet 10.250.3.20 netmask ffffff00 broadcast 10.250.3.255
           ether 0:0:0:0:0:0
   cgtp0:1: flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> \
   mtu 1500 index 4
           inet 10.250.3.1 netmask ffffff00 broadcast 10.250.3.255
   ```

   This output shows the `hme0:101` interface as configured in the `nhfs.conf` file.

6. **Run the `ifconfig` command on the vice-master node:**

   ```
   # ifconfig -a
   ```

   Output similar to this is displayed:

   ```
   lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv> mtu 8232 index 1
            inet 127.0.0.1 netmask ff000000
   hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
            inet 10.98.1.10 netmask ffffff00 broadcast 10.98.1.255
            ether 8:0:20:da:8f:c6
   hme0:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
   1500 index 2
            inet 10.98.1.1 netmask ffffff00 broadcast 10.98.1.255
   hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
            inet 10.98.2.10 netmask ffffff00 broadcast 10.98.2.255
            ether 8:0:20:da:8f:c7
   hme1:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
   1500 index 3
            inet 10.98.2.1 netmask ffffff00 broadcast 10.98.2.255
   ```

```
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
         inet 10.98.3.10 netmask ffffff00 broadcast 10.98.3.255
         ether 0:0:0:0:0:0
cgtp0:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu \
1500 index 4
         inet 10.98.3.1 netmask ffffff00 broadcast 10.98.3.255
```

This output shows that there is no floating external address on the vice-master node.

Examining the Network Configuration Files

Each interface of each peer node must have the following configuration information:

- An entry in the `/etc/hosts` file for the node
- An `/etc/hostname.`*interface-name* file or an `/etc/nodename` file

Examine these files to understand the network configuration of a peer node. To find the name of the current node, see "Mapping the Network Configuration of a Peer Node" on page 47.

The network configuration files contain the following example information. In the following examples, the node name is `MEN-C250-N20`.

- `/etc/hostname.hme0` and `/etc/hostname.hme1`

  These files defines the host name for the `hme0` and `hme1` interface on a node.

  The `/etc/hostname.hme0:100` file defines the host name for an additional external interface configured on a node.

- `/etc/nodename`

  This file defines the host name of the `cgtp0` interface on a node.

- `/etc/hosts`

  This file contains a list of host names. For a cluster of three nodes, using the default class C addressing scheme, the `/etc/hosts` file is like this:

  Local host

  ```
  127.0.0.1 localhost
  ```

  Peer node 1 address triplet (Current node)

  ```
  10.250.1.20 MEN-C250-N20 MEN-C250-N20.localdomain loghost
  10.250.2.20 MEN-C250-N20-nic1 MEN-C250-N20-nic1.localdomain
  10.250.3.20 MEN-C250-N20-cgtp MEN-C250-N20-cgtp.localdomain
  ```

Peer node 2 address triplet

```
10.250.1.10 MEN-C250-N10
10.250.2.10 MEN-C250-N10-nic1
10.250.3.10 MEN-C250-N10-cgtp
```

Peer node 3 address triplet

```
10.250.1.30 NMEN-C250-N30
10.250.2.30 NMEN-C250-N30-nic1
10.250.3.30 NMEN-C250-N30-cgtp
```

External address

```
172.36.128.51 sol52
```

Floating address triplet

```
10.250.1.1 master-nic0
10.250.2.1 master-nic1
10.250.3.1 master-cgtp
```

Examining the Routes on a Node

At startup, the Cluster Membership Manager (CMM) creates a routing table for the interfaces on each peer node. The routing table is used by Carrier Grade Transport Protocol (CGTP) for data replication. To examine the routing table for a peer node, perform the following procedure. For simplicity, this procedure does not show external network access.

▼ To Examine the Routing Table for a Node

1. **Log in to the peer node whose routes you want to examine.**
2. **Display the routing table for the node:**

```
# netstat -r
```

For a node using the class C addressing scheme, output similar to this is displayed:

```
Routing Table: IPv4
  Destination           Gateway           Flags    Ref   Use  Interface
-------------------- -------------------- -----   ----- ------ ---------
10.250.3.255         10.250.1.255          UGHMS     1      0
10.250.3.255         10.250.2.255          UGHMS     1      0
NMEN-C250-N30-cgtp   NMEN-C250-N30         UGHMS     1   2557
NMEN-C250-N30-cgtp   NMEN-C250-N30-nic1    UGHMS     1   2554
master-cgtp0         master-nic0           UGHMS     1      1
master-cgtp0         master-nic1           UGHMS     1      1
```

```
MEN-C250-N10-cgtp    MEN-C250-N10         UGHMS     1   2644
MEN-C250-N10-cgtp    MEN-C250-N10-nic1    UGHMS     1   2640
10.250.1.0           MEN-C250-N20         U         1   3886  hme0
10.250.1.0           MEN-C250-N20         U         1      0  hme0:1
10.250.2.0           MEN-C250-N20-nic1    U         1   3866  hme1
10.250.2.0           MEN-C250-N20-nic1    U         1      0  hme1:1
10.250.3.0           MEN-C250-N20-cgtp    U         1      0  cgtp0
10.250.3.0           MEN-C250-N20-cgtp    U         1      0  cgtp0:1
default              10.250.1.253         UG        1      0
localhost            localhost            UH       17    747  lo0
```

For a description of the column headings, see "Output of the `netstat -r` Command" on page 56.

From the sample output, you can conclude the following facts:

- This is a three node cluster. The cluster contains the nodes `NMEN-C250-N30`, `MEN-C250-N10`, and the current node.
- The CGTP routes to broadcast `10.250.1.255` are as follows:

```
  Destination           Gateway           Flags    Ref   Use  Interface
-------------------- -------------------- -----    ----- ------ ---------
10.250.3.255         10.250.1.255         UGHMS     1      0
10.250.3.255         10.250.2.255         UGHMS     1      0
```

- The CGTP routes to node 30 are as follows:

```
  Destination           Gateway           Flags    Ref   Use  Interface
-------------------- -------------------- -----    ----- ------ ---------
NMEN-C250-N30-cgtp   NMEN-C250-N30        UGHMS     1   2557
NMEN-C250-N30-cgtp   NMEN-C250-N30-nic1   UGHMS     1   2554
```

- The CGTP routes to the CGTP floating address, `cgtp0:1`, are as follows:

```
  Destination           Gateway           Flags    Ref   Use  Interface
-------------------- -------------------- -----    ----- ------ ---------
master-cgtp          master-nic0          UGHMS     1      1
master-cgtp          master-nic1          UGHMS     1      1
```

- The CGTP routes to `MEN-C250-N10` are as follows:

```
  Destination           Gateway           Flags    Ref   Use  Interface
-------------------- -------------------- -----    ----- ------ ---------
MEN-C250-N10-cgtp    MEN-C250-N10         UGHMS     1   2644
MEN-C250-N10-cgtp    MEN-C250-N10-nic1    UGHMS     1   2640
```

The other entries in the table are standard Solaris routes.

To change the entries in the routing table, use the `route` command as described in the `route`(1M) man page.

Output of the `netstat -r` Command

The following table explains the output of the `netstat -r` command.

TABLE 4–1 Description of the Output of the `netstat -r` Command

Field	Description
`Flag`	■ `D` – The route was dynamically created using a redirect. ■ `G` – The route is through a gateway. ■ `H` – The route is to a host. ■ `M` – The route is a redundant route established with the `multirt` option. ■ `S` – The route is a redundant route established with the `setsrc` option. ■ `U` – The route is up.
`Ref`	Shows the current number of routes sharing the same link layer. If this value is greater than 1, the corresponding route cannot be deleted.
`Use`	Indicates the number of packets sent using a combined routing and address resolution or a broadcast route.
`Interface`	Lists the network interface used for the route.

The flags `D`, `G`, `H`, and `U` are part of standard Solaris routing. The flags `M` and `S` are used for CGTP. The flags configure the way in which data is replicated, as follows:

- If the flag `M` is set, the data is replicated through all routes that have the destination flag `M`.
- If the flag `S` is set, and if the application did not provide a source address for the data to be sent to, the source address is set to the value specified in the route.

PART III

Maintaining a Cluster

For information about how to maintain a cluster, see the following chapters:

Chapter 5 explains how to manage differences between files that are not shared. This chapter also describes how to configure the naming services.

Chapter 6 describes how to stop and start the Foundation Services, a node, or a cluster.

Chapter 7 describes how to install a patch for the Foundation Services or for the Solaris operating system.

Chapter 8 describes how to resize, add, and share a disk partition.

CHAPTER 5

Using and Managing System Files

For information about how to manage system files, see the following sections:

- "Using Data Caches in Shared File Systems" on page 61
- "Changing the Location of the Scoreboard Bitmap" on page 62
- "Managing Differences Between Files That Are Not Shared" on page 64
- "Using the Naming Services" on page 65

Using Data Caches in Shared File Systems

If the impact on performance is acceptable, do not use data and attribute caches when writing data to shared file systems. If it is necessary to use data and attribute caches to improve performance, ensure that your applications minimize the risk of using inconsistent data. Consider using the `O_SYNC` or `O_DSYNC` mount options on some files. For information about these options, see the `fcntl`(3head) man page.

Data and attribute caching is disabled by the `noac` mount option. The following procedure describes how to enable or disable the `noac` mount option.

▼ To Enable or Disable Data and Attribute Caching

1. **Log in to the vice-master node as superuser.**
2. **Open the `/etc/vfstab` file in a text editor.**
   - If data and attribute caching is disabled, the file should contain the `noac` option, as follows:

```
master-cgtp:/SUNWcgha/local/export/data -       \
/SUNWcgha/remote  nfs     -       no    rw,hard,fg,intr,noac

master-cgtp:/SUNWcgha/local/export/services/ha_v21/opt  \
-    /SUNWcgha/services   nfs     -   no   rw,hard,fg,intr,noac

master-cgtp:/SUNWcgha/local/export/services     -     \
/SUNWcgha/swdb  nfs    -       no     rw,hard,fg,intr,noac
```

- If data and attribute caching is enabled, the file should not contain the `noac` option, as follows:

```
master-cgtp:/SUNWcgha/local/export/data -       \
/SUNWcgha/remote  nfs     -       no    rw,hard,fg,intr

master-cgtp:/SUNWcgha/local/export/services/ha_v21/opt  \
-    /SUNWcgha/services   nfs     -   no   rw,hard,fg,intr

master-cgtp:/SUNWcgha/local/export/services     -     \
/SUNWcgha/swdb  nfs    -       no     rw,hard,fg,intr
```

3. **Remove or add the `noac` option, as required.**
   - To enable data and attribute caching, remove `noac` from the lists of options.
   - To disable data and attribute caching, add `noac` to the lists of options.

4. **Save and close the file.**

5. **Reboot the node:**

```
# uadmin 1 1
```

6. **Trigger a switchover, as described in "To Trigger a Switchover With `nhcmmstat`" on page 75.**

7. **Repeat Step 1 through Step 5 on the new vice-master node.**

8. **Log in to each of the diskless peer nodes or dataless peer nodes and repeat Step 2 through Step 5.**

Changing the Location of the Scoreboard Bitmap

When data is written to the master node, a write is made to the replicated partition on the disk and to the corresponding scoreboard bitmap. The scoreboard bitmap can be configured in two ways:

- The scoreboard bitmap can be stored on a replicated partition and updated every time that the corresponding data partition is updated.

- The scoreboard bitmap can be stored in memory and updated every time that the corresponding data partition is updated. The scoreboard bitmap is written to a replicated partition only when the node is shut down gracefully.

For information about the advantages and disadvantages of storing the scoreboard bitmap in memory, see "IP Mirroring" in the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*. For information about how to reconfigure the scoreboard bitmap, see the following procedure.

▼ To Change the Location of the Scoreboard Bitmap

1. **Log in to the vice-master node as superuser.**
2. **Open the `/usr/kernel/drv/rdc.conf` file in a text editor.**
    - If the scoreboard bitmap is stored on a replicated partition, the value of the `rdc_bitmap_mode` parameter is 1.
    - If the scoreboard bitmap is stored in memory, the value of the `rdc_bitmap_mode` parameter is 2.
3. **Change the value of the `rdc_bitmap_mode` parameter:**
    - To store the scoreboard bitmap on a replicated partition, set the value of the `rdc_bitmap_mode` parameter to 1.
    - To store the scoreboard bitmap in memory, set the value of the `rdc_bitmap_mode` parameter to 2.
4. **Save and close the file.**
5. **Reboot the node:**

    ```
    # uadmin 1 1
    ```

6. **Trigger a switchover, as described in "To Trigger a Switchover With `nhcmmstat`" on page 75.**
7. **Repeat Step 1 to Step 5 on the new vice-master node.**
8. **Verify that the master node and vice-master node are synchronized, as described in "To Verify That the Master Node and Vice-Master Node Are Synchronized" on page 42.**

Managing Differences Between Files That Are Not Shared

Files on a shared file system have the same content, as viewed from the master node and vice-master node. The following files are stored locally on the master node and vice-master node. The files must contain identical information, but they are not shared.

`cluster_nodes_table`
: Contains the *nodeid* and node name of each peer node. For more information, see the `cluster_nodes_table`(4) man page.

`/etc/hosts`
: Contains the hostnames of all nodes on the cluster network. For more information, see the `hosts`(4) man page.

`nhfs.conf`
: Describes the cluster configuration, including network interfaces, mirrored disk partitions, and the floating external address. For more information, see the `nhfs.conf`(4) man page.

To manage differences that exist between files that are not shared, performing the following procedure.

▼ To Manage Differences Between Files That Are Not Shared

1. **Log in to the master node as superuser.**
2. **Open or create the `/SUNWcgha/remote/etc/nhadmsync.conf` file in a text editor.**
3. **Specify the names of the files that you want to compare by adding them to the `nhadmsync.conf` file.**

   For more information, see the `nhadmsync.conf`(4) man page.
4. **Save the `nhadmsync.conf` file and exit the text editor.**
5. **Log in to the vice-master node, as superuser.**
6. **Repeat Step 2 through Step 4.**
7. **Verify that the listed files are the same on the master node and the vice-master node:**

   ```
   # nhadm synccheck
   ```

- If the files are not identical on the master node and vice-master node, analyze the differences between the copies of the files.
- If the differences between the files are acceptable, accept them:

  ```
  # nhadm syncgen
  ```

  If you accept the differences between two files, the differences will no longer be signaled by the `nhadm synccheck` command.

For more information about the `nhadm` command, see the `nhadm`(1M) man page.

Using the Naming Services

This section provides guidelines for using naming services on the Foundation Services.

If you use a naming service such as the Network Information Services (NIS) or the Domain Name System (DNS), avoid conflicts between the names of nodes and services by doing the following:

1. Verify that the names of nodes specified in the `/etc/hosts` file are used before node names generated by your naming service.

   The name assigned to a node during cluster configuration must not conflict with the name assigned to a node by the naming service.
2. Verify that the entries for `hosts`, `network`, and `services` in the `/etc/nsswitch.conf` file are set as follows:

   ```
   [...]
   hosts:      files [...]
   network:    files [...]
   [...]
   services:   files [...]
   ```

CHAPTER 6

Starting and Stopping Services, Nodes, and Clusters

This chapter describes how to stop and start the Foundation Services, a node, or a cluster. This chapter contains the following sections:

Stopping and Restarting the Foundation Services

Maintenance on a peer node can disrupt communication between this node and services and applications running on other peer nodes. During maintenance, you must isolate a node from the cluster by starting the node without the Foundation Services. After maintenance, reintegrate the node into the cluster by restarting the Foundation Services.

▼ To Start a Node Without the Foundation Services

1. **Log in as superuser to the node on which you want to stop the Foundation Services.**

2. **Create the `not_configured` file on the node:**

```
# touch /etc/opt/SUNWcgha/not_configured
```

3. **Reboot the node:**

   ```
   # init 6
   ```

   The node restarts without the Foundation Services running. If the node is the master node, this procedure causes a failover.

4. **Verify that the Foundation Services are not running**

   ```
   # pgrep -x nhcrfsd
   ```

   If the Foundation Services have been stopped, no process identifier should be displayed for the `nhcrfsd` daemon.

▼ To Restart the Foundation Services

Use this procedure to restart the Foundation Services on a node after performing the procedure in "To Start a Node Without the Foundation Services" on page 67.

1. **Log in as superuser to the node on which you want to restart the Foundation Services.**

2. **Check that the `/etc/opt/SUNWcgha/not_configured` file is not present.**

   If the `/etc/opt/SUNWcgha/not_configured` file is present, delete it.

3. **Reboot the node:**

   ```
   # init 6
   ```

4. **Verify the configuration of the node:**

   ```
   # nhadm check configuration
   ```

   If the node is configured correctly, the `nhadm` command does not encounter any errors.

   For information about the `nhadm` command, see the `nhadm`(1M) man page.

5. **Verify that the services have started correctly:**

   ```
   # nhadm check starting
   ```

   If the Foundation Services have started correctly, the `nhadm` command does not encounter any errors.

Stopping and Restarting Daemon Monitoring

Sometimes you need to stop Daemon Monitoring to investigate why a monitored daemon has failed. This section describes how to stop and restart Daemon Monitoring.

For information about the causes of daemon failure at startup and runtime, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

▼ To Stop Daemon Monitoring

Note – Use this procedure with extreme caution. Incorrect editing of the `/etc/inittab` file can lead to major problems on your cluster.

This procedure stops the Daemon Monitoring. On reboot, the Daemon Monitoring is not automatically restarted.

1. **Log in as superuser to the node on which you want to stop the `nhpmd` daemon.**

2. **Open the `/etc/inittab` file in a text editor.**

3. **Delete the string `.HA` from the `rc2.HA` and `rc3.HA` entries shown here.**

   ```
   s2:23:wait:/sbin/rc2.HA    >/dev/msglog 2<>/dev/msglog </dev/console
   s3:3:wait:/sbin/rc3.HA    >/dev/msglog 2<>/dev/msglog </dev/console
   ```

   This converts these entries back to the original Solaris `rc2` and `rc3` entries.

4. **Save and close the `/etc/inittab` file.**

5. **Reboot the node.**

   ```
   # init 6
   ```

   The Foundation Services start, and the Solaris and Foundation Services daemons that were monitored are no longer monitored.

▼ To Restart Daemon Monitoring

Note – Use this procedure with extreme caution. Incorrect editing of the `/etc/inittab` file can lead to major problems on your cluster.

If Daemon Monitoring was stopped using "To Stop Daemon Monitoring" on page 69, restart Daemon Monitoring as follows:

1. **Log in to the node on which you want to restart the Daemon Monitoring.**
2. **Open the `/etc/inittab` file in a text editor.**
3. **Paste the string `.HA` in the `rc2` and `rc3` entries of the `/etc/inittab` file.**

   The entries should be as follows:

   ```
   s2:23:wait:/sbin/rc2.HA    >/dev/msglog 2<>/dev/msglog </dev/console
   s3:3:wait:/sbin/rc3.HA    >/dev/msglog 2<>/dev/msglog </dev/console
   ```

4. **Use the `sync` command, as described in the `sync`(1M) man page.**
5. **Reboot the node.**

   The Foundation Services start and are monitored by the Daemon Monitor.

Shutting Down and Restarting a Node

This section describes how to shut down a node and how to restart a node. The consequences of stopping a node depend on the role of the node. If you shut down a master-eligible node, you no longer have a redundant cluster.

General Rules for Shutting Down a Node

To shut down nodes, observe the following rules:

▼ To Perform a Clean Reboot of a Node

- **Stop the operating system and reboot the node by typing the following command:**

  ```
  # init 6
  ```

▼ To Perform a Clean Poweroff of a Node

- **Shut the node down by typing the following command:**

  ```
  # init 5
  ```

▼ To Perform a Clean Halt of a Node

- **Go into the firmware and type the following command:**

  ```
  # init 0
  ```

▼ To Abruptly Reboot a Node

- **Reboot the node by typing the following command:**

  ```
  # uadmin 1 1 (A_REBOOT AD_BOOT)
  ```

  The node stops immediately without any further processing, and is rebooted.

▼ To Abruptly Power Off a Node

- **Power off the node by typing the following command:**

  ```
  # uadmin 1 6 (A_REBOOT AD_POWEROFF)
  ```

  The node stops immediately without any further processing.

▼ To Abruptly Halt a Node

- **Halt the node by typing the following command:**

  ```
  # uadmin 1 0 (A_REBOOT AD_HALT)
  ```

  The node stops immediately without any further processing.

Shutting Down a Node

This section describes how to shut down a master node, a vice-master node, a diskless node, and a dataless node.

▼ To Shut Down the Master Node

Before shutting down the master node, perform a switchover as described in "To Trigger a Switchover With `nhcmmstat`" on page 75. The vice-master node becomes the new master node, and the old master node should reboot as the new vice-master node. Then, shut down the new vice-master node as described in "To Shut Down the Vice-Master Node" on page 72.

Should you want to shut down the master node without first performing a switchover, do the following:

1. **Log in to the master node as superuser.**
2. **Shut down the master node:**

   ```
   # init 5
   ```

   The vice-master node becomes the master node. Because there are only two master-eligible nodes in the cluster and one is shut down, your cluster is not highly available. To restore high-availability, restart the stopped node.

▼ To Shut Down the Vice-Master Node

1. **Log in to the vice-master node as superuser.**
2. **Shut down the vice-master node.**

   ```
   # init 5
   ```

   Because there are only two master-eligible nodes in the cluster and one is shut down, your cluster is not highly available. To restore high-availability, restart the stopped node.

▼ To Shut Down a Diskless Node or Dataless Node

1. **Log in as superuser to the node you want to shut down.**
2. **Shut down the node:**

   ```
   # init 5
   ```

   When a diskless node or dataless node is shut down, there is no impact on the roles of the other peer nodes.

Restarting a Node

This section describes how to restart a node that has been stopped by one of the procedures in "Shutting Down a Node" on page 71.

▼ To Restart a Node

1. **Restart the node.**
   - If the node is powered off, power on the node.
   - If the node is not powered off but is at the open boot prompt, boot the node:

```
ok> boot
```

If the node is in single-user mode, go to multi-user mode using CTRL-D.

If the node is a peer node, restarting the node reintegrates it into the cluster.

2. **Log in to the restarted node as superuser.**
3. **Verify that the node has started correctly:**

```
# nhadm check
```

For more information, see the `nhadm`(1M) man page.

Shutting Down and Restarting a Cluster

This section describes how to shut down and restart a cluster.

▼ To Shut Down a Cluster

1. **Log in to a peer node as superuser.**
2. **Identify the role of each peer node:**

```
# nhcmmstat -c all
```

Record the role of each node.

3. **Shut down each diskless and dataless node:**

```
# init 5
```

4. **Verify that the vice-master node is synchronized with the master node:**

```
# /usr/opt/SUNWesm/sbin/scmadm -S -M
```

If the vice-master node is not synchronized with the master node, synchronize it:

```
# nhcrfsadm -f all
```

5. **Shut down the vice-master node by logging in to the vice-master node and typing this command:**

```
# init 5
```

6. **Shut down the master node by logging in to the master node and typing this command:**

```
# init 5
```

For further information about the `init` command, see the `init`(1M) man pages.

▼ To Restart a Cluster

This procedure describes how to restart a cluster that has been shut down as described in "To Shut Down a Cluster" on page 73.

Caution – To restart a cluster, you boot each peer node. The order in which you boot the nodes is important. Restart the nodes so that they have the same role as they had before cluster was shut down. If you do not maintain the roles of the nodes, you might lose data.

---

1. **Log in to the master node and in the console window type:**

   ```
   ok> boot
   ```

2. **When the node has finished booting, verify that the master node is correctly configured:**

   ```
   # nhadm check configuration
   ```

3. **Log in to the vice-master node, and in the console window type:**

   ```
   ok> boot
   ```

4. **When the node has finished booting, verify that the vice-master node is correctly configured:**

   ```
   # nhadm check configuration
   ```

5. **Log in to each diskless node or dataless node, and in the console window type:**

   ```
   ok> boot
   ```

6. **When the nodes have finished booting, verify that each node is correctly configured:**

   ```
   # nhadm check configuration
   ```

7. **From any node in the cluster, verify that the cluster has started up successfully:**

   ```
   # nhadm check starting
   ```

8. **Confirm that each node has the same role as it had before it was shut down.**

Caution – After an emergency shut down, the order in which the nodes are rebooted is important if availability or data integrity are a priority on your cluster. The order in which these nodes are restarted depends on the Data Management Policy you have selected in your initial cluster configuration. For more information, see the `nhfs.conf`(4) and `cluster_definition.conf`(4) man pages.

Triggering a Switchover

Before you perform a switchover, verify that the master and vice-master disks are synchronized, as described in "To Verify That the Master Node and Vice-Master Node Are Synchronized" on page 42. To trigger a switchover, perform the following procedure.

▼ To Trigger a Switchover With `nhcmmstat`

1. **Log in to the master node as superuser.**

2. **Trigger a switchover:**

   ```
   # nhcmmstat -c so
   ```

   - If there is a vice-master node qualified to become master, this node is elected as the master node. The old master node becomes the vice-master node.
   - If there is no potential master node, `nhcmmstat` does not perform the switchover.

3. **Verify the cluster configuration:**

   ```
   # nhadm check
   ```

   If the switchover was successful, the current node is the vice-master node.

4. **Verify that the current node is synchronized with the new master node:**

   ```
   # nhcmmstat -c vice
   ```

For more information, see the `nhcmmstat`(1M) man page.

Recovering a Cluster

This section describes how to recover when a cluster fails.

If the master node and the vice-master node both act as master nodes, this error is called *split brain*. For information about how to recover from split brain at startup and at runtime, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Troubleshooting Guide*.

▼ To Recover a Cluster After Failure

1. **Stop all of the nodes in the cluster:**

   ```
   # init 5
   ```

2. **Boot both of the master-eligible nodes in single user mode.**

   ```
   ok> boot -s
   ```

3. **Confirm that the master-eligible nodes are configured correctly.**

   For each master-eligible node, do the following:

   a. **Confirm that the following files exist and are not empty:**

      - `cluster_nodes_table`
      - `target.conf`

   b. **Reset the replication configuration (answer = `y`):**

      ```
      # /usr/opt/SUNWscm/sbin/dscfg -i
      ```

   c. **Re-create an empty replicated configuration file by typing `Y` at this prompt:**

      ```
      # (Type Y for YES) Y
      ```

      ```
      # /usr/opt/SUNWscm/sbin/dscfg -i -p /etc/opt/SUNWesm/pconfig
      ```

   d. **Synchronize the file system by using `/sbin/sync`.**

   e. **Stop the master-eligible node.**

4. **Boot the nodes in the following order:**

   a. **Boot the first master-eligible node. This node has the most up-to-date set of data.**

Caution – The node that becomes the vice-master node will have the recent file system data erased.

---

   b. **Confirm that the first master-eligible node has become the master node.**

   c. **Boot the second master-eligible node.**

   d. **Confirm that the second master-eligible node has become the vice-master node.**

   e. **Wait until the master node and vice-master node are synchronized.**

      This is a full resynchronization and might take some time.

   f. **Boot the diskless and dataless nodes if there are any.**

Diskless and dataless nodes can be booted in any order.

CHAPTER 7

Patching Software on the Cluster

When you are patching the software on the cluster, consult the patch README and your hardware documentation for any information about patch dependencies or special installation instructions. You can add a patch to the cluster during initial cluster installation with the `nhinstall` tool by adding the the patch details to the `addon.conf` file. For further information, see the `addon.conf`(4) and `nhinstall`(1M) man pages.

For information about how to manually add a patch to the Foundation Services and Solaris packages, see the following sections:

- "Introduction to Patching Software on a Cluster" on page 79
- "Patching a Nonshared Package on Both Master-Eligible Nodes" on page 81
- "Patching a Dataless Node" on page 82
- "Patching a Diskless Node" on page 82
- "Patching a Shared Package" on page 85

Introduction to Patching Software on a Cluster

Some Foundation Services packages are shared packages and some are not shared. A patch for a shared package is installed on the master node only. A patch for a nonshared package is installed on all peer nodes for which the patch is relevant.

Before you install a patch, consider the following:

- Whether you are patching shared or nonshared packages
- The type of node on which you are installing the patch

  If you are installing a patch on a diskless node, determine whether you are patching a package installed under `/usr` or `root`.

- Whether the cluster was installed by the `nhinstall` tool or manually.

  The location of the packages for diskless nodes depends on the installation method. Where the location of a package depends on the installation method, it is stated.

Information about the patches is held in the patch database. The information is node-specific for patches that are applied to nonshared packages. The database describes the file system where the patch contents are installed. To view the patches that are currently installed on a node, use the `patchadd -p` command. For further information, see the `patchadd`(1M) man page.

Choosing a Procedure to Patch Software on a Cluster

The following packages can be patched. The path to the patch database for each package and a pointer to how to install the patch are also provided.

- A Solaris package or nonshared Foundation Services package on the master-eligible nodes

  The patch database for these packages is at `/var/sadm/install`. To install a patch for a Solaris package or a nonshared Foundation Services package on a master-eligible node, see "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.
- A Solaris package or nonshared Foundation Services package on a dataless node

  The patch database for these packages is at `/var/sadm/install` on the dataless node. To install a patch on a dataless node, see "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.
- A Solaris package for the `/usr` directory common to diskless nodes
  - On a cluster, the patch database for diskless nodes is at `/export/root` on the master node. To install a Solaris patch for a diskless node on the `/usr` directory, see "To Patch the `/usr` Directory for Diskless Nodes" on page 83.
- A Solaris package or Foundation Services package for the `root` file system of each diskless node
  - On a cluster, the patch database for diskless nodes is at `/export/root/`*diskless-node-name* on the master node. To install a patch for a diskless node on the `root` file, see "To Patch the `root` File System for Diskless Nodes" on page 84.
- Shared packages

  A shared package can be a Foundation Services package or a user application. The patch database for shared packages is at `/SUNWcgha/local/export/services`. To install a patch on a shared package, see "Patching a Shared Package" on page 85.

Overwriting Modified Foundation Services Files

The following patches overwrite Foundation Services files:

- A Solaris patch that changes the IP stack. The IP stack is modified during installation of CGTP.
- A Solaris patch that changes the modified `rc2` and `rc3` scripts. The modified scripts are required by the Daemon Monitor.

If you install one of these patches, you must reinstall the Foundation Services.

Patching a Nonshared Package on Both Master-Eligible Nodes

This section describes how to patch a nonshared package on the master-eligible nodes. You must install the patch on both master-eligible nodes.

▼ To Patch a Nonshared Package on Both Master-Eligible Nodes

1. **Log in to a master-eligible node as superuser.**
2. **Consult the patch README for the *patchid* and for any constraints on applying this patch.**

   Constraints can include the requirement to be in single-user mode, or the requirement to stop the node or the cluster.
3. **Copy the patch to the `/var/spool/patch` directory.**

   The `/var/spool/patch` directory is the default directory where the `patchadd` command looks for a patch.
4. **Take steps to comply with any constraints described in the patch README.**
5. **Apply the patch.**
   - If you have copied the patch to `/var/spool/patch`, install the patch as follows:

     ```
     # patchadd patchid
     ```
   - If you have not copied the patch to `/var/spool/patch`, specify the location of the patch by using the `-M` option of the `patchadd` command:

```
# patchadd -M patch-directory  patchid
```

6. **Verify that the patch has been installed:**

   ```
   # patchadd -p
   ```

   A list of all patches on the node is displayed on the screen.

   ```
   # patchadd -p | grep patchid
   ```

   The patch for which you specified the `patchid` is displayed, if this patch is installed.

7. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

   For more verification procedures, see Chapter 3.

8. **Log in to the second master-eligible node as superuser.**

9. **Repeat Step 3 through Step 7.**

Patching a Dataless Node

This section describes how to install a patch on a dataless node. You must install the patch on each dataless node.

▼ To Patch a Dataless Node

1. **Log in to a dataless node as superuser.**

2. **Perform Step 2 through Step 7 of "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.**

3. **Repeat Step 1 through Step 2 for each dataless node in the cluster.**

Patching a Diskless Node

Solaris packages for diskless nodes are installed on the master node, in the `/usr` directory or the `root` directory. Before proceeding, determine whether the package you want to patch is installed in the `/usr` directory or the `root` directory.

For information about how to apply a patch to a package installed in the `/usr` directory, see "Patching the `/usr` Directory for a Diskless Node" on page 83. For information about how to apply a patch to a package installed in the `root` directory, see "Patching the `root` Directory for Diskless Nodes" on page 84.

Patching the `/usr` Directory for a Diskless Node

This section describes how to apply a patch to a package installed in the `/usr` directory for diskless nodes. The `/usr` directory is common to the diskless nodes in a cluster.

▼ To Patch the `/usr` Directory for Diskless Nodes

1. **Log in to the master node as superuser.**

   The master node must be running the Foundation Services when you install a patch for diskless nodes.

2. **Perform Step 2 through Step 4 of "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.**

3. **Apply the patch.**

   - If you have copied the patch to `/var/spool/patch`, install the patch as follows:

     ```
     # patchadd -S target-OS patchid
     ```

     Possible values for *target-OS* are `Solaris_9` or `Solaris_8`.

   - If you have not copied the patch to `/var/spool/patch`, specify the location of the patch (*patch-directory*) by using the `-M` option of the `patchadd` command:

     ```
     # patchadd -M patch-directory -S target-OS patchid
     ```

4. **Verify that the patch has been installed:**

   ```
   # patchadd -p target-OS
   ```

   where *target-OS* is the Solaris operating system. For example, `Solaris_9` or `Solaris_8`.

   A list of all patches on the node is displayed on the screen.

5. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

   For more verification procedures, see Chapter 3.

Patching the `root` Directory for Diskless Nodes

This section describes how to apply a patch to a package installed in the `root` directory for diskless nodes.

▼ To Patch the `root` File System for Diskless Nodes

1. **Log in to the master node as superuser.**

   The master node must be running the Foundation Services when you install a patch for diskless nodes.

2. **Perform Step 2 through Step 4 of "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.**

3. **Apply the patch for a diskless node:**

   - If you have copied the patch to `/var/spool/patch`, install the patch as follows:

     ```
     # patchadd -R /export/root/diskless-node-name  patchid
     ```

   - If you have not copied the patch to `/var/spool/patch`, specify the location of the patch using the `-M` option of the `patchadd` command:

     ```
     # patchadd -M patch-directory  -R /export/root/diskless-node-name  \
     patchid
     ```

4. **Reboot the diskless node:**

   ```
   # init 6
   ```

5. **Verify that the patch has been installed:**

   ```
   # patchadd -p -R /export/root/diskless-name
   ```

   A list of all patches on the node is displayed on the screen.

6. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

   For more verification procedures, see Chapter 3.

7. **Repeat Step 3 through Step 6 for each diskless node with an `/export/root/`*diskless-name* directory on the master node.**

Patching a Shared Package

This section describes how to install a patch on a shared Foundation Services package.

▼ To Patch a Shared Package on a Cluster

1. **Log in to the master node as superuser.**

   The master node must be running the Foundation Services when you install a shared package.

2. **Perform Step 2 through Step 4 of "To Patch a Nonshared Package on Both Master-Eligible Nodes" on page 81.**

3. **If your cluster was installed by the `nhinstall` tool, create the `INST_RELEASE` file used by the `patchadd` command:**

   ```
   # nhadm confshare
   ```

   For information about `nhadm confshare`, see the `nhadm`(1M)man page.

4. **Apply the patch.**

   - If you have copied the patch to `/var/spool/patch`, install the patch as follows:

     ```
     # patchadd -R /SUNWcgha/local/export/services patchid
     ```

   - If you have not copied the patch to `/var/spool/patch`, specify the location of the patch by using the `-M` option of the `patchadd` command:

     ```
     # patchadd -M patch-directory -R /SUNWcgha/local/export/services patchid
     ```

5. **Verify that the patch has been installed:**

   ```
   # patchadd -p -R /SUNWcgha/local/export/services
   ```

   A list of all patches on the node is displayed on the screen.

   You can also search for a specific patch as follows:

   ```
   # patchadd -p -R /SUNWcgha/local/export/services | grep patchid
   ```

6. **If you stopped the cluster to install the patch, restart the cluster.**

   For information, see "To Restart a Cluster" on page 74.

7. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

   For more verification procedures, see Chapter 3.

CHAPTER **8**

Modifying and Adding Disk Partitions

This chapter describes how to examine or modify the configuration of a disk partition on the master-eligible nodes. Master-eligible nodes can have more than one disk. The disk partitions discussed in this chapter pertain to the disk that contains the cluster configuration. For more information about disk partitioning, see "Volume Management" in the *Netra High Availability Suite Foundation Services 2.1 6/03 Overview*.

The disk on a dataless node is not used to store cluster data and is not discussed in this chapter. For information about the initial disk configuration of a dataless node, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

This chapter contains the following sections:

- "Using the `format` Utility to Display and Modify the Configuration of a Disk Partition" on page 87
- "Increasing the Size of a Mirrored Data Partition on a Physical Disk" on page 91
- "Increasing the Size of a Mirrored Data Partition on a Virtual Disk" on page 93
- "Adding a Mirrored Data Partition to a Virtual Disk" on page 95

Using the `format` Utility to Display and Modify the Configuration of a Disk Partition

This section describes how to use the `format` utility to display and modify the configuration of a disk partition. For additional information, see the `format`(1M) man page.

Before you change the configuration of a disk partition, you must stop all the nodes in the cluster. If you change the size of a data partition on one master-eligible node, you must change the size of the corresponding bitmap partition on that disk. You must also change the size of the corresponding data partition and bitmap partition on the other master-eligible node.

▼ To Display the Configuration of a Disk Partition

Use this procedure to identify the name of each partition on a disk and to determine whether a disk partition is big enough.

1. **Log in to a master-eligible node as superuser.**
2. **Run the format utility:**

   ```
   # format
   ```

   Output similar to this is displayed in the console window:

   ```
   Searching for disks...done


   AVAILABLE DISK SELECTIONS:
          0. c0t0d0 <SUN36G cyl 24620 alt 2 hd 27 sec 107>
             /pci@1f,0/pci@1,1/scsi@2/sd@0,0
   Specify disk (enter its number):
   ```

3. **Choose the disk you want to examine:**

   ```
   Specify disk (enter its number): 0
   ```

   Output similar to this is displayed in the console window:

   ```
   selecting c0t0d0
   [disk formatted]
   Warning: Current Disk has mounted partitions.


   FORMAT MENU:
           disk       - select a disk
           type       - select (define) a disk type
           partition  - select (define) a partition table
           current    - describe the current disk
           format     - format and analyze the disk
           repair     - repair a defective sector
           label      - write label to the disk
           analyze    - surface analysis
           defect     - defect list management
           backup     - search for backup labels
           verify     - read and display labels
           save       - save new disk/partition definitions
           inquiry    - show vendor, product and revision
           volname    - set 8-character volume name
   ```

```
        !<cmd>     - execute <cmd>, then return
        quit
```

4. **Choose the partition option:**

   ```
   format> partition
   ```

   Output similar to this is displayed in the console window:

   ```
   PARTITION MENU:
           0      - change ‘0’ partition
           1      - change ‘1’ partition
           2      - change ‘2’ partition
           3      - change ‘3’ partition
           4      - change ‘4’ partition
           5      - change ‘5’ partition
           6      - change ‘6’ partition
           7      - change ‘7’ partition
           select - select a predefined table
           modify - modify a predefined partition table
           name   - name the current table
           print  - display the current table
           label  - write partition map and label to the disk
           !<cmd> - execute <cmd>, then return
           quit
   ```

5. **Display the configuration of the current disk partitions:**

   ```
   partition> print
   ```

   Output similar to this is displayed in the console window:

   ```
   Current partition table (original):
   Total disk cylinders available: 24620 + 2 (reserved cylinders)

   Part      Tag    Flag     Cylinders         Size            Blocks
     0       root    wm        0 -  1451        2.00GB    (1452/0/0)   4194828
     1       swap    wu     1452 -  2177        1.00GB    (726/0/0)    2097414
     2     backup    wm        0 - 24619       33.92GB    (24620/0/0) 71127180
     3 unassigned    wm     2178 -  3629        2.00GB    (1452/0/0)   4194828
     4 unassigned    wm     3630 -  3771      200.31MB    (142/0/0)     410238
     5 unassigned    wm     3772 -  3772        1.41MB    (1/0/0)         2889
     6 unassigned    wm     3773 -  3773        1.41MB    (1/0/0)         2889
     7 unassigned    wm        0                0         (0/0/0)            0
   ```

6. **Quit if you do not want to modify the size of a partition:**

   ```
   partition> quit
   ```

   This output is displayed in the console window:

   ```
   FORMAT MENU:
           disk       - select a disk
           type       - select (define) a disk type
           partition  - select (define) a partition table
           current    - describe the current disk
           format     - format and analyze the disk
           repair     - repair a defective sector
   ```

```
        label      - write label to the disk
        analyze    - surface analysis
        defect     - defect list management
        backup     - search for backup labels
        verify     - read and display labels
        save       - save new disk/partition definitions
        inquiry    - show vendor, product and revision
        volname    - set 8-character volume name
        !<cmd>     - execute <cmd>, then return
        quit
```

```
format> quit
```

You return to the console prompt, `#`.

▼ To Change the Size of a Disk Partition on a Master-Eligible Node

1. **Perform Step 1 through Step 5 of "To Display the Configuration of a Disk Partition" on page 88.**
2. **Specify the physical partition you want to modify.**

   Type the entry from the `Part` column that corresponds to this partition:

   ```
   partition> Part-entry
   ```

   Information about the partition that you have chosen is displayed in the console window. For example, partition 3 produces this output:

   ```
   Part      Tag    Flag     Cylinders         Size            Blocks
     3 unassigned    wm     2178 -  3629        2.00GB    (1452/0/0)   4194828
   ```
3. **When the `format` tool prompts you to change some of the parameters of the partition configuration, press Return to accept the existing configuration.**

   ```
   Enter partition id tag[unassigned]:
   Enter partition permission flags[wm]:
   Enter new starting cyl[2178]:
   ```
4. **Change the size of the partition:**

   ```
   Enter partition size[4194828b, 1452c, 2048.26mb, 2.00gb]: new-partition-size
   ```
5. **Write the new partition layout to the disk:**

   ```
   partition> label
   ```
6. **Confirm that you want to change the partition size:**

   ```
   Ready to label disk, continue? y
   ```
7. **Verify that your modifications have been implemented:**

   ```
   partition> print
   ```

8. **Quit the `format` utility:**

```
FORMAT MENU:
        disk       - select a disk
        type       - select (define) a disk type
        partition  - select (define) a partition table
        current    - describe the current disk
        format     - format and analyze the disk
        repair     - repair a defective sector
        label      - write label to the disk
        analyze    - surface analysis
        defect     - defect list management
        backup     - search for backup labels
        verify     - read and display labels
        save       - save new disk/partition definitions
        inquiry    - show vendor, product and revision
        volname    - set 8-character volume name
        !<cmd>     - execute <cmd>, then return
        quit
```

```
format> quit
```

You return to the console prompt, `#`.

Increasing the Size of a Mirrored Data Partition on a Physical Disk

This section describes how to increase the size of a mirrored data partition on a physical disk.

▼ To Increase the Size of a Mirrored Data Partition on a Physical Disk

1. **Stop the cluster.**

   For information, see "To Shut Down a Cluster" on page 73.

2. **Log in to a master-eligible node in single-user mode, as superuser:**

   ```
   ok> boot -s
   ```

3. **Back up the partition by using a tool such as `ufsdump`.**

   For information, see the `ufsdump`(1M) man page.

4. **Identify the name of the data partition that you want to modify.**

For information, see "To Display the Configuration of a Disk Partition" on page 88.

5. **Change the size of the data partition.**

   For information, see "To Change the Size of a Disk Partition on a Master-Eligible Node" on page 90.

6. **Verify that the bitmap partition is the correct size.**

   a. **Identify the name of the local bitmap partition associated with the chosen data partition.**

      For information, see "To Display the Configuration of a Disk Partition" on page 88.

   b. **Confirm that the bitmap partition is at least the following size:**

      1 Kbyte + 4 Kbytes per Gbyte of data in the associated data partition

      If the bitmap partition is not big enough, increase the size of this partition, as described in "To Change the Size of a Disk Partition on a Master-Eligible Node" on page 90.

7. **Restore the data on the partition using the `ufsrestore` command.**

   For information, see the `ufsrestore`(1M) man page.

8. **Reset the replication configuration:**

```
# /usr/opt/SUNWscm/sbin/dscfg -i
```

9. **Re-create an empty replicated configuration file by typing `Y` at this prompt:**

```
# (Type Y for YES) Y
```

```
# /usr/opt/SUNWscm/sbin/dscfg -i -p /etc/opt/SUNWesm/pconfig
```

10. **Repeat Step 2 through Step 9 for the second master-eligible node.**

11. **Reboot the original master node:**

```
# init 6
```

12. **Reboot the original vice-master node:**

```
# init 6
```

13. **Log in to the master node as superuser.**

14. **Resynchronize the vice-master node with the master node:**

```
# nhcrfsadm -f all
```

   Do not perform a switchover before the resynchronization is complete.

15. **Verify that the vice-master node is synchronized with the master node:**

```
# /usr/opt/SUNWesm/sbin/scmadm -S -M
```

16. **Restart the diskless and dataless nodes.**

    For information, see "To Restart a Node" on page 72.

17. **Verify that the nodes have the correct configuration:**

    ```
    # nhadm check
    ```

    For information, see the `nhadm`(1M) man page.

Increasing the Size of a Mirrored Data Partition on a Virtual Disk

This section describes how to increase the size of a mirrored data partition created by the Solaris Volume Manager. This mirrored partition is called a *soft partition*. Perform this procedure to facilitate backup or to increase the partition size available to services or applications.

▼ To Increase the Size of a Mirrored Data Partition on a Virtual Disk

1. **Stop the cluster.**

   For information, see "To Shut Down a Cluster" on page 73.

2. **Log in to a master-eligible node in single-user mode, as superuser:**

   ```
   ok> boot -s
   ```

3. **Back up the partition by using a tool such as `ufsdump`.**

   For information, see the `ufsdump`(1M) man page.

4. **Mount the file system.**

   For example, mount `dsk/d20` on `home2`, as follows:

   ```
   # mount /dev/md/dsk/d20 /home2
   ```

5. **Add space to the virtual disk.**

   For example, add 10 Gbytes to the `d20` device:

   ```
   # metattach d20 10g
   ```

   For further information about virtual disks, see the *Solaris Volume Manager Administration Guide*.

6. **Grow the file system to its new size:**

```
# growfs -M /home2 /dev/md/rdsk/d20
```

7. **Verify that the bitmap partition is the correct size.**
    1. **Identify the name of the local bitmap partition associated with the chosen data partition.**

       For information, see "To Display the Configuration of a Disk Partition" on page 88.
    2. **Confirm that the bitmap partition is at least the following size:**

       1 Kbyte + 4 Kbytes per Gbyte of data in the associated data partition

       If the bitmap partition is not big enough, increase the size of this partition as described in "To Change the Size of a Disk Partition on a Master-Eligible Node" on page 90.

8. **Reset the replication configuration:**

```
# /usr/opt/SUNWscm/sbin/dscfg -i
```

9. **Re-create an empty replicated configuration file by typing `Y` at this prompt:**

```
# (Type Y for YES) Y
```

```
# /usr/opt/SUNWscm/sbin/dscfg -i -p /etc/opt/SUNWesm/pconfig
```

10. **Restore the backup of the disk using the `ufsrestore` command.**

    For information, see the `ufsrestore`(1M) man page.

11. **Repeat Step 2 through Step 10 for the second master-eligible node.**

12. **Reboot the master node:**

```
# init 6
```

13. **Reboot the vice-master node:**

```
# init 6
```

14. **Log in to the master node as superuser.**

15. **Resynchronize the vice-master node with the master node:**

```
# nhcrfsadm -f all
```

    Do not perform a switchover before the resynchronization is complete.

16. **Verify that the vice-master node is synchronized with the master node.**

```
# /usr/opt/SUNWesm/sbin/scmadm -S -M
```

17. **Restart the diskless nodes or dataless nodes.**

    For information, see "To Restart a Node" on page 72.

18. **Confirm that the nodes have the correct configuration.**

```
# nhadm check
```

For information, see the `nhadm`(1M) man page.

Adding a Mirrored Data Partition to a Virtual Disk

This section describes how to add a mirrored data partition to your disk configuration. To configure a virtual disk on the master-eligible nodes, you must include Solaris Volume Manager or the Solstice DiskSuite™ in the initial cluster configuration. For information about configuring Solaris Volume Manager, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

▼ To Add a Mirrored Data Partition to a Virtual Disk

1. **Stop the cluster.**

   For information, see "To Shut Down a Cluster" on page 73.

2. **Log in to a master-eligible node in single-user mode, as superuser:**

   ```
   ok> boot -s
   ```

3. **Create two virtual disk partitions: one data partition and one bitmap partition.**

   For information, see the following documents:

   - For Solaris 8, "Creating DiskSuite Objects" in the *Solstice DiskSuite 4.2 User's Guide*
   - For Solaris 9, "Creating Soft Partitions" in the *Solaris Volume Manager Administration Guide*

4. **Format the disk to include the configuration information for the two new partitions.**

   For information, see the `format`(1M) man page.

5. **Ensure that the partition is replicated, by adding the `RNFS.Slice` parameter for the new soft partition to the `nhfs.conf` file.**

   ```
   RNFS.Slice = slice-description
   ```

   For information, see the `nhfs.conf`(4) man page.

6. **Add the `RNFS.Slice` parameter for the new partitions to the `nhfs.conf` file.**

   The `RNFS.Slice` parameter is used to replicate the partition.

7. **Repeat Step 2 through Step 6 for the second master-eligible node.**

   The changes you make must be identical on both master-eligible nodes.

8. **Reboot all of the peer nodes.**

   For information, see "To Restart a Cluster" on page 74.

9. **Verify that the vice-master node is synchronized with the master node:**

   ```
   # /usr/opt/SUNWesm/sbin/scmadm -S -M
   ```

   If the vice-master node is not synchronized with the master node, resynchronize it:

   ```
   # nhcrfsadm -f all
   ```

10. **Confirm that the nodes have the correct configuration:**

    ```
    # nhadm check
    ```

    For information, see the `nhadm`(1M) man page.

PART **IV**

Adding and Replacing Hardware

For information about how to add nodes to a cluster or replace cluster hardware, see the following chapters:

- Chapter 9 describes the options for adding a diskless node or dataless node to a cluster. This chapter contains pointers to the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.
- Chapter 10 describes what to consider before replacing hardware in a cluster.
- Chapter 11 describes how to replace a board.
- Chapter 12 describes how to replace Ethernet cards.
- Chapter 13 describes how to replace a disk on a dataless node or the vice-master node.

Do not replace hardware on the master node. Instead, perform a switchover as described in "Triggering a Switchover" on page 75. A switchover forces the master node to become the vice-master node. You can then change the hardware on the vice-master node.

CHAPTER **9**

Adding a Node to a Cluster

For a description of the options for adding a diskless node or dataless node to a cluster, see the following sections:

- "Deciding How to Add a Node to a Cluster" on page 99
- "Determining Whether a Cluster Has Predefined Nodes" on page 100

Deciding How to Add a Node to a Cluster

Before adding a node to a cluster, ask the following questions:

- Is the node of a supported hardware type?

  For information about supported hardware types, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Hardware Guide* and the *Netra High Availability Suite Foundation Services 2.1 6/03 Release Notes*.

- How was the cluster originally installed?

  The way you add a node to a cluster depends on how the cluster was originally installed. A cluster can be installed by the `nhinstall` tool or manually. To determine whether a cluster was installed by the `nhinstall` tool or manually, consult the system operator who installed the cluster.

- Is the node already defined in the cluster configuration?

  A node that is defined as part of a cluster but is not physically connected to the cluster is called a *predefined node*. To find out if your cluster contains predefined nodes, see "To Determine Whether the Cluster Has Predefined Nodes" on page 100.

For information about how to add a node to a cluster installed by the `nhinstall` tool or manually, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

To use the CMM API to add or remove a node from a cluster, see the `cmm_config_reload`(3CMM) man page.

Determining Whether a Cluster Has Predefined Nodes

To determine whether a cluster has predefined nodes, perform the following procedure.

▼ To Determine Whether the Cluster Has Predefined Nodes

1. **Log in to the master node as superuser.**
2. **Determine how many nodes are up and running in the cluster:**

   ```
   # nhcmmstat -c all
   ```

   This command returns information about the nodes that are up and running in the cluster, but does not return information about nodes that are physically present but down.
3. **Determine how many nodes are defined in the cluster configuration.**
   1. **Look at the `cluster_nodes_table` file.**

      The `/etc/opt/SUNWcgha/cluster_nodes_table` file contains a line for each node in the cluster definition. Count the number of nodes configured in this file.
   2. **If there is an `/export/root` directory on the master node, examine its contents.**

      There is one `/`*diskless-node-name* directory for each diskless node in the cluster.
4. **Count the number of nodes physically present in the cluster.**

   Use the output of the `nhcmmstat` command and your knowledge of any nodes that are physically present but down. If a cluster has predefined nodes that are not diskless nodes, they are dataless nodes.

CHAPTER 10

Preparing to Replace Hardware in a Cluster

For information about what to consider before replacing hardware in a cluster, see the following sections.

- "Guidelines for Replacing Hardware" on page 101
- "Reference for Replacing Hardware" on page 102

Guidelines for Replacing Hardware

Follow these guidelines when replacing hardware on your cluster:

- Before replacing hardware on the master node, perform a switchover and then replace the hardware on the vice-master node.
- Replace hardware of one type with hardware of the same type, for the following reasons:
  - The vice-master node and master node must have the same hardware and software configuration.
  - If several nodes are running the same applications, the nodes might require the same hardware.

  To replace hardware of one type with hardware of another type, you must reinstall the node with a configuration adapted to the new hardware type.
- Perform a procedure for each item of hardware that you change:
  - If you change one item of hardware, perform the procedure for that type of hardware. For example, if you replace a disk perform only the procedure to change the disk.
  - If you change a board that contains Ethernet cards, perform the procedure to change the board *and* the procedure to change the Ethernet card.

- If you change a board that contains Ethernet cards and a disk, perform the procedure to change the board, the procedure to change the Ethernet cards, *and* the procedure to change the disk.

Reference for Replacing Hardware

The following table points to the procedures for replacing boards, Ethernet cards, and disks, for each type of node.

TABLE 10–1 Reference for Replacing Hardware

Node Type	Hardware Type	For information, see
Vice-master	CPU Board	"Replacing a Board on a Node" on page 103
	Ethernet Card	"Replacing Ethernet Cards on the Vice-Master Node or a Dataless Node" on page 107
	Disk	"Replacing the Disk on the Vice-Master Node" on page 111
Diskless	CPU Board	"Replacing a Board on a Node" on page 103 or "Replacing a Board on a Diskless Node With the DHCP Client ID Boot Policy" on page 104
	Ethernet Card	"Replacing Ethernet Cards on a Diskless Node" on page 108
	Disk	N/A
Dataless	CPU Board	"Replacing a Board on a Node" on page 103
	Ethernet Card	"Replacing Ethernet Cards on the Vice-Master Node or a Dataless Node" on page 107
	Disk	"Replacing a Dataless Node Disk" on page 112

CHAPTER 11

Replacing a CPU Board

For information about how to replace a CPU board on a node, see the following sections:

- "Replacing a Board on a Node" on page 103
- "Replacing a Board on a Diskless Node With the DHCP Client ID Boot Policy" on page 104

Replacing a Board on a Node

To replace the board on a the vice-master node, a diskless node, or a dataless node, perform the following procedure. If the node is a diskless node that is using the DHCP client ID boot policy, perform the procedure in "Replacing a Board on a Diskless Node With the DHCP Client ID Boot Policy" on page 104.

▼ To Replace a Board on a Node

1. **Verify that the new board is of the same type as the old board.**
2. **Replace the board using information in the hardware documentation on `http://www.sun.com/products-n-solutions/hardware/docs/`.**
3. **Log in to the new node.**
4. **Get the `ok` prompt.**
5. **Configure the OpenBoot™ PROM (OBP) parameters.**

   The following examples show the OBP parameters for a Netra T1 diskless node and a Netra T1 master-eligible node.

- A Netra T1 diskless node has these OBP parameters:

  ```
  ok> setenv local-mac-address? true
  ok> setenv auto-boot? true
  ok> setenv diag-switch? false
  ok> setenv boot-device net:dhcp,,,,5 net2:dhcp,,,,5
  ```

- A Netra T1 master-eligible node or dataless node has these OBP parameters:

  ```
  ok> setenv local-mac-address? true
  ok> setenv auto-boot? true
  ok> setenv diag-switch? false
  ok> setenv boot-device disk net
  ```

6. **Reboot the node:**

   ```
   ok> boot
   ```

7. **Log into the node as superuser.**

8. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

Replacing a Board on a Diskless Node With the DHCP Client ID Boot Policy

To replace the board on a diskless node that is using the DHCP client ID boot policy, perform the following procedure.

▼ To Replace a Board on Diskless Nodes With the DHCP Client ID Boot Policy

This procedure is an example of how to replace a board on a diskless node in a Netra CT 410, Netra CT 810 configuration. For information specific to your hardware configuration, see the hardware documentation.

1. **Retrieve the client identifier in ASCII format, for the diskless node.**

   If the node is running, you can retrieve the client identifier for the diskless node, as follows.

   a. **Log in to the diskless node as superuser.**

   b. **Get the `ok` prompt.**

c. **Display the client identifier of the diskless node.**

   ```
   ok> printenv dhcp-clientid
   client-id-name
   ```

   In this example the client identifier is the string *client-id-name*.

2. **Perform Step 1 to Step 5 of "Replacing a Board on a Node" on page 103.**

3. **Define the client identifier for the new diskless node.**

   ```
   ok> setenv dhcp-clientid client-id-name
   ```

4. **Perform Step 6 to Step 8 of "Replacing a Board on a Node" on page 103.**

CHAPTER 12

Replacing Ethernet Cards

This chapter describes how to replace Ethernet cards on a peer node. To replace Ethernet cards with cards of a different type, you must reinstall the node. To replace the Ethernet cards on diskless nodes, select the procedure according to the boot policy of the diskless node. This chapter contains the following sections:

- "Replacing Ethernet Cards on the Vice-Master Node or a Dataless Node" on page 107
- "Replacing Ethernet Cards on a Diskless Node" on page 108

Replacing Ethernet Cards on the Vice-Master Node or a Dataless Node

To replace the Ethernet cards on the vice-master node or a dataless node, perform the following procedure.

▼ To Replace Ethernet Cards on the Vice-Master Node or a Dataless Node

1. **Verify that the new Ethernet cards are of the same type as the old Ethernet cards.**
2. **Replace the Ethernet cards using information in the hardware documentation on `http://www.sun.com/products-n-solutions/hardware/docs/`.**
3. **Power on the node.**
4. **Log in to the node as superuser.**
5. **Verify that the node is configured correctly:**

```
# nhadm check
```

Replacing Ethernet Cards on a Diskless Node

To replace the Ethernet cards on diskless nodes with the DHCP dynamic boot policy or the DHCP client ID boot policy, perform the procedure in "Replacing Ethernet Cards on the Vice-Master Node or a Dataless Node" on page 107. To replace the Ethernet cards on diskless nodes with the DHCP static boot policy, perform the following procedure.

▼ To Replace Ethernet Cards on a Diskless Node With the DHCP Static Boot Policy

1. **Verify that the new Ethernet cards are of the same type as the old Ethernet cards.**
2. **Identify the IP address - Ethernet address couplet for the network interface cards that are to be replaced.**
3. **Replace the Ethernet cards by using the hardware documentation on `http://www.sun.com/products-n-solutions/hardware/docs/`.**
4. **Record the Ethernet addresses of the network cards on the new node.**

   To find the Ethernet addresses of the network cards, perform the following step:

   a. **Log in to the diskless node.**

   b. **Identify the Ethernet address of `NIC0`:**

   ```
   ok> banner
   ```

   The Ethernet address of `NIC0` is provided in the output.

   The Ethernet address of `NIC1` is derived as follows:

   ```
   NIC0 + 0x1
   ```

   For example, if the output of the `banner` command is this:

   ```
   Ethernet address 8:0:20:fa:2a:6e, Host ID: 80fa2a6e
   ```

   The Ethernet address of `NIC0` is `8:0:20:fa:2a:6e`, and the Ethernet address of `NIC1` is `8:0:20:fa:2a:6f`.

   In the DHCP configuration files, the Ethernet addresses of `NIC0` and `NIC1` are given as `01080020FA2A6E` and `01080020FA2A6F`, respectively.

For another example, the output of the `banner` command is this:

```
Ethernet address 8:0:20:f9:b3:60, Host ID: 80f9b360
```

In the DHCP configuration files in the `/SUNWcgha/remote/var/dhcp/` directory, the Ethernet addresses of `NIC0` and `NIC1` are given as `01080020F9B360` and `01080020F9B361`, respectively.

5. **Log in to the master node as superuser.**

6. **Modify the DHCP configuration for `NIC0`:**

```
# pntadm -M NIC01IP-address -i newEthernet-address \
 -f 'PERMANENT+MANUAL' -m NIC0IP-address subnet1
```

The parameters of this command are as follows:

NIC0IP-address	The IP address of the `NIC0` interface
newEthernet-address	The Ethernet address of the `NIC0` interface in DHCP configuration format
subnet1	The subnet connecting the `NIC0` interfaces

7. **Modify the DHCP configuration for `NIC1`:**

```
# pntadm -M NIC1IP-address -i newEthernet-address \
 -f 'PERMANENT+MANUAL' -m NIC1IP-address subnet2
```

The parameters of this command are as follows:

NIC1IP-address	The IP address of the `NIC1` interface
newEthernet-address	The Ethernet address of the `NIC1` interface in DHCP configuration format
subnet2	The subnet connecting the `NIC1` interfaces

8. **Refresh the DHCP configuration on the master node:**

```
# pkill -1 in.dhcpd
```

9. **Reboot the diskless node:**

```
ok> boot
```

10. **Verify that the node is configured correctly:**

```
# nhadm check
```

CHAPTER 13

Replacing a Disk

This chapter describes how to replace a disk on the vice-master node or a dataless node. To reduce the risk of loosing data, perform a backup before replacing a disk.

This chapter contains the following sections:

- "Replacing the Disk on the Vice-Master Node" on page 111
- "Replacing a Dataless Node Disk" on page 112

Replacing the Disk on the Vice-Master Node

This section describes how to replace the disk on the vice-master node.

▼ To Replace the Vice-Master Node Disk

1. **Verify that the new disk is the same hardware type and can have the same disk partition configuration as the old disk.**

2. **Replace the hardware by using the hardware documentation on** `http://www.sun.com/products-n-solutions/hardware/docs/`.

3. **Reformat the new disk by recreating the format of the old disk.**

   You must restore the local file system from backup.

4. **Restore the disk configuration.**

   a. **Install the Solaris operating system on the vice-master node.**

b. **Install the Foundation Services on the vice-master node.**

For information, see the *Netra High Availability Suite Foundation Services 2.1 6/03 Custom Installation Guide*.

5. **Power on the vice-master node.**

The master node detects that the vice-master node is not synchronized. A message is displayed in the system log file asking whether you want to restart the replication.

6. **Log in to the master node as superuser.**

7. **Accept a replication restart:**

```
# nhcrfsadm -a
```

8. **When the replication has resumed, resynchronize the vice-master node with the master node:**

```
# nhcrfsadm -f all
```

Do not perform a switchover before the vice-master node is synchronized with the master node.

9. **Verify that the synchronization is completed:**

```
# /usr/opt/SUNWesm/sbin/scmadm -S -M
```

While the synchronization is taking place, the `sync` label is displayed. When the synchronization is complete, the `sync` label is replaced by the replicating label.

10. **Verify that the node is configured correctly:**

```
# nhadm check
```

11. **Power on the dataless nodes or diskless nodes.**

Replacing a Dataless Node Disk

This section describes how to replace a dataless node disk.

▼ To Replace a Dataless Node Disk

1. **Verify that the new disk is of the same hardware type as the old disk.**

2. **Replace the hardware by referring to information in the hardware documentation on `http://www.sun.com/products-n-solutions/hardware/docs/`.**

3. **Reformat the new disk by recreating the format of the old disk.**

4. **Use the disk backup to restore the file system on the new disk.**

   For example, if the backup was created using the `ufsdump` command, use the `ufsrestore` command to restore the file system.

5. **Power on the dataless node.**

6. **Verify that the node is configured correctly:**

   ```
   # nhadm check
   ```

Index