



Netra™ High Availability Suite 3.0 1/08 Foundation Services Cluster Administration Guide

Sun Microsystems, Inc.
www.sun.com

Part No. 819-5235-13
March 2008, Revision A

Submit comments about this document at: <http://www.sun.com/hwdocs/feedback>

Copyright 2008 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents>, and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, docs.sun.com, Netra, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2008 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuels relatants à la technologie qui est décrit dans ce document. En particulier, et sans la limitation, ces droits de propriété intellectuels peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Java, docs.sun.com, Netra, et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciées de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

LA DOCUMENTATION EST FOURNIE "EN L'ÉTAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.



Adobe PostScript

Contents

Preface xiii

1. Using Cluster Administration Tools and Configuration Files 1

Using the Tools 1

Using Netra HA Suite Tools for Cluster Administration 2

Using Solaris Tools for Cluster Administration 5

Using Linux Tools for Cluster Administration 7

Using the Foundation Services Configuration Files in the Cluster Environment 8

Using Solaris Configuration Files in the Cluster Environment 10

Using Linux Configuration Files in the Cluster Environment 11

2. Accessing and Maintaining Log Files 15

Accessing System Log Files 15

▼ To Redirect Netra HA Suite Messages to a File on the Master Node 16

▼ To Define `loghost` in `/etc/hosts` 17

Maintaining System Log Files 18

▼ To Redirect System Log Messages to a Nonpeer Node 18

▼ To Remove Links to Crashed Daemons 20

Logging NFS Operations 20

▼ To Log NFS Operations 20

3. Determining Cluster Validity 23

Defining Minimum Criteria for a Cluster Running Highly Available Services 23

Verifying Services on Peer Nodes 24

- ▼ To Verify That the Cluster Has a Master Node and a Vice-Master Node 24
- ▼ To Verify That an nhcmmd Daemon Is Running on Each Peer Node 25
- ▼ To Verify That the Cluster Has a Redundant Ethernet Network 25
- ▼ To Verify That the Master Node and Vice-Master Node Are Synchronized 26
- ▼ To Verify That the Reliable Boot Service Is Running 28

Verifying That a Cluster Is Configured Correctly 28

- ▼ To Verify That a Cluster Is Configured Correctly 28

Reacting to a Failover 29

- ▼ To React to a Failover 29

4. Examining the Cluster Networking Configuration 31

Using Tools to Examine the Cluster Network 31

Verifying the Network Configuration 32

- ▼ To Verify the Network Interfaces of a Node 32

Creating a Network Topology Map 32

Mapping the Network Configuration of a Peer Node 35

- ▼ To Identify the Name of a Node 35
- ▼ To Identify the Host Name of the Network Interfaces of a Node 35
- ▼ To Obtain Configuration Information About the Network Interfaces of a Node 35

Mapping the External Network Configuration of a Node 38

- ▼ To Examine Interfaces Configured for External Communication 38

Mapping the Floating Address Triplet of the Master Node 39

- ▼ To Map the Floating Address Triplet of the Master Node 39

Mapping the Floating External Address of the Master Node 41

▼ To Map the Floating External Address of the Master Node	41
Examining the Network Configuration Files	48
Examining the Routes on a Node	50
▼ To Examine the Routing Table for a Node	50
Output of the <code>netstat -r</code> Command	52
Debugging Diskless Nodes With the <code>snoop</code> or <code>tcpdump</code> Tools	53
▼ To Examine a Diskless Node From the Master Node	53
5. Using and Managing System Files	55
Using Data Caches in Shared File Systems	55
▼ To Enable or Disable Data and Attribute Caching on Solaris	55
▼ To Enable or Disable Data and Attribute Caching on Linux	56
Changing the Location of the Scoreboard Bitmap	58
Storing Bitmap Scorecards on Disk or in Memory	58
▼ To Change the Location of the Scoreboard Bitmap	59
Managing Differences Between Files That Are Not Shared	60
▼ To Manage Differences Between Files That Are Not Shared	60
Using the Naming Services	61
6. Starting and Stopping Services, Nodes, and Clusters	63
Stopping and Restarting the Foundation Services	63
▼ To Start a Node Without the Foundation Services	64
▼ To Stop and Restart the Foundation Services Without Stopping the Solaris OS	64
▼ To Stop and Restart the Foundation Services Without Stopping Linux	65
▼ To Restart the Foundation Services	65
Stopping and Restarting Daemon Monitoring	66
▼ To Stop Daemon Monitoring	66
▼ To Restart Daemon Monitoring	67
Shutting Down and Restarting a Node	67

General Rules for Shutting Down a Node 67

- ▼ To Perform a Clean Reboot of a Solaris OS Node 68
- ▼ To Perform a Clean Reboot of a Linux Node 68
- ▼ To Perform a Clean Power off of a Solaris Node 68
- ▼ To Perform a Clean Power off of a Linux Node 69
- ▼ To Perform a Clean Halt of a Solaris Node 69
- ▼ To Perform a Clean Halt of a Linux Node 70
- ▼ To Abruptly Reboot a Solaris Node 70
- ▼ To Abruptly Reboot a Linux Node 70
- ▼ To Abruptly Power Off a Solaris Node 70
- ▼ To Abruptly Power Off a Linux Node 71
- ▼ To Abruptly Halt a Solaris Node 71
- ▼ To Abruptly Halt a Linux Node 71

Shutting Down a Node 71

- ▼ To Shut Down the Master Node 71
- ▼ To Shut Down the Vice-Master Node 72
- ▼ To Shut Down a Diskless Node or Dataless Node 72

Restarting a Node 72

- ▼ To Restart a Node 73

Shutting Down and Restarting a Cluster 73

- ▼ To Shut Down a Cluster 73
- ▼ To Restart a Cluster 74

Triggering a Switchover 76

- ▼ To Trigger a Switchover With `nhcmmstat` 76

Recovering an IP-Replicated Cluster 77

- ▼ To Recover a Solaris Cluster After Failure 77
- ▼ To Recover a Linux Cluster After Failure 78

7. Patching Software on a Solaris Cluster 81

About Patching Software on a Solaris Cluster	81
Choosing a Procedure to Patch Software on a Cluster	82
Overwriting Modified Netra HA Suite Files	83
Patching a Nonshared Package on Both Master-Eligible Nodes	83
▼ To Patch a Nonshared Package on Both Master-Eligible Nodes	83
Patching a Dataless Node	84
▼ To Patch a Dataless Node	84
Patching a Diskless Node	85
Patching the /usr Directory for a Diskless Node	85
▼ To Patch the /usr Directory for Diskless Nodes	85
Patching the root Directory for Diskless Nodes	86
▼ To Patch the root File System for Diskless Nodes	86
Patching a Shared Package	87
▼ To Patch a Shared Package on a Cluster	87
8. Patching Software on a Linux Cluster	89
About Patching Software on a Linux Cluster	89
▼ To Patch a Package on Both Master-Eligible Nodes	90
▼ To Patch a Dataless Node	90
9. Modifying and Adding Disk Partitions for Replicated Data on the Solaris OS	93
Using the <code>format</code> Utility to Display and Modify the Configuration of a Disk Partition on a Solaris Node	94
▼ To Display the Configuration of a Disk Partition on a Solaris Node	94
▼ To Change the Size of a Disk Partition on a Master-Eligible Node on the Solaris OS	97
Increasing the Size of a Replicated Data Partition on a Physical Disk on a Solaris Node	99
▼ To Increase the Size of a Replicated Data Partition on a Physical Disk on a Solaris Node	99

Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node 102

- ▼ To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node 102

Adding a Mirrored Data Partition to a Virtual Disk on a Solaris Node 104

- ▼ To Add a Mirrored Data Partition to a Virtual Disk on a Solaris Node 105

10. Modifying and Adding Disk Partitions for Replicated Data on Linux 107

Using the `parted` Utility to Display and Modify the Configuration of a Disk Partition on a Linux Node 108

- ▼ To Display the Configuration of a Disk Partition on a Linux Node 108
- ▼ To Change the Size of a Disk Partition on a Master-Eligible Node on Linux 109

Increasing the Size of a Replicated Data Partition on a Physical Disk on a Linux Node 110

- ▼ To Increase the Size of a Replicated Data Partition on a Physical Disk on a Linux Node 110

Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node 111

- ▼ To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node 112

Adding a Mirrored Data Partition to a Virtual Disk on a Linux Node 114

- ▼ To Add a Mirrored Data Partition to a Virtual Disk on a Linux Node 114

11. Adding a Node to a Cluster 117

Deciding How to Add a Node to a Cluster 117

Determining Whether a Cluster Has Predefined Nodes 118

- ▼ To Determine Whether the Cluster Has Predefined Nodes 118

12. Replacing Hardware in a Cluster 119

Preparing to Replace Hardware in a Cluster 119

Replacing a CPU Board on a Node 121

- ▼ To Replace a Board on a Node 121

Replacing a CPU Board on a Diskless Node 122

- ▼ To Replace a Diskless Node With the DHCP Static Boot Policy 122

Replacing Ethernet Cards on a Vice-Master or Dataless Node 124

- ▼ To Replace Ethernet Cards on the Vice-Master Node or a Dataless Node 124

Replacing Ethernet Cards on a Diskless Node 125

- ▼ To Replace Ethernet Cards on a Diskless Node With the DHCP Static Boot Policy 125

Replacing the Disk on the Vice-Master Node 127

- ▼ To Replace the Disk on the Vice-Master Node Using IP Replication 127
- ▼ To Replace the Disk on the Vice-Master Node Using Shared Disk 129

Replacing Disks on Both Master-Eligible Nodes Without Cluster Shutdown 130

- ▼ To Replace Disks on Both Master-Eligible Nodes Using IP Replication Without Full Cluster Shutdown 130

Replacing a Dataless Node Disk 132

- ▼ To Replace a Dataless Node Disk 132

Index 133

Tables

TABLE 1-1	Netra HA Suite Cluster Administration Tools	3
TABLE 1-2	Solaris Tools Used for Cluster Administration	5
TABLE 1-3	Linux Tools Used for Cluster Administration	7
TABLE 1-4	Netra HA Suite Files Created During Initial Cluster Configuration	9
TABLE 1-5	Solaris Files Used With the Foundation Services	10
TABLE 1-6	Linux Files Used With the Foundation Services	12
TABLE 4-1	Description of the Output of the <code>netstat -r</code> Command	52
TABLE 12-1	Reference for Replacing Hardware	120

Preface

This book describes how to maintain a cluster running the Netra™ High Availability (HA) Suite 3.0 1/08 Foundation Services.

Who Should Use This Book

This book is for system administrators who are maintaining a cluster running Netra HA Suite software. This book helps administrators perform 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 working with a cluster environment. Knowledge of networking and system administration of the operating system is also required.

How This Book Is Organized

This book contains the following chapters:

- [Chapter 1](#) describes how to use the Netra HA Suite tools and configuration files. It also describes the operating system tools and configuration files that you can use.
- [Chapter 2](#) explains how to configure and maintain your system log files.
- [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.
- [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 on the Solaris™ Operating System (Solairs OS).
- [Chapter 8](#) describes how to install a patch for the Foundation Services on Linux.
- [Chapter 9](#) describes how to resize a disk partition, how to add a new disk partition, and how to share a disk partition on the Solaris OS.
- [Chapter 10](#) describes how to resize a disk partition, how to add a new disk partition, and how to share a disk partition on Linux.
- [Chapter 11](#) describes the options for adding diskless or dataless nodes to a cluster.
- [Chapter 12](#) explains how to replace a CPU board, Ethernet card, or disk in a cluster.

Using UNIX Commands

This document might not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices. Refer to the following for this information:

- Software documentation that you received with your system
- Solaris Operating System documentation, which is at:

Shell Prompts

Shell	Prompt
C shell	<i>machine-name%</i>
C shell superuser	<i>machine-name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

Typographic Conventions

Typeface*	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; onscreen computer output	Edit your <i>.login</i> file. Use <i>ls -a</i> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with onscreen computer output	% su Password:
<i>AaBbCc123</i>	Book titles, new words or terms, and words to be emphasized. Replace command-line variables with real names or values.	Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. To delete a file, type rm <i>filename</i> .

* The settings on your browser might differ from these settings.

Related Documentation

The following table lists the documentation for this product. The online documentation is available at:

<http://docs.sun.com/app/docs/prod/netra.ha30>

Application	Title	Part Number
Late-breaking news	<i>Netra High Availability Suite 3.0 1/08 Release Notes</i>	819-5249-14
Introduction to concepts	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Overview</i>	819-5240-13
Basic setup, supported hardware, and configurations	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Getting Started Guide</i>	819-5241-13
Automated installation methods	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Installation Guide</i>	819-5242-13
Detailed installation methods	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS</i>	819-5237-13
Cluster administration	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Cluster Administration Guide</i>	819-5235-13
Using the Cluster Membership Manager	<i>Netra High Availability Suite 3.0 1/08 Foundation Services CMM Programming Guide</i>	819-5236-13
Using the SAF CMM API	<i>Netra High Availability Suite 3.0 1/08 Foundation Services SA Forum Programming Guide</i>	819-5246-13
Using the Node Management Agent	<i>Netra High Availability Suite 3.0 1/08 Foundation Services NMA Programming Guide</i>	819-5239-13
Configuring outside the cluster using CGTP	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Standalone CGTP Guide</i>	819-5247-13
Man pages for Foundation Services features and APIs using the Solaris OS	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Solaris Reference Manual</i>	819-5244-13
Man pages for Foundation Services features and APIs using Linux	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Linux Reference Manual</i>	819-5245-12
Definitions and acronyms	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Glossary</i>	819-5238-13
Common problems	<i>Netra High Availability Suite 3.0 1/08 Foundation Services Troubleshooting Guide</i>	819-5248-13

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (<http://www.sun.com/documentation>)
- Support (<http://www.sun.com/support>)
- Training (<http://www.sun.com/training>)

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

<http://www.sun.com/hwdocs/feedback>

Please include the title and part number of your document with your feedback:

Netra™ High Availability Suite 3.0 1/08 Foundation Services Cluster Administration Guide, part number 819-5235-13

Using Cluster Administration Tools and Configuration Files

The Netra High Availability (HA) Suite 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, refer to the *Netra High Availability Suite 3.0 1/08 Foundation Services Overview*.

This chapter contains the following sections:

- [“Using the Tools” on page 1](#)
- [“Using Netra HA Suite Tools for Cluster Administration” on page 2](#)
- [“Using Solaris Tools for Cluster Administration” on page 5](#)
- [“Using Linux Tools for Cluster Administration” on page 7](#)
- [“Using the Foundation Services Configuration Files in the Cluster Environment” on page 8](#)
- [“Using Solaris Configuration Files in the Cluster Environment” on page 10](#)
- [“Using Linux Configuration Files in the Cluster Environment” on page 11](#)

Using the Tools

The Foundation Services tools are located in `/opt/SUNWcgha/sbin` on Solaris systems, and in `/opt/sun/sbin` on Linux systems. Include the appropriate directory in your `PATH` environment variable.

To use some of the Netra HA Suite 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, refer to the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

Using Netra HA Suite Tools for Cluster Administration

[TABLE 1-1](#) 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, refer to the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

TABLE 1-1 Netra HA Suite Cluster Administration Tools

Tool	Description
nhadm	<ul style="list-style-type: none"> • Verify the cluster configuration: <code>nhadm check</code> <code>nhadm check installation</code> <code>nhadm check configuration</code> <code>nhadm check starting</code> • Foundation Services post-installation and OS configuration (Linux only) <code>nhadm configure</code> • Create the file required for patching shared packages: <code>nhadm confshare</code> • Display information about node configuration: <code>nhadm display</code> • Copy files from the master node to the vice-master node. Files listed can be passed as an argument or listed in the data-file file. <code>nhadm [d data-file] copy [file]</code> • List differences between specified nonreplicated files on the master node and vice-master node: <code>nhadm synccheck</code> • Store differences between specified nonreplicated files on the master node and vice-master node: <code>nhadm syncgen</code> • List all possible error messages returned by the <code>nhadm</code> command: <code>nhadm -z</code> • Display current help options: <code>nhadm -h</code> • Provide detailed debugging information: <code>nhadm -v</code> • Display only error messages: <code>nhadm -s</code> <p>For more information, see the <code>nhamd1M</code> man page on the Solaris OS or the <code>nhamd8</code> man page on Linux.</p>
nhcmmqualif	<p>Qualify a node as the master node if the node is master-eligible, and if the cluster has no master node.</p> <p>For more information, see the <code>nhcmmqualif1M</code> man page on the Solaris OS or the <code>nhcmmqualif8</code> man page on Linux.</p>

TABLE 1-1 Netra HA Suite Cluster Administration Tools (*Continued*)

Tool	Description
nhcmmrole	<ul style="list-style-type: none"> • Get the role of the node. A node can have the following roles: MASTER VICEMASTER IN_CLUSTER OUT_OF_CLUSTER • List the role of the current node: nhcmmrole -v • Determine the time out period in seconds > 0: nhcmmrole -t <p>For more information about roles, see the <i>Netra High Availability Suite 3.0 1/08 Foundation Services Glossary</i> and the nhcmmrole1M man page on the Solaris OS or the nhcmmrole8 man page on Linux.</p>
nhcmmstat	<ul style="list-style-type: none"> • Analyze the state of a node: info local master mynode potential vice • Analyze the state of a group of nodes: all count • Force a reload of the cluster_nodes_table configuration: reload • Trigger a switchover: so • Trigger a switchover, from a master node only, switching to a view-master role: * so • Qualify a node: squalif • Exit: exit quit • Display help information: help <p>For more information, see the nhcmmstat1M man page on the Solaris OS or the nhcmmstat8 man page on Linux.</p>

TABLE 1-1 Netra HA Suite Cluster Administration Tools (*Continued*)

Tool	Description
nhcrfsadm	Authorize or refuse permission to start replication of the master node to the vice-master node. For more information, see the <code>nhcrfsadm1M</code> man page on the Solaris OS or the <code>nhcrfsadm8</code> man page on Linux.
nhenablesync	Trigger disk synchronization. The <code>nhenablesync</code> tool has no effect when it is used in a shared disk configuration. For more information, see the <code>nhenablesync1M</code> man page on the Solaris OS or the <code>nhenablesync8</code> man page on Linux.
nhpmdadm	Administer the Daemon Monitor. <ul style="list-style-type: none">• List all available tags: <code>nhpmdadm -L</code>• List the individual Netra HA Suite status: <code>nhpmdadm -l <name></code> For more information about the Daemon Monitor, see the <code>nhpmd1M</code> man page on the Solaris OS or the <code>nhpmd1M</code> man page on Linux.

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, refer to the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

TABLE 1-2 Solaris Tools Used for Cluster Administration

Tool	Description
boot1M	Boot a specific node. For an example of using the <code>boot</code> command, see “To Restart a Cluster” on page 74 .
format1M	Format, label, repair, and analyze disks on your system. For more information about using the <code>format</code> utility on a cluster, see “Using the format Utility to Display and Modify the Configuration of a Disk Partition on a Solaris Node” on page 94 .
ifconfig1M	Check and set a network interface configuration, as described in “To Obtain Configuration Information About the Network Interfaces of a Node” on page 35 .

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

Tool	Description
init1M	Create processes from information stored in the <code>/etc/inittab</code> file. Do not use this command to reboot or shut down a node. It can be used only to stop and restart Netra HA Suite by reaching the single-user state and then going back to the normal state. To reboot or shut down a node running Netra HA Suite, see “General Rules for Shutting Down a Node” on page 67.
mount1M	Mount a file system. For an example of using the <code>mount</code> command, see “To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node” on page 102.
netstat1M	Show the network status as described in “Examining the Routes on a Node” on page 50. Useful options include <code>netstat -rn</code> for details about routing tables.
patchadd1M	Add a patch, as described in Chapter 7 .
pgrep1	Find processes by name and other attributes. For an example of using the <code>pgrep</code> command, see “To Verify That an <code>nhcmmmd</code> Daemon Is Running on Each Peer Node” on page 25.
pkill1M	Kill processes. For an example of using the <code>pkill</code> command, see “To Replace Ethernet Cards on a Diskless Node With the DHCP Static Boot Policy” on page 125.
ps1	List processes. For a list of monitored daemons, see the <code>nhpmd1M</code> man page.
reboot1M	Do not use this command to reboot a node running the Foundation Services. Instead, refer to “General Rules for Shutting Down a Node” on page 67 for information about rebooting a node that is running the Foundation Services.
route1M	Add or remove a route, as described in “Examining the Routes on a Node” on page 50.
truss1	Trace system calls and signals.
ufsdump1M	Back 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 Replicated Data Partition on a Virtual Disk on a Solaris Node” on page 102.
ufsrestore1M	Restore files from backup media created with the <code>ufsdump</code> command. For an example of using this command, see “To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node” on page 102.

Using Linux Tools for Cluster Administration

The following table lists Linux tools that you can use for cluster administration. The table describes the purpose of the tools and provides links to their man pages.

TABLE 1-3 Linux Tools Used for Cluster Administration

Tool	Description
parted 8	A partition table manipulator for Linux. For more information about using the parted utility on a cluster, see “Using the parted Utility to Display and Modify the Configuration of a Disk Partition on a Linux Node” on page 108.
ifconfig8	Check and set a network interface configuration, as described in “To Obtain Configuration Information About the Network Interfaces of a Node” on page 35.
init8	Create processes from information stored in the /etc/inittab file. Do not use this command to reboot or shutdown a node. It can be used only to stop and restart the Foundation Services by reaching the single-user state and then going back to the normal state. To reboot or shutdown a node running Netra HA Suite, see “General Rules for Shutting Down a Node” on page 67.
mount8	Mount a file system. For an example of using the mount command, see “Increasing the Size of a Replicated Data Partition on a Physical Disk on a Linux Node” on page 110.
netstat8	Show the network status as described in “Examining the Routes on a Node” on page 50. Useful options include netstat -rn for details about routing tables.
rpm8	RPM Package Manager, used to remove, install, or upgrade a package as described in Chapter 7 .
pgrep1	Find processes by name and other attributes. For an example of using the pgrep command, see “To Verify That an nhcmmnd Daemon Is Running on Each Peer Node” on page 25.
pkill11	Kill processes. For a list of monitored daemons, see the nhpmd1 man page.
reboot8	Reboot a node running the Foundation Services only with the -f option: do not force halt or reboot using shutdown8. Refer to “General Rules for Shutting Down a Node” on page 67 for information about rebooting a node that is running the Foundation Services.
route8	Add or remove a route, as described in “Examining the Routes on a Node” on page 50.

TABLE 1-3 Linux Tools Used for Cluster Administration *(Continued)*

Tool	Description
strace1	Trace system calls and signals. For information about using this command, see “Stopping and Restarting Daemon Monitoring” on page 66
dump8	Back 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 Replicated Data Partition on a Virtual Disk on a Linux Node” on page 112.
restore8	Restore files from backups created with the dump command. For an example of using this command, see “To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node” on page 112.



Using the 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 on Solaris systems, and `/etc/opt/sun/nhas/` on Linux systems. The following table lists the files created during initial cluster configuration.

Note – NMA is not delivered on Linux. NMA configuration files are not needed for Linux systems.

TABLE 1-4 Netra HA Suite Files Created During Initial Cluster Configuration

Configuration File	Description
<code>cluster_nodes_table</code>	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 <code>cluster_nodes_table4</code> man page on the Solaris OS or the <code>cluster_nodes_table5</code> man page on Linux.
<code>nhfs.conf</code>	Contains configuration information for the individual Netra HA Suite. There is a copy of this file on each peer node. For information, see the <code>nhfs.conf4</code> man page on the Solaris OS or the <code>nhfs.conf5</code> man page on Linux.
<code>nma.notifs.txt</code>	Defines types and targets for the trap notifications of the Node Management Agent (NMA) SNMP. For information, see the <code>nma.notifs.txt4</code> man page on the Solaris OS or the <code>nma.notifs.txt5</code> man page on Linux.
<code>nma.params.txt</code>	Defines the communication, implementation and accessibility configuration of SNMP in the NMA. For information, see the <code>nma.params.txt4</code> man page on the Solaris OS or the <code>nma.params.txt5</code> man page on Linux.
<code>nma.properties</code>	Lists the properties of the NMA on a node. There is a copy of this file on each peer node. For information, see the <code>nma.properties4</code> man page on the Solaris OS or the <code>nma.properties5</code> man page on Linux.
<code>nma.targets.txt</code>	Defines the targets to which the NMA sends SNMP traps. For information, see the <code>nma.targets.txt4</code> man page on the Solaris OS or the <code>nma.targets.txt5</code> man page on Linux.
<code>nma.security</code>	Configures SNMP security for the NMA. For information, see the <code>nma.security4</code> man page on the Solaris OS or the <code>nma.security5</code> man page on Linux.

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 you can edit during cluster configuration.

TABLE 1-5 Solaris Files Used With the Foundation Services

Configuration File	Description
<code>/etc/bootparams</code>	A database of boot parameters. For information, see <code>bootparams4</code> .
<code>/etc/coreadm.conf</code>	Parameters for system core file configuration.
<code>/etc/dfs/dfstab</code>	Commands for sharing resources across a cluster. This file does not contain commands for partitions mirrored by Reliable NFS. For information, see <code>dfstab4</code> .
<code>/etc/hostname.interface-name</code>	Configuration of each network interface on a node. The file contains the host name of the interface specified by <i>interface-name</i> . If you create a network interface, you must create a <code>/etc/hostname.interface-name</code> file for that interface. For information about network interface configuration, see “Examining the Network Configuration Files” on page 48 .
<code>/etc/hosts</code>	A local database of host names. For information, see the <code>hosts4</code> man page. For information about network configuration files, see “Examining the Network Configuration Files” on page 48 .
<code>/etc/inet/inetd.conf</code>	The list of servers that <code>inetd</code> invokes when it receives an Internet request over a socket. For information, see the <code>inetd.conf4</code> man page.
<code>/etc/inittab</code>	The process dispatching control by the <code>init</code> command. This file contains the <code>rc2</code> and <code>rc3</code> entries. For information, see the <code>inittab4</code> man page. For information about the Daemon Monitor, see the <code>nhpmd1M</code> man page. For the Solaris 10 OS and later, see the <code>smf5</code> man page.
<code>/etc/netmasks</code>	A list of the network masks used to implement IP subnetting and their associated IP network numbers. For information, see the <code>netmasks4</code> man page. For information about network configuration, see Chapter 4 .
<code>/etc/nodename</code>	The local source for the system name. For information, see the <code>nodename4</code> man page. For information about network configuration files, see “Examining the Network Configuration Files” on page 48 .

TABLE 1-5 Solaris Files Used With the Foundation Services (*Continued*)

Configuration File	Description
<code>/etc/notrouter</code>	A file to specify that a node does not act as a router. Use this file to protect against route mismatch. This file must be configured on each peer node to ensure that the node is not routable.
<code>/etc/nsswitch.conf</code>	A configuration file for the name service that provides the sources of database information and their lookup order. For information, see the <code>nsswitch.conf4</code> man page. For information about using the naming services, see “Using the Naming Services” on page 61 .
<code>/etc/services</code>	A list of the services and aliases available through the Internet. The <code>/etc/services</code> file can be changed to modify the Netra HA Suite ports if, for example, an application is already using the ports configured by default at startup. For information, see the <code>services4</code> man page. For information about prioritizing the Foundation Services, see “Using the Naming Services” on page 61 .
<code>/etc/syslog.conf</code>	A file used by the system log daemon, <code>syslogd</code> , to forward a system message to the appropriate log files and users. For information, see the <code>syslog.conf4</code> man page. For information about configuring system log files, see Chapter 2 .
<code>/etc/system</code>	A file that customizes the operation of the operating system kernel. You should not have to change the <code>/etc/system</code> file. If you do change the <code>/etc/system</code> file, you must reboot the node for the changes to take effect.
<code>/etc/vfstab</code>	A table of file system defaults. For information, see <code>vfstab4</code> .
<code>.profile</code>	A file to set up an environment for a user at login time. For information about how to use this file, see the <code>profile4</code> man page.
<code>.rhosts</code>	A list of trusted hosts and users. The <code>.rhosts</code> file specifies remote trusted hosts and users. Modify this file to change remote access authorization between peer nodes. For information, see <code>rhosts4</code> .

Using Linux Configuration Files in the Cluster Environment

This section describes the Linux configuration files that can help you determine the current configuration of the cluster. For information about how and when to modify a Linux configuration file, consult the man page for the file and the Linux documentation set.

The following table lists the Linux configuration files that you can edit during cluster configuration.

TABLE 1-6 Linux Files Used With the Foundation Services

Configuration File	Description
/etc/exports	An access control list for NFS file systems being exported. This file does not contain the access control list for partitions mirrored by Reliable NFS. For information, see the <code>exports5</code> man page.
/etc/fstab	A table of file system defaults. For information, see <code>fstab5</code> .
/etc/hosts	A local database of host names. For information, see the <code>hosts5</code> man page. For information about network configuration files, see “Examining the Network Configuration Files” on page 48 .
/etc/inetd.conf	The list of servers that <code>inetd</code> invokes when it receives an Internet request over a socket. For information, see the <code>inetd.conf5</code> man page.
/etc/inittab	The process dispatching control by the <code>init</code> command. This file contains the <code>rc2</code> and <code>rc3</code> entries. For information, see the <code>inittab5</code> 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 <code>netmasks4</code> man page. For information about network configuration, see Chapter 4 .
/etc/network/interface	A configuration of each network interface on a node on MontaVista Linux. For information, see the <code>interface5</code> man page. For information about network interface configuration see “Examining the Network Configuration Files” on page 48 .
/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 <code>nsswitch.conf5</code> man page. For information about using the naming services, see “Using the Naming Services” on page 61 .
/etc/services	A list of the services and aliases available through the Internet. The <code>/etc/services</code> 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 <code>services5</code> man page. For information about prioritizing the Foundation Services, see “Using the Naming Services” on page 61 .
/etc/sysconfig/network-scripts/ifcfg-*	A configuration of a network interface on Wind River CGL (for example, <code>/etc/sysconfig/network-scripts/ifcfg-eth0</code>)

TABLE 1-6 *(Continued)*Linux Files Used With the Foundation Services *(Continued)*

Configuration File	Description
<code>/etc/syslog.conf</code>	A file used by the system log daemon, <code>syslogd</code> , to forward a system message to the appropriate log files and users. For information, see the <code>syslog.conf5</code> man page. For information about configuring system log files, see Chapter 2 .
<code>/etc/xinetd.conf</code>	The list of servers that <code>inetd</code> invokes when it receives an Internet request over a socket. For information, see the <code>xinetd.conf5</code> man page.
<code>.rhosts</code>	A list of trusted hosts and users. The <code>.rhosts</code> file specifies remote trusted hosts and users. Modify this file to change remote access authorization between peer nodes.

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. This makes the information in logs more accessible and prevents 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.conf4` and `syslogd1M` man pages.

For information about using log files generated by the Linux Operating System and the Foundation Services, consult the `syslog.conf5` and `syslogd8` man pages.

This chapter contains the following sections:

- “Accessing System Log Files” on page 15
- “Maintaining System Log Files” on page 18
- “Logging NFS Operations” on page 20

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) on Solaris systems (NMA is not provided for Linux systems). 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.conf4` man page on the Solaris OS or the `syslog.conf5` man page on Linux. 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. Netra HA Suite software has the following error message categories:

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

For information about error message categories, see the `syslog.conf4` man page.

▼ To Redirect Netra HA Suite 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 /var/log/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/log/logfile
```

6. Restart the syslogd daemon:

For versions of the Solaris OS earlier than version 10:

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

For the Solaris 10 OS and later:

```
# svcadm restart svc:/system/system-log
```

For the Linux OS:

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

▼ To Define loghost in /etc/hosts

This procedure configures messages to be sent to a specific node. Use this procedure to prevent disruption of Network File System (NFS) traffic by high volumes of messages from diskless nodes.

Note – Netra HA Suite does not support diskless Linux nodes.

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 cgtcp0 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 cgtcp0 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. Restart the syslogd daemon:

For versions of the Solaris OS earlier than version 10:

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

For the Solaris 10 OS and later:

```
# svcadm restart svc:/system/system-log
```

For the Linux OS:

```
# /etc/rc.d/init.d/syslog stop
# /etc/rc.d/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 18](#).

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. This prevents 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. Restart the syslogd daemon:

For versions of the Solaris OS earlier than version 10:

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

For the Solaris 10 OS and later:

```
# svcadm restart svc:/system/system-log
```

For the Linux OS:

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

Note – Logging NFS operations is not supported on Linux.

▼ 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 the following line:**

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

where:

<i>my-tag</i>	is the label to identify NFS log messages
<i>my-dir</i>	is the directory that contains the log of the NFS operations

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

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

where:

my-tag is the label to identify NFS log messages

shared-nfs is the shared directory to which all NFS operations, such as `rm` and `mkdir`, are logged.

4. Restart the NFS server:

For versions of the Solaris OS earlier than version 10:

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

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

```
# share
```

You will see output similar to the following 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, you will see output similar to the following:

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 as superuser to a peer node other than the master node.

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](#).

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” on page 23](#)
 - [“Verifying Services on Peer Nodes” on page 24](#)
 - [“Verifying That a Cluster Is Configured Correctly” on page 28](#)
 - [“Reacting to a Failover” on page 29](#)
-

Defining Minimum Criteria for a Cluster Running Highly Available Services

A Netra HA Suite cluster can run the following highly available services: Reliable NFS and the Reliable Boot Service (RBS). For information about highly available services, see the *Netra High Availability Suite 3.0 1/08 Foundation Services 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 24](#).

- An `nhcmmmd` daemon on each peer node. The `nhcmmmd` daemon on the master node manages the membership of the other peer nodes. The `nhcmmmd` daemon on other peer nodes receives cluster information from the `nhcmmmd` daemon on the master node. To verify that there is an `nhcmmmd` daemon on each peer node, perform the procedure described in [“To Verify That an `nhcmmmd` Daemon Is Running on Each Peer Node” on page 25](#).
- 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 25](#).
- 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 26](#).

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 the following command:

```
# 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 `nhcmmstat1M` man page.

- If there is a master node but no vice-master node, reboot the second master-eligible node as described in [“To Perform a Clean Reboot of a Linux Node” on page 68](#).

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 3.0 1/08 Foundation Services 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.conf4` and `cluster_nodes_table4` 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 3.0 1/08 Foundation Services Troubleshooting Guide*.

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

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

```
# pgrep -x nhcmmnd
```

- 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 3.0 1/08 Foundation Services 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 the following message:

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

The `nhcmmnd` daemon creates this message 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

This procedure only applies to systems using IP replication, rather than shared disk.

1. Log in to a master node as superuser.

2. Test whether the vice-master node is synchronized with the master node:

For versions earlier than the Solaris 10 OS:

```
# /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.

```
# /usr/sbin/dsstat 1
```

For the Solaris 10 OS and later:

- If the `dsstat` command indicates "R" in the "S" column, the vice-master node is synchronized with the master node.
- If the `dsstat` command indicates "L" in the "S" column, the vice-master node is not synchronized and no synchronization is currently taking place.

- If the `dsstat` command indicates "SY" in the "S" column, the vice-master node is not synchronized and synchronization is currently taking place.

```
# drbdadm cstate all
```

Note – Refer to the `dsstat1M` man page for more information.

For the Linux OS:

- If the `drbdadm` command indicates "Connected", the vice-master node is synchronized with the master node.
- If the `drbdadm` command indicates "StandAlone" or "WfConnection", the vice-master node is not synchronized and no synchronization is currently taking place.
- If the `drbdadm` command indicates "SyncSource", the vice-master node is not synchronized and synchronization is currently taking place.

Note – Refer to the `drbdadm8` man page for more information.

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 `nhenablesync`, see the `nhenablesync1M` man page.

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 3.0 1/08 Foundation Services Troubleshooting Guide*.

For more information about the `scmadm` command, see the `scmadm1M` man page. For more information about the *RNFS.EnableSync* parameter, see the `nhfs.conf4` man page.

▼ To Verify That the Reliable Boot Service Is Running

Diskless nodes and the Reliable Boot Service can be used on the Solaris OS, but are not supported on Linux.

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 3.0 1/08 Foundation Services 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 23](#). 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 their 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 text
```

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

For more information, see the `nhadm1M` 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 3.0 1/08 Foundation Services 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 23.**

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

```
# nhadm check
```


Examining the Cluster Networking Configuration

For 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 3.0 1/08 Foundation Services Overview*.

- [“Using Tools to Examine the Cluster Network” on page 31](#)
- [“Creating a Network Topology Map” on page 32](#)
- [“Mapping the Network Configuration of a Peer Node” on page 35](#)
- [“Mapping the External Network Configuration of a Node” on page 38](#)
- [“Mapping the Floating Address Triplet of the Master Node” on page 39](#)
- [“Mapping the Floating External Address of the Master Node” on page 41](#)
- [“Examining the Network Configuration Files” on page 48](#)
- [“Examining the Routes on a Node” on page 50](#)

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 3.0 1/08 Foundation Services 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 network interfaces are configured. For example if the current node is running the Solaris OS, the command checks that `/etc/hostname.NIC0` and `/etc/hostname.NIC1` exist. If the current node is running MontaVista Linux, the command checks that `NIC0` and `NIC1` are configured in the file `/etc/network/interfaces`
- Whether the `/etc/hosts` file contains IP addresses that correspond to the `NIC0`, `NIC1`, and `cgtp0` interfaces for each peer node
- If the current node is running Wind River CGL, the command checks that the following directories exist:

```
/etc/sysconfig/network-scripts/ifcfg-NIC0
```

```
/etc/sysconfig/network-scripts/ifcfg-NIC1
```

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

One set per peer node master or otherwise	Additional addresses for master-eligible nodes
<p>Address triplet</p> <p>NIC0 interface name IP address Ethernet address netmask host name</p> <p>NIC1 interface name IP address Ethernet address netmask host name</p> <p>cgtp0 interface name IP address Ethernet address netmask host name</p> <p>Additional interface (optional)</p> <p>NIC2 interface name IP address Ethernet address netmask host name</p>	<p>Floating address triplet</p> <p>NIC0:1 interface name IP address netmask host name</p> <p>NIC1:1 interface name IP address netmask host name</p> <p>cgtp0:1 interface name IP address netmask host name</p> <p>External floating address (optional)</p> <p>NIC2 interface name IP address netmask host name group name</p> <p>NIC3 interface name IP address netmask host name group name</p> <p>NIC2:1 interface name IP address netmask host name</p>

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 50](#).

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` as follows:**

```
# 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` as follows:**

```
# netstat -i
```

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

For more information, see the `netstat1M` 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 running the Solaris OS and is using the class B addressing scheme, output similar to the following is displayed:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
  inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
  inet 172.17.0.25 netmask ffff0000 broadcast 172.17.255.255
  ether 0:0:0:0:0:0
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 172.15.0.25 netmask ffff0000 broadcast 172.15.255.255
  ether 8:0:20:f9:a5:56
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
  inet 172.16.0.25 netmask ffff0000 broadcast 172.16.255.255
  ether 8:0:20:f9:a5:57
```

In this example, the *nodeid* is 25, the netmask is ffff0000, and the *subnetworks* 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 the following is displayed:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
  inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
  inet 10.25.3.25 netmask fffffff0 broadcast 10.25.3.255
  ether 0:0:0:0:0:0
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 10.25.1.25 netmask fffffff0 broadcast 10.25.1.255
```

```
ether 8:0:20:f9:a5:56
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
inet 10.25.2.25 netmask ffffffff00 broadcast 10.25.2.255
ether 8:0:20:f9:a5:57
```

In this example, the *nodeid* is 25, the netmask is ffffffff00, and the *subnetworks* are 10.25.1.0, 10.25.2.0, and 10.25.3.0.

- If the node is running Linux and is using the default class C addressing scheme, output similar to the following is displayed:

```
cgtp0      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
  inet addr:10.125.3.20  Bcast:10.125.3.255  Mask:255.255.255.0
  inet6 addr: fe80::200:ff:fe00:0/64 Scope:Link
  UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
  TX packets:80209 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:0 (0.0 b)  TX bytes:9631182 (9.1 MiB)

eth0       Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:58
  inet addr:10.125.1.20  Bcast:10.125.1.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7b58/64 Scope:Link
  UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
  RX packets:64908 errors:0 dropped:0 overruns:0 frame:0
  TX packets:56376 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:47373431 (45.1 MiB)  TX bytes:6795936 (6.4 MiB)
  Interrupt:32

eth1       Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:59
  inet addr:10.125.2.20  Bcast:10.125.2.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7b59/64 Scope:Link
  UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
  RX packets:25575 errors:0 dropped:0 overruns:0 frame:0
  TX packets:23833 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:18522762 (17.6 MiB)  TX bytes:2835246 (2.7 MiB)
  Interrupt:33

lo         Link encap:Local Loopback
  inet addr:127.0.0.1  Mask:255.0.0.0
  inet6 addr: ::1/128 Scope:Host
  UP LOOPBACK RUNNING  MTU:16436  Metric:1
  RX packets:20877 errors:0 dropped:0 overruns:0 frame:0
  TX packets:20877 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:3042793 (2.9 MiB)  TX bytes:3042793 (2.9 MiB)
```

In this example, the *nodeid* is 20, the netmask is ffffffff00, and the *subnetworks* are 10.125.1.0, 10.125.2.0, and 10.125.3.0.

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 3.0 1/08 Foundation Services 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 running the Solaris OS and using the class C addressing scheme, and configured for external communication through additional logical interfaces.

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
  inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
  inet 10.25.3.25 netmask ffffffff0 broadcast 10.25.3.255
  ether 0:0:0:0:0:0
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 10.25.1.25 netmask ffffffff0 broadcast 10.25.1.255
  ether 8:0:20:f9:a5:56
hme0:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 10.17.1.125 netmask ffffffff0 broadcast 10.17.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
  inet 10.25.2.25 netmask ffffffff0 broadcast 10.25.2.255
  ether 8:0:20:f9:a5:57
```


3. Search in the output of [Step 2](#) for an interface to an external network.

In this example, the logical interface `hme0:1` 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 3.0 1/08 Foundation Services Overview*.

▼ To Map the Floating Address Triplet of the Master Node

1. Log in to a peer node.
2. Identify the master node:

```
# nhcmmstat -c 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
```

The following table shows sample output for a Solaris OS node using the class C addressing scheme:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
    inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
    inet 10.25.3.25 netmask ffffffff00 broadcast 10.25.3.255
    ether 0:0:0:0:0:0
cgtp0:1:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 2
    inet 10.25.3.1 netmask ffffffff00 broadcast 10.25.3.255
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 10.25.1.25 netmask ffffffff00 broadcast 10.25.1.255
    ether 8:0:20:f9:a5:56
hme0:2:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 3
    inet 10.25.1.1 netmask ffffffff00 broadcast 10.25.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
    inet 10.25.2.25 netmask ffffffff00 broadcast 10.25.2.255
    ether 8:0:20:f9:a5:57
hme1:1:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 4
    inet 10.25.2.1 netmask ffffffff00 broadcast 10.25.2.255
```

The floating address triplet has the logical interfaces hme0:2, 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.

The following table shows sample output:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
    inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
    inet 10.25.3.26 netmask ffffffff00 broadcast 10.25.3.255
    ether 0:0:0:0:0:0
cgtp0:1:
flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 2
    inet 10.25.3.1 netmask ffffffff00 broadcast 10.25.3.255
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 10.25.1.26 netmask ffffffff00 broadcast 10.25.1.255
    ether 8:0:20:fa:3f:70
```

```
hme0:2:
flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 3
inet 10.25.1.1 netmask ffffffff00 broadcast 10.25.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
inet 10.25.2.26 netmask ffffffff00 broadcast 10.25.2.255
ether 8:0:20:fa:3f:71
hme1:1:
flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 4
inet 10.25.2.1 netmask ffffffff00 broadcast 10.25.2.255
```

Note – Note that the hme0:2, hme1:1, and cgtcp0:1 interfaces of the vice-master are configured but are not marked UP.

Mapping the Floating External Address of the Master Node

The External Address Manager (EAM) and IP MultiPathing (IPMP) both manage the external floating addresses. The external floating addresses are configured, but are initially in DOWN state. EAM is the program that changes the state to UP on the master node.

▼ To Map the Floating External Address of the Master Node

1. Log in to a peer node.
2. Identify the master node:

```
# nhcmmstat -c all
```

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

3. Log in to the master node.

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

For a node using the class C addressing scheme, the EAM configuration is similar to the following examples.

For Solaris systems:

```
Node.External.FloatingAddress.0=12.10.10.1
Node.External.Monitor.Group.0=ext_group
```

In this example, a floating external address is configured on the master node. The floating external address is 12.10.10.1, and the physical interface belongs to the IPMP group `ext_group`. The logical interface `hme2` has IPMP's test address, which is 12.10.10.225. Note that both addresses belong to the same subnetwork. After failover or switchover, the floating external address is configured on the new master node.

For Linux systems:

```
Node.External.FloatingAddress.0=12.10.10.1
Node.External.Monitor.Group.0=bond0 eth2 eth3
```

In this example, a floating external address is configured on the master node. The floating external address is 12.10.10.1, and the physical interfaces `eth2` and `eth3` are attached to the interface `bond0`. All three interfaces have the same IP address, 12.10.10.1.

5. Run the ifconfig command on the master node:

```
# ifconfig -a
```

The following table shows sample output for the Solaris OS:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
    inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
    inet 10.25.3.25 netmask ffffffff00 broadcast 10.25.3.255
    ether 0:0:0:0:0:0
cgtp0:1:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 2
    inet 10.25.3.1 netmask ffffffff00 broadcast 10.25.3.255
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 10.25.1.25 netmask ffffffff00 broadcast 10.25.1.255
    ether 8:0:20:f9:a5:56
hme0:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 10.17.1.125 netmask ffffffff00 broadcast 10.17.1.255
hme0:2:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 3
    inet 10.25.1.1 netmask ffffffff00 broadcast 10.25.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
    inet 10.25.2.25 netmask ffffffff00 broadcast 10.25.2.255
    ether 8:0:20:f9:a5:57
hme1:1:
flags=1040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500 index 4
    inet 10.25.2.1 netmask ffffffff00 broadcast 10.25.2.255
hme2:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu
1500 index 5
    inet 12.10.10.225 netmask ffffffff00 broadcast 12.10.10.255
    groupname ext_group
    ether 0:3:ba:31:c2:5e
hme2:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 5
    inet 12.10.10.1 netmask ffffffff00 broadcast 12.10.10.255
hme3:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu
1500 index 6
    inet 12.10.10.235 netmask ffffffff00 broadcast 12.10.10.255
    groupname ext_group
    ether 0:3:ba:31:c2:5f
```

This output shows the hme2:1 interface as configured in the nhfs.conf file.

On Linux, the sample output for a master node appears as follows:

```
bond0      Link encap:Ethernet  HWaddr 00:03:BA:F1:76:1A
  inet addr:12.10.10.1  Bcast:12.10.10.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:761a/64 Scope:Link
  UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
  TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

cgtp0      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
  inet addr:10.125.3.10  Bcast:10.125.3.255  Mask:255.255.255.0
  inet6 addr: fe80::200:ff:fe00:0/64 Scope:Link
  UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
  TX packets:238077 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:0 (0.0 b)  TX bytes:310149562 (295.7 MiB)

cgtp0:0    Link encap:Ethernet  HWaddr 00:00:00:00:00:00
  inet addr:10.125.3.1  Bcast:10.125.3.255  Mask:255.255.255.0
  UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

eth0       Link encap:Ethernet  HWaddr 00:03:BA:F1:76:18
  inet addr:10.125.1.10  Bcast:10.125.1.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7618/64 Scope:Link
  UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
  RX packets:615003 errors:0 dropped:21216 overruns:0 frame:0
  TX packets:124213 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:840327200 (801.3 MiB)  TX bytes:155735404 (148.5 MiB)
  Interrupt:32

eth0:0     Link encap:Ethernet  HWaddr 00:03:BA:F1:76:18
  inet addr:10.125.1.1  Bcast:10.125.1.255  Mask:255.255.255.0
  UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
  Interrupt:32

eth1       Link encap:Ethernet  HWaddr 00:03:BA:F1:76:19
  inet addr:10.125.2.10  Bcast:10.125.2.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7619/64 Scope:Link
  UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
  RX packets:722144 errors:0 dropped:17459 overruns:0 frame:0
  TX packets:113864 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:1000
  RX bytes:843923725 (804.8 MiB)  TX bytes:154414158 (147.2 MiB)
  Interrupt:33
```

```

eth1:0          Link encap:Ethernet HWaddr 00:03:BA:F1:76:19
    inet addr:10.125.2.1 Bcast:10.125.2.255 Mask:255.255.255.0
UP BROADCAST RUNNING SLAVE MULTICAST MTU:1500 Metric:1
Interrupt:33

eth2            Link encap:Ethernet HWaddr 00:03:BA:F1:76:1B
    inet addr:12.10.10.1 Bcast:12.10.10.255 Mask:255.255.255.0
    inet6 addr: fe80::203:baff:fef1:761b/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:247 errors:0 dropped:0 overruns:0 frame:0
TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:15826 (15.4 KiB) TX bytes:158 (158.0 b)
Interrupt:26

eth2            Link encap:Ethernet HWaddr 00:03:BA:F1:76:1B
    inet addr:12.10.10.1 Bcast:12.10.10.255 Mask:255.255.255.0
    inet6 addr: fe80::203:baff:fef1:761b/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:247 errors:0 dropped:0 overruns:0 frame:0
TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:15826 (15.4 KiB) TX bytes:158 (158.0 b)
Interrupt:26

eth3            Link encap:Ethernet HWaddr 00:03:BA:F1:76:1B
    inet addr:12.10.10.1 Bcast:12.10.10.255 Mask:255.255.255.0
    inet6 addr: fe80::203:baff:fef1:761b/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:248 errors:0 dropped:0 overruns:0 frame:0
TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:15872 (15.5 KiB) TX bytes:462 (462.0 b)
Interrupt:27

lo              Link encap:Local Loopback
    inet addr:127.0.0.1 Mask:255.0.0.0
    inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:7149 errors:0 dropped:0 overruns:0 frame:0
TX packets:7149 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:956060 (933.6 KiB) TX bytes:956060 (933.6 KiB)

```

This output shows the bond0 interface as configured in the `nhfs.conf` file.

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

```
# ifconfig -a
```

The following table shows sample output for the Solaris OS:

```
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
  inet 127.0.0.1 netmask ff000000
cgtp0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
  inet 10.25.3.26 netmask ffffffff0 broadcast 10.25.3.255
  ether 0:0:0:0:0:0
cgtp0:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4>
  mtu 1500 index 2
  inet 10.25.3.1 netmask ffffffff0 broadcast 10.25.3.255
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 10.25.1.26 netmask ffffffff0 broadcast 10.25.1.255
  ether 8:0:20:fa:3f:70
hme0:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
  inet 10.17.1.126 netmask ffffffff0 broadcast 10.17.1.255
hme0:2: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500
index 3
  inet 10.25.1.1 netmask ffffffff0 broadcast 10.25.1.255
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
  inet 10.25.2.26 netmask ffffffff0 broadcast 10.25.2.255
  ether 8:0:20:fa:3f:71
hme1:1: flags=1040842<BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4> mtu 1500
index 4
  inet 10.25.2.1 netmask ffffffff0 broadcast 10.25.2.255
hme2:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu
1500 index 5
  inet 12.10.10.226 netmask ffffffff0 broadcast 12.10.10.255
  groupname ext_group
  ether 0:3:ba:31:c2:77
hme2:1: flags=1000842<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 5
  inet 12.10.10.1 netmask ffffffff0 broadcast 12.10.10.255
hme3:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu
1500 index 5
  inet 12.10.10.236 netmask ffffffff0 broadcast 12.10.10.255
  groupname ext_group
  ether 0:3:ba:31:c2:78
```

This output shows that the external floating address is configured but in a DOWN state.

On Linux, the sample output for a vice-master node appears as follows:

```
bond0      Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:5A
  inet addr:12.10.10.1  Bcast:12.10.10.255  Mask:255.255.255.0
BROADCAST MASTER MULTICAST  MTU:1500  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

cgtp0      Link encap:Ethernet  HWaddr 00:00:00:00:00:00
  inet addr:10.125.3.20  Bcast:10.125.3.255  Mask:255.255.255.0
  inet6 addr: fe80::200:ff:fe00:0/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:137149 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b)  TX bytes:17165584 (16.3 MiB)

eth0       Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:58
  inet addr:10.125.1.20  Bcast:10.125.1.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7b58/64 Scope:Link
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX packets:16230 errors:0 dropped:0 overruns:0 frame:0
TX packets:68990 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:2442974 (2.3 MiB)  TX bytes:8636946 (8.2 MiB)
Interrupt:32

eth1       Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:59
  inet addr:10.125.2.20  Bcast:10.125.2.255  Mask:255.255.255.0
  inet6 addr: fe80::203:baff:fef1:7b59/64 Scope:Link
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX packets:190888 errors:0 dropped:0 overruns:0 frame:0
TX packets:68159 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:260118091 (248.0 MiB)  TX bytes:8528638 (8.1 MiB)
Interrupt:33

eth2       Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:5B
  inet addr:12.10.10.1  Bcast:12.10.10.255  Mask:255.255.255.0
BROADCAST MULTICAST  MTU:1500  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
Interrupt:26
```

```

eth3      Link encap:Ethernet  HWaddr 00:03:BA:F1:7B:5B
          inet addr:12.10.10.1  Bcast:12.10.10.255  Mask:255.255.255.0
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:27

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:840 errors:0 dropped:0 overruns:0 frame:0
          TX packets:840 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:131099 (128.0 KiB)  TX bytes:131099 (128.0 KiB)

```

This output shows that the external floating address is configured, but in a DOWN state on eth2 and eth3.

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 if the node is running the Solaris OS
- An entry in `/etc/network/interfaces` if the node is running Linux

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 35](#).

The network configuration files contain the following example information. In the following examples, the node is running the Solaris OS and the name is `node25`.

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

These files define the host names, IP addresses, and attributes for the `hme0` and `hme1` interface 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 as follows:

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. Carrier Grade Transport Protocol (CGTP) uses the routing table 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.

Note – Because CTGP on Linux uses ARP requests to resolve the CGTP destination and does not require entries in the routing table, the following section is applicable only for Solaris nodes.

▼ 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
```

The following table shows sample output for a node using the class C addressing scheme:

Routing Table: IPv4					
Destination	Gateway	Flags	Ref	Use	Interface
node26-cgtp	node26	UGHMS	1	8	
node26-cgtp	node26-nic1	UGHMS	1	8	
10.25.3.255	10.25.1.255	UGHMS	1	0	
10.25.3.255	10.25.2.255	UGHMS	1	0	
node32-cgtp	node32	UGHMS	1	4	
node32-cgtp	node32-nic1	UGHMS	1	4	
master-cgtp	master-nic0	UGHMS	1	1	
master-cgtp	master-nic1	UGHMS	1	1	
12.10.10.0	ipmp-host	U	1	9826	hme2:1
12.10.10.0	ipmp-host	U	1	0	hme2
10.25.3.0	node25-cgtp	U	1	0	cgtp0
10.25.3.0	node25-cgtp	U	1	0	cgtp0:1
10.25.2.0	node25-nic1	U	1	7	hme1
10.25.2.0	node25-nic1	U	1	0	hme1:1
10.25.1.0	node25	U	1	7	hme0
10.25.1.0	node25	U	1	0	hme0:2
10.17.1.0	16-t1-25	U	1	3	hme0:1
224.0.0.0	node25-cgtp	U	1	0	cgtp0
default	10.17.1.1	UG	1	4	
localhost	localhost	UH	4	40	lo0

For a description of the column headings, see [“Output of the netstat -r Command” on page 52](#).

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

- This is a three-node cluster. The cluster contains the nodes node26-cgtp, node32-cgtp, and the current node.
- The CGTP routes to broadcast 10.25.3.255 are as follows:

Destination	Gateway	Flags	Ref	Use	Interface
10.25.3.255	10.25.1.255	UGHMS	1	0	
10.25.3.255	10.25.2.255	UGHMS	1	0	

- The CGTP routes to node 26 are as follows:

Destination	Gateway	Flags	Ref	Use	Interface
node26-cgtp	node26	UGHMS	1	8	
node26-cgtp	node26-nic1	UGHMS	1	8	

- 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 node 32 are as follows:

Destination	Gateway	Flags	Ref	Use	Interface
node32-cgtp	node32	UGHMS	1	4	
node32-cgtp	node32-nic1	UGHMS	1	4	

The other entries in the table are relative to the external addresses (floating and static) and the standard Solaris routes.

To change the entries in the routing table, use the `route` command as described in the `route1M` 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	<ul style="list-style-type: none"> • 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 <code>multirt</code> option. • S – The route is a redundant route established with the <code>setsrc</code> option. • U – The route is up.

TABLE 4-1 Description of the Output of the `netstat -r` Command (*Continued*)

Field	Description
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 to which the data should be sent, the source address is set to the value specified in the route.

Debugging Diskless Nodes With the `snoop` or `tcpdump` Tools

When debugging diskless nodes, use the `snoop` command for Solaris systems, or the `tcpdump` command for Linux systems, as follows.

▼ To Examine a Diskless Node From the Master Node

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, examine the first Ethernet address.

On Solaris:

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

On Linux:

```
% tcpdump -i NIC0 ether host diskless-node-NIC0-Ethernet-address
```

The `snoop` or `tcpdump` command captures packets from the network and displays their contents.

5. From another console window, examine the next Ethernet address.

On Solaris:

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

On Linux:

```
% tcpdump -i NIC1 ether host diskless-node-NIC1-Ethernet-address
```

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

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 55](#)
- [“Changing the Location of the Scoreboard Bitmap” on page 58](#)
- [“Managing Differences Between Files That Are Not Shared” on page 60](#)
- [“Using the Naming Services” on page 61](#)

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. If the cluster is running the Solaris OS, consider using the `O_SYNC` or `O_DSYNC` mount options on some files. For information about these options, see the `fcntl3head` 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 on Solaris

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_3.0/opt \
- /SUNWcgha/services nfs - no rw,hard,fg,intr,noac
master-cgtp:/SUNWcgha/local/export/services/ha_3.0 - \
/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_3.0/opt \
- /SUNWcgha/services nfs - no rw,hard,fg,intr
master-cgtp:/SUNWcgha/local/export/services/ha_3.0 - \
/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 76](#).
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.

▼ To Enable or Disable Data and Attribute Caching on Linux

1. Log in to the vice-master node as superuser.
2. Open the `/etc/fstab` 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      noauto,rw,hard,fg,intr,noac 0 0  
  
master-cgtp:/SUNWcgha/local/export/services/ha_3.0/opt \  
/SUNWcgha/services    nfs      noauto,rw,hard,fg,intr,noac 0 0  
  
master-cgtp:/SUNWcgha/local/export/services/ha_3.0 \  
/SUNWcgha/swdb nfs      noauto,rw,hard,fg,intr,noac 0 0
```

- 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      noauto,rw,hard,fg,intr 0 0  
  
master-cgtp:/SUNWcgha/local/export/services/ha_3.0/opt \  
/SUNWcgha/services    nfs      noauto,rw,hard,fg,intr 0 0  
  
master-cgtp:/SUNWcgha/local/export/services/ha_3.0 \  
/SUNWcgha/swdb nfs      noauto,rw,hard,fg,intr 0 0
```

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:

```
# reboot -n -f
```

6. Trigger a switchover, as described in [“To Trigger a Switchover With `nhcmmstat`”](#) on page 76.

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.

Note – The term "scoreboard bitmap" is the Solaris name for the data area used to keep track of modifications to data blocks. On Linux, the term "DRBD metadata" is used for the same purpose. DRBD on Linux supports only metadata stored on disk. The information presented in this section of the guide applies only to clusters running the Solaris OS.

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.

The scoreboard bitmap is only needed for IP-replicated systems. Systems using shared disk do not need it.

For examples of the two methods available for storing the scoreboard bitmaps, see "IP Mirroring" in the *Netra High Availability Suite 3.0 1/08 Foundation Services Overview*. For information about how to reconfigure the scoreboard bitmap, see the following section and procedure.

Storing Bitmap Scorecards on Disk or in Memory

The "bitmaps on disk" and "bitmaps in memory" setting is a system-wide tunable, which cannot be set for each slice. The setting can be changed in the file `/usr/kernel/drv/rdc.conf`. Netra HA Suite software supports two of the available modes for the `rdc_bitmap_mode` parameter:

- `rdc_bitmap_mode=1` (store the bitmap on the replicated partition) forces bitmap writes for every write operation, so an update resync can be performed after a crash or reboot.
- `rdc_bitmap_mode=2` (store the bitmap in memory) only writes the bitmap on shutdown, so a full resync is required after a crash, but an update resync is required after a reboot.

These options have the following advantages and drawbacks:

- Storing the bitmap to memory (mode 2) is preferred over storing it on the replicated partition (mode 1) when higher throughput is to be considered; mode 2 is about 50 percent faster than mode 1.
- Mode 2 is preferred over mode 1 when many writers are to be considered. In case of failover recovery, the time required to become "Synchro Ready" (for example, the time required for replicated slices to become synchronized again) can be significantly longer if mode 1 is used instead of mode 2.
- Storing the bitmap on the replicated partition (mode 1) is preferred over storing it to memory (mode 2) when faster recovery in case of dual failure (for example, if both the master and vice-master fail) is to be considered. When mode 2 is used, a full synchronization takes place when a node is elected Master upon boot; when using mode 1, only a regular synchronization happens. Future enhancements to the product should improve the resynchronization time involved when using mode 2.

Note – Storing the scoreboard bitmap in memory is encouraged only when data is continuously and frequently updated, or when data is written to a replicated partition during a switchover. In most other cases, storing the scorecard bitmap to memory is not recommended because synchronizing the partitions after a switchover or during a full synchronization can require a significant amount of time.

▼ 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 76.
7. **Repeat Step 1 through 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 26.

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.

<code>cluster_nodes_table</code>	Contains the <i>nodeid</i> and node name of each peer node. For more information, see the <code>cluster_nodes_table4</code> man page.
<code>/etc/hosts</code>	Contains the hostnames of all nodes on the cluster network. For more information, see the <code>hosts4</code> man page.
<code>nhfs.conf</code>	Describes the cluster configuration, including network interfaces, mirrored disk partitions, and the floating external address. For more information, see the <code>nhfs.conf4</code> man page.

To manage differences that exist between files that are not shared, perform the procedure as follows.

▼ 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.conf4` 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 `nhadm1M` man page.

Using the Naming Services

This section provides guidelines for using naming services with 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 [...]
```


Starting and Stopping Services, Nodes, and Clusters

This chapter describes how to stop and start the Netra HA Suite software, a node, or a cluster. This chapter contains the following sections:

- [“Stopping and Restarting the Foundation Services” on page 63](#)
- [“Stopping and Restarting Daemon Monitoring” on page 66](#)
- [“Shutting Down and Restarting a Node” on page 67](#)
- [“Shutting Down and Restarting a Cluster” on page 73](#)
- [“Triggering a Switchover” on page 76](#)
- [“Recovering an IP-Replicated Cluster” on page 77](#)

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 Netra HA Suite software.
2. Create the `not_configured` file on the node.

On Solaris OS systems:

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

On Linux systems:

```
# touch /etc/opt/sun/nhas/not_configured
```

3. Reboot the node as described in [“To Perform a Clean Reboot of a Solaris OS Node” on page 68](#) or [“To Perform a Clean Reboot of a Linux Node” on page 68](#).

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 nhcmmnd
```

If the Foundation Services have been stopped, no process identifier should appear for the `nhcmmnd` daemon.

▼ To Stop and Restart the Foundation Services Without Stopping the Solaris OS

Use this procedure to restart the Foundation Services when the Solaris OS does not need to come down (to apply a new patch, for example).

1. Go to single-user mode:

```
# init s
```

2. Return to multi-user mode:

```
# init 3
```

▼ To Stop and Restart the Foundation Services Without Stopping Linux

Use this procedure to restart the Foundation Services when Linux does not need to come down (to apply a new patch, for example).

1. Go to single-user mode:

```
# telinit 1
```

2. Return to multi-user mode:

```
# telinit 3
```

▼ 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 64](#).

1. Log in as superuser to the node on which you want to restart the Foundation Services.
2. Check that the `not_configured` file is not present.
The file is located at `/etc/opt/SUNWcgha/not_configured` on Solaris systems, and `/etc/opt/sun/nhas/not_configured` on Linux systems. If that file is present, delete it.
3. Reboot the node as described in [“To Perform a Clean Reboot of a Solaris OS Node” on page 68](#) or in [“To Perform a Clean Reboot of a Linux Node” on page 68](#), depending on the OS your system uses.
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 `nhadm1M` 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 3.0 1/08 Foundation Services Troubleshooting Guide*.

▼ To Stop Daemon Monitoring

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

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

2. **Create the special file:**

If the node is running the Solaris OS:

```
# touch /etc/opt/SUNWcgha/not_under_pmd_control
```

If the node is running Linux:

```
# touch /etc/opt/sun/nhas/not_under_pmd_control
```

3. **Reboot the node as described in “To Perform a Clean Reboot of a Solaris OS Node” on page 68 or in “To Perform a Clean Reboot of a Linux Node” on page 68, depending on the OS your system uses.**

The Foundation Services start, and the OS and Netra HA Suite daemons that were monitored are no longer monitored.

▼ To Restart Daemon Monitoring

If Daemon Monitoring was stopped using “To Stop Daemon Monitoring” on page 66, restart Daemon Monitoring as follows:

1. **Log in to the node on which you want to restart the Daemon Monitoring.**
2. **Remove the special file.**

If the node is running the Solaris OS:

```
# rm /etc/opt/SUNWcgha/not_under_pmd_control
```

If the node is running Linux:

```
# rm /etc/opt/sun/nhas/not_under_pmd_control
```

3. **Reboot the node as described in “To Perform a Clean Reboot of a Solaris OS Node” on page 68 or in “To Perform a Clean Reboot of a Linux Node” on page 68, depending on the OS your system uses.**

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

Shutting Down and Restarting a Node

This section describes how to shut down and 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 procedures:

▼ To Perform a Clean Reboot of a Solaris OS Node

Determine if the Foundation Services are running.

1. **If the Foundation Services are not running, use init 6 to reboot a node.**
2. **When the Foundation Services are running, use the following procedure:**
 - a. **Stop the user applications.**
 - b. **Flush the file systems:**

```
# lockfs -fa
```

- c. **Perform an immediate reboot:**

```
# uadmin 1 1 /*A_REBOOT AD_REBOOT*/
```

▼ To Perform a Clean Reboot of a Linux Node

Determine if the Foundation Services are running.

1. **If the Foundation Services are not running, use init 6 to reboot a node.**
2. **When the Foundation Services are running, use the following procedure:**
 - a. **Stop the user applications.**
 - b. **Flush the file systems:**

```
# sync
```

- c. **Perform an immediate reboot:**

```
# reboot -n -f
```

▼ To Perform a Clean Power off of a Solaris Node

Determine if the Foundation Services are running.

1. **If the Foundation Services are not running, use init 5 to power off a node.**
2. **When the Foundation Services are running use the following procedure:**
 - a. **Stop the user applications.**

b. Flush the file systems:

```
# lockfs -fa
```

c. Perform an immediate power off:

```
# uadmin 1 6 /*A_REBOOT AD_POWEROFF*/
```

▼ To Perform a Clean Power off of a Linux Node

Determine if the Foundation Services are running.

1. If the Foundation Services are not running, use `poweroff` to power off a node.
2. When the Foundation Services are running use the following procedure:
 - a. Stop the user applications.
 - b. Flush the file systems:

```
# sync
```

c. Perform an immediate power off:

```
# poweroff -n -f
```

▼ To Perform a Clean Halt of a Solaris Node

Determine if the Foundation Services are running.

1. If the Foundation Services are not running, use `init 0` to halt a node.
2. When the Foundation Services are running use the following procedure:
 - a. Stop the user applications.
 - b. Flush the file systems:

```
# lockfs -fa
```

c. Perform an immediate halt:

```
# uadmin 1 0 /*A_REBOOT AD_HALT*/
```

▼ To Perform a Clean Halt of a Linux Node

Determine if the Foundation Services are running.

1. If the Foundation Services are not running, use `halt` to halt a node.
2. When the Foundation Services are running use the following procedure:
 - a. Stop the user applications.
 - b. Flush the file systems:

```
# sync
```

- c. Perform an immediate halt:

```
# halt -n -f
```

▼ To Abruptly Reboot a Solaris Node

- Reboot the node:

```
# uadmin 1 1 /*A_REBOOT AD_BOOT*/
```

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

▼ To Abruptly Reboot a Linux Node

- Reboot the node:

```
# reboot -n -f
```

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

▼ To Abruptly Power Off a Solaris Node

- Power off the node:

```
# uadmin 1 6 /*A_REBOOT AD_POWEROFF*/
```

The node stops immediately without any further processing.

▼ To Abruptly Power Off a Linux Node

- **Power off the node:**

```
# poweroff -n -f
```

The node stops immediately without any further processing.

▼ To Abruptly Halt a Solaris Node

- **Halt the node:**

```
# uadmin 1 0 */A_REBOOT AD_HALT/*
```

The node stops immediately without any further processing.

▼ To Abruptly Halt a Linux Node

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

```
# halt -n -f
```

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 76](#). The vice-master node becomes the new master node, and the old master node becomes 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](#).

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 as described in “To Perform a Clean Power off of a Solaris Node” on page 68 or “To Perform a Clean Power off of a Linux Node” on page 69, depending on the OS your system uses.**

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 as described in “To Perform a Clean Power off of a Solaris Node” on page 68 or “To Perform a Clean Power off of a Linux Node” on page 69, depending on the OS your system uses.**

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 as described in “To Perform a Clean Power off of a Solaris Node” on page 68 or “To Perform a Clean Power off of a Linux Node” on page 69, depending on the OS your system uses.**

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.

Note – For x64 platforms, refer to the hardware documentation for information about performing tasks that reference OBP commands and that, therefore, apply only to the SPARC architecture.

▼ 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 `nhadm1M` 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 as described in [“To Perform a Clean Power off of a Linux Node”](#) on page 69.**

4. **Verify that the vice-master node is synchronized with the master node (not applicable for shared disk configurations):**

For versions of the Solaris OS earlier than version 10:

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

For the Solaris 10 OS and later:

```
# /usr/sbin/dsstat 1
```

For the Linux OS:

```
# drbdadm cstate all
```

- If the `drbdadm` command indicates "Connected," the vice-master node is synchronized with the master node.
- 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 following the steps provided in [“To Perform a Clean Power off of a Solaris Node” on page 68](#) or in [“To Perform a Clean Power off of a Linux Node” on page 69](#), depending on the OS your system uses.**
6. **Shut down the master node by logging in to the master node and following the steps provided in [“To Perform a Clean Power off of a Solaris Node” on page 68](#) or in [“To Perform a Clean Power off of a Linux Node” on page 69](#), depending on the OS your system uses.**

For further information about the `init` command, see the `init1M` 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 the cluster was shut down. If you do not maintain the roles of the nodes, you might lose data on systems using IP replication.

1. Access the the master node's system console and type the following:

```
ok> boot
```

Note – For x64 platforms, refer to the hardware documentation for information about performing tasks that reference OpenBoot™ PROM (OBP) commands and, therefore, apply only to the SPARC architecture.

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

```
# nhadm check configuration
```

3. Access the vice-master node's system console and type the following:

```
ok> boot
```

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

```
# nhadm check configuration
```

5. Access the system consoles of each diskless or dataless node and type the following:

```
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 it had before it was shut down.



Caution – After an emergency shutdown, 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.conf4` and `cluster_definition.conf4` 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 26](#). 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 now the vice-master node:

```
# nhcmmstat -c vice
```

For more information, see the `nhcmmstat1M` man page.

Recovering an IP-Replicated Cluster

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 3.0 1/08 Foundation Services Troubleshooting Guide*.

The following procedure is specific to IP-replicated clusters because a split brain error is unlikely to happen with a shared disk configuration. For shared disk, just check that the configuration is normal and then reboot.

▼ To Recover a Solaris Cluster After Failure

1. **Stop all of the nodes in the cluster as described in “To Perform a Clean Power off of a Solaris Node” on page 68.**
2. **Boot both of the master-eligible nodes in single-user mode.**

```
ok> boot -s
```

Note – For x64 platforms, refer to the hardware documentation for information about performing tasks that reference OBP commands and, therefore, apply only to the SPARC architecture.

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):

On the Solaris 9 OS:

```
# /opt/SUNWesm/SUNWrdc/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

On the Solaris 10 OS:

```
# /usr/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

c. Synchronize the file system by using `/sbin/sync`.

d. 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 any exist.

You can boot diskless and dataless nodes in any order.

▼ To Recover a Linux Cluster After Failure

1. Stop all peer nodes in the cluster as described in [“To Perform a Clean Power off of a Linux Node”](#) on page 69.

2. Restart both of the master-eligible nodes with Netra HA Suite software disabled. Note which node is master and which node is vice-master before restarting the nodes.

```
# touch /etc/opt/sun/nhas/not_configured
# reboot -n -f
```

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 DRBD replication configuration:

On the vice-master node:

```
# drbdadm secondary all
```

On the master node:

```
# drbdadm primary all
# drbdadm invalidate_remote all
```

This will trigger a full re-synchronization from the master node to the vice-master node.



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

- c. Wait until the master node and vice-master node are synchronized. This is a full re-synchronization and might take some time.

- d. Remove the not_configured file on both the master and vice-master node:

```
# rm /etc/opt/sun/nhas/not_configured
```

4. Boot the nodes in the following order:

- a. Boot the first master-eligible node.
- 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.**
- f. **Boot the diskless and dataless nodes, if any exist.**

You can boot diskless and dataless nodes in any order.

Patching Software on a Solaris 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 patch details to the `addon.conf` file. For further information, see the `addon.conf4` and `nhinstall11M` man pages.

For information about how to manually add a patch to the Netra HA Suite software and Solaris packages, see the following sections:

- [“About Patching Software on a Solaris Cluster” on page 81](#)
- [“Patching a Nonshared Package on Both Master-Eligible Nodes” on page 83](#)
- [“Patching a Dataless Node” on page 84](#)
- [“Patching a Diskless Node” on page 85](#)
- [“Patching a Shared Package” on page 87](#)

About Patching Software on a Solaris Cluster

Some Netra HA Suite packages are shared 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.

The patch database holds information about the patches. 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 `patchadd1M` man page.

Choosing a Procedure to Patch Software on a Cluster

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

- Solaris package or nonshared Netra HA Suite 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 Netra HA Suite package on a master-eligible node, see [“To Patch a Nonshared Package on Both Master-Eligible Nodes” on page 83](#).

- Solaris package or nonshared Netra HA Suite 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 83](#).

- 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 85](#).

- Solaris package or Netra HA Suite 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 86](#).

- Shared packages

A shared package can be a Netra HA Suite 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 [“To Patch a Shared Package on a Cluster” on page 87](#).

Overwriting Modified Netra HA Suite Files

The following patches overwrite Netra HA Suite files:

- Solaris patch that changes the IP stack. The IP stack is modified during installation of CGTP.
- 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 Netra HA Suite software.

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 83.**
3. **Repeat Step 1 and 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 [“To Patch the /usr Directory for Diskless Nodes” on page 85](#). For information about how to apply a patch to a package installed in the `root` directory, see [“To Patch the root File System for Diskless Nodes” on page 86](#).

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 83.**

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_10`.

- 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_10`.

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 Netra HA Suite 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 83.

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 as described in “To Perform a Clean Reboot of a Linux Node” on page 68.**
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 Netra HA Suite 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 83.**

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 `nhadm1Mman` 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/ha_3.0 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/ha_3.0 patchid
```

5. Verify that the patch has been installed:

```
# patchadd -p -R /SUNWcgha/local/export/services/ha_3.0
```

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/ha_3.0 | \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](#).

Patching Software on a Linux 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 patch details to the `addon.conf` file. For further information, see the `addon.conf5` and `nhinstall8` man pages.

For information about how to manually add a patch to the Netra HA Suite software and Linux packages, see the following sections:

- [“About Patching Software on a Linux Cluster” on page 89](#)

About Patching Software on a Linux Cluster

Patches for Linux are normally shipped as a new complete version of an RPM package. They ship with a README file that contains patch installation instructions. Generally, you use the `rpm -F` command to install a patch, however, special installation instructions might be included in the README file.

The RPM database holds node-specific information about the installed packages. The database describes the file system where the package contents are installed. To view the packages that are currently installed on a node, use the `rpm -qa` command. For more information, see the `rpm8man` page.

▼ To Patch a 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/tmp` directory.**
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/tmp`, install the patch as follows:

```
# rpm -F /var/tmp package-name.rpm
```

This will freshen your existing RPM installation.

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

```
# rpm -qa
```

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

```
# rpm -q package-name
```

The package for which you specified the `package-name` 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.**

▼ 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 Package on Both Master-Eligible Nodes” on page 90.**
3. **Repeat Step 1 and Step 2 for each dataless node in the cluster.**

Modifying and Adding Disk Partitions for Replicated Data on the Solaris OS

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 3.0 1/08 Foundation Services 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 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

This chapter contains the following sections:

- “Using the `format` Utility to Display and Modify the Configuration of a Disk Partition on a Solaris Node” on page 94
- “Increasing the Size of a Replicated Data Partition on a Physical Disk on a Solaris Node” on page 99
- “Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node” on page 102
- “Adding a Mirrored Data Partition to a Virtual Disk on a Solaris Node” on page 104

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

This section describes how to use the `format` utility to display and modify the configuration of a disk partition. For additional information, see the `format1M` 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 on a Solaris Node

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 the following 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 the following 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 the following 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 the following 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
```

The following 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 on the Solaris OS

1. Perform Step 1 through Step 5 of [“To Display the Configuration of a Disk Partition on a Solaris Node”](#) on page 94.
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 the following 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[wn]  
Enter new starting cyl(2178):
```

4. Change the size of the partition:

```
new-partition-size]: [4194828b, 1452c, 2048.26mb, 2.00gbEnter  
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 Replicated Data Partition on a Physical Disk on a Solaris Node

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

▼ To Increase the Size of a Replicated Data Partition on a Physical Disk on a Solaris Node

1. **Log in to a cluster node and note which node is the master node and which node is the vice-master node.**

```
# nhcmmstat -all
```

2. Stop the cluster.

For information, see [“To Shut Down a Cluster” on page 73.](#)

3. Log in to the original master node in single-user mode as superuser:

```
ok> boot -s
```

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

For information, see the `ufsdump1M` man page.

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

For information, see [“To Display the Configuration of a Disk Partition on a Solaris Node” on page 94.](#)

6. Change the size of the data partition.

For information, see [“To Change the Size of a Disk Partition on a Master-Eligible Node on the Solaris OS” on page 97.](#)

7. 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 a Solaris Node” on page 94.](#)

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 the Solaris OS” on page 97.](#)

8. Restore the data on the partition using the `ufsrestore` command.

For information, see the `ufsrestore1M` man page.

9. Reset the replication configuration on the master node:

On the Solaris 9 OS:

```
# /opt/SUNWesm/SUNWrdc/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

On the Solaris 10 OS:

```
# /usr/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

10. Log in to the old vice-master node in single-user mode, as superuser:

```
ok> boot -s
```

11. Change the size of the data partition by repeating Step 5 through Step 7.
12. Reset the replication configuration on the vice-master node by repeating Step 9 and Step 10.
13. Reboot the original master node as described in [“To Perform a Clean Reboot of a Solaris OS Node”](#) on page 68.
14. When the master node has booted fully, reboot the original vice-master node as described in [“To Perform a Clean Reboot of a Solaris OS Node”](#) on page 68.
15. Log in to the master node as superuser.
16. Verify that the vice-master node is synchronized with the master node:
For versions earlier than the Solaris 10 OS:

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

For the Solaris 10 OS and later:

```
# /usr/sbin/dsstat 1
```

17. Restart the diskless and dataless nodes.
For information, see [“To Restart a Node”](#) on page 73.
18. Verify that the nodes have the correct configuration:

```
# nhadm check
```

For information, see the `nhadm1M` man page.

Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node

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

▼ To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Solaris Node

1. **Log in to a cluster node and note which node is the master node and which node is the vice-master node:**

```
# nhcmmstat -all
```

2. **Stop the cluster.**

For information, see [“To Shut Down a Cluster” on page 73](#).

3. **Log in to the original master node in single-user mode as superuser:**

```
ok> boot -s
```

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

For information, see the `ufsdump1M` man page.

5. **Mount the file system.**

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

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


6. Add space to the virtual disk.

For example, add 10 Gbytes to the d20 device:

```
# metattach d20 10g
```

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

7. Grow the file system to its new size:

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

8. 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 a Solaris Node” on page 94](#).

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 the Solaris OS” on page 97](#).

9. Reset the replication configuration:

On the Solaris 9 OS:

```
# /opt/SUNWesm/SUNWrdc/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

On the Solaris 10 OS:

```
# /usr/sbin/sndradm -d
Disable Remote Mirror? (Y/N) [N]: Y
#
```

10. Log in to the original vice-master node in single-user mode, as superuser:

```
ok> boot -s
```

11. Add space to the virtual disk by repeating Step 6.

12. Repeat Step 8 through Step 9 on the vice-master node.

13. **Reboot the master node as described in “To Perform a Clean Reboot of a Solaris OS Node” on page 68.**
14. **When the master node has fully booted, reboot the vice-master node as described in “To Perform a Clean Reboot of a Solaris OS Node” on page 68.**
15. **Log in to the master node as superuser.**
16. **Verify that the vice-master node is synchronized with the master node:**
For versions earlier than the Solaris 10 OS:

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

For the Solaris 10 OS and later:

```
# /usr/sbin/dsstat 1
```

17. **Restart the diskless nodes or dataless nodes.**
For information, see “To Restart a Node” on page 73.
18. **Confirm that the nodes have the correct configuration:**

```
# nhadm check
```

For information, see the `nhadm1M` man page.

Adding a Mirrored Data Partition to a Virtual Disk on a Solaris Node

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 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

▼ To Add a Mirrored Data Partition to a Virtual Disk on a Solaris Node

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 the Solaris 9 and Solaris 10 OS, “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 `format1M` 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.conf4` 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:

For versions earlier than the Solaris 10 OS:

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

For the Solaris 10 OS and later:

```
# /usr/sbin/dsstat 1
```

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 `nhadm1M` man page.

Modifying and Adding Disk Partitions for Replicated Data on Linux

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 3.0 1/08 Foundation Services 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 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

This chapter contains the following sections:

- “Using the `parted` Utility to Display and Modify the Configuration of a Disk Partition on a Linux Node” on page 108
- “Increasing the Size of a Replicated Data Partition on a Physical Disk on a Linux Node” on page 110
- “Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node” on page 111
- “Adding a Mirrored Data Partition to a Virtual Disk on a Linux Node” on page 114

Using the parted Utility to Display and Modify the Configuration of a Disk Partition on a Linux Node

This section describes how to use the `parted` utility to display and modify the configuration of a disk partition. For additional information, see the `parted8` man page.

Before you change the configuration of a disk partition, you must stop all the nodes in the cluster. 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 on a Linux Node

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 `parted` utility to display partitions of the disk:**

```
# parted /dev/sda print
```

Output similar to the following is displayed in the console window:

```
Disk geometry for /dev/scsi/host0/bus0/target0/lun0/disc: 0.000-70007.196 megabytes
Disk label type: msdos
Minor    Start      End        Type       Filesystem  Flags
1         0.031     8189.384   primary    ext3        boot
2        8189.385  12291.921  primary    linux-swap
3       12291.921  70001.982  extended
5       12291.952  16386.613  logical    ext3
6       16386.645  20481.306  logical    ext3
7       20481.337  21501.057  logical    ext3
8       21501.088  39997.771  logical    ext3
Information: Don't forget to update /etc/fstab, if necessary.
```

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

Changing the size of a disk partition using the `parted` command requires that the partition that is going to be changed is deleted, and then recreated with the new size. There must also be enough unused space on the disk to allow the partition to be increased.



Caution – Deleting a partition also deletes all data on that partition. Make a backup copy of the data before deleting a partition.

1. Log in to a master-eligible node as superuser.
2. Delete the partition that you want to modify:

```
# parted /dev/sda rm 8
```

This deletes partition number 8.

3. Recreate the partition with increased size:

```
# parted /dev/sda mkpart logical ext3 21501.058 70007.196
```

4. Verify the change by displaying the new partition table:

```
# parted /dev/sda print
Disk geometry for /dev/scsi/host0/bus0/target0/lun0/disc: 0.000-70007.196
mbytes
Disk label type: msdos
Minor      Start       End         Type        Filesystem  Flags
1           0.031      8189.384    primary     ext3        boot
2          8189.385   12291.921   primary     linux-swaps
3          12291.921   70001.982   extended
5          12291.952   16386.613   logical     ext3
6          16386.645   20481.306   logical     ext3
7          20481.337   21501.057   logical     ext3
8          21501.088   70001.982   logical     ext3
```

Note – Refer to the `parted8man` page for more information about changing the partitions on a disk.

Increasing the Size of a Replicated Data Partition on a Physical Disk on a Linux Node

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

▼ To Increase the Size of a Replicated Data Partition on a Physical Disk on a Linux Node

1. **Log in to a cluster node and note which node is the master node and which node is the vice-master node.**

```
# nhcmmstat -all
```

2. **Stop the cluster.**

For information, see [“To Shut Down a Cluster” on page 73](#).

3. **Log in to the original master node in single-user mode, as superuser:**

4. **Back up the partition by using a tool such as `dump`.**

For information, see the `dump8` man page.

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

For information, see [“To Display the Configuration of a Disk Partition on a Linux Node” on page 108](#).

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

For information, see [“To Change the Size of a Disk Partition on a Master-Eligible Node on Linux” on page 109](#).

7. **Verify that the metadata partition is the correct size.**

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

For information, see [“To Display the Configuration of a Disk Partition on a Linux Node” on page 108](#).

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

128 Mbyte * Number of replicated file systems

If there are two replicated file systems, then the size of the meta data partition should be at least 256 MBytes.

If the metadata 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 Linux”](#) on page 109.

8. Restore the data on the partition using the `restore` command.

For information, see the `restore8` man page.

9. Log in to the old vice-master node in single-user mode, as superuser:

10. Change the size of the data partition by repeating Step 5 through Step 7.

11. Reboot the original master node as described in [“To Perform a Clean Reboot of a Linux Node”](#) on page 68.

12. When the master node has booted fully, reboot the original vice-master node as described in [“To Perform a Clean Reboot of a Linux Node”](#) on page 68.

13. Log in to the master node as superuser.

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

```
# drbdadm cstate all
```

If the `drbdadm` command indicates "SyncSource," then synchronization is currently taking place. When the `drbdadm` command indicates "Connected," the vice-master node is synchronized with the master node.

15. Restart the diskless and dataless nodes.

For information, see [“To Restart a Node”](#) on page 73.

Increasing the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node

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

▼ To Increase the Size of a Replicated Data Partition on a Virtual Disk on a Linux Node

1. **Log in to a cluster node and note which node is the master node and which node is the vice-master node:**

```
# nhcmmstat -all
```

2. **Stop the cluster.**

For information, see [“To Shut Down a Cluster” on page 73](#).

3. **Log in to the original master node in single-user mode, as superuser:**

4. **Back up the partition by using a tool such as `dump`.**

For information, see the `dump8` man page.

5. **Start the LVM and mount the file system.**

For example, mount `/dev/vg0/lvol4` on `home2`, as follows:

```
# /etc/init.d/nhlmv start
# mount -t ext3 /dev/vg0/lvol4 /home2
```

6. **Add space to the virtual disk by adding a new, unused partition to the volume group.**

For example, add `/dev/sda8` to the `vg0` device:

```
# vgextend vg0 /dev/sda8
```

For more information about virtual disks, see the *Logical Volume Manager How To* at: <http://tldp.org/HOWTO/LVM-HOWTO/>

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

```
# resize2fs /dev/vg0/lvol4
```

8. **Verify that the metadata partition is the correct size.**

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

For information, see [“To Display the Configuration of a Disk Partition on a Linux Node” on page 108](#).

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

128 Mbyte * Number of replicated file systems

If there are two replicated file systems, then the size of the metadata partition should be at least 256 MBytes.

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 Linux”](#) on page 109.

9. Log in to the original vice-master node in single-user mode, as superuser.
10. Add space to the virtual disk on the vice-master by repeating Step 6.
11. Repeat Step 8 on the vice-master node.
12. Reboot the master node as described in [“To Perform a Clean Reboot of a Linux Node”](#) on page 68.
13. When the master node has fully booted, reboot the vice-master node as described in [“To Perform a Clean Reboot of a Linux Node”](#) on page 68.
14. Log in to the master node as superuser.
15. Verify that the vice-master node is synchronized with the master node:
For versions earlier than the Solaris 10 OS:

```
# drbdadm cstate all
```

If the `drbdadm` command indicates "SyncSource," then synchronization is currently taking place. When the `drbdadm` command indicates "Connected," the vice-master node is synchronized with the master node.

16. Restart the diskless nodes or dataless nodes.
For information, see [“To Restart a Node”](#) on page 73.
17. Confirm that the nodes have the correct configuration:

```
# nhadm check
```

For information, see the `nhadm8` man page.

Adding a Mirrored Data Partition to a Virtual Disk on a Linux Node

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 the Logical Volume Manager software in the initial cluster configuration. For information about configuring the LVM, see the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

▼ To Add a Mirrored Data Partition to a Virtual Disk on a Linux Node

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:**

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

The metadata partition can be skipped if the system already has sufficient large metadata partition. The size of the metadata partition must be at least 128 MBytes multiplied by the number of replicated file systems. For information, see the *Logical Volume Manager How To* at: <http://tldp.org/HOWTO/LVM-HOWTO/>

4. **Create a file system on the new partition.**

For information, see the `mkfs8` 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.**

For information, see the `nhfs.conf5` 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:

```
# drbdadm cstate all
```

If the `drbdadm` command indicates "SyncSource," then synchronization is currently taking place. When the `drbdadm` command indicates "Connected," the vice-master node is synchronized with the master node.

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 `nhadm8` man page.

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 117](#)
- [“Determining Whether a Cluster Has Predefined Nodes” on page 118](#)

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 3.0 1/08 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 118](#).

For information about how to add a node to a cluster installed by the `nhinstall` tool or manually, see the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

To use the CMM API to add or remove a node from a cluster, see the `cmm_config_reload3CMM` 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.

- a. Look at the `cluster_nodes_table` file.

The file contains a line for each node in the cluster definition. Count the number of nodes configured in this file. On Solaris systems, the file is located at `/etc/opt/SUNWcgha/cluster_nodes_table`. On Linux systems, the file is located at `/etc/opt/sun/nhas/cluster_nodes_table`.

- b. 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.

Replacing Hardware in a Cluster

This chapter covers the following topics:

- “Preparing to Replace Hardware in a Cluster” on page 119
 - “Replacing a CPU Board on a Node” on page 121
 - “Replacing a CPU Board on a Diskless Node” on page 122
 - “Replacing Ethernet Cards on a Vice-Master or Dataless Node” on page 124
 - “Replacing Ethernet Cards on a Diskless Node” on page 125
 - “Replacing the Disk on the Vice-Master Node” on page 127
 - “Replacing Disks on Both Master-Eligible Nodes Without Cluster Shutdown” on page 130
 - “Replacing a Dataless Node Disk” on page 132
-

Preparing to Replace Hardware in a Cluster

Follow these guidelines before replacing hardware in a 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 one of the following procedures 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.

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

TABLE 12-1 Reference for Replacing Hardware

Node Type	Hardware Type	For Information
Vice-master	CPU Board	“Replacing a CPU Board on a Node” on page 121
	Ethernet Card	“Replacing Ethernet Cards on a Vice-Master or Dataless Node” on page 124
	Disk	“Replacing the Disk on the Vice-Master Node” on page 127
Diskless	CPU Board	“Replacing a CPU Board on a Node” on page 121 or “Replacing a CPU Board on a Diskless Node” on page 122
	Ethernet Card	“Replacing Ethernet Cards on a Diskless Node” on page 125
	Disk	N/A
Dataless	CPU Board	“Replacing a CPU Board on a Node” on page 121
	Ethernet Card	“Replacing Ethernet Cards on a Vice-Master or Dataless Node” on page 124
	Disk	“Replacing a Dataless Node Disk” on page 132

Replacing a CPU Board on a Node

To replace the board on 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 CPU Board on a Diskless Node](#)” on page 122.

▼ 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 at <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 parameters.

Note – For x64 platforms, refer to the hardware documentation for information about performing tasks that reference OpenBoot PROM commands and, therefore, apply only to the UltraSPARC architecture.

The following examples show the OpenBoot PROM parameters for an UltraSPARC-based diskless node and an UltraSPARC®-based master-eligible node.

- An UltraSPARC diskless node has the following OpenBoot PROM 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
```

- An UltraSPARC-based master-eligible node or dataless node has the following OpenBoot PROM parameters:

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

Note – If the auto-boot-retry variable exists on your system, it must be set to true; if it does not exist on your system, disregard references to it in the preceding examples.

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 CPU Board on a Diskless Node

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

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

1. **Identify the IP address (the Ethernet address couplet for the diskless nodes that are to be replaced).**
2. **Replace the diskless nodes using information in the hardware documentation at <http://www.sun.com/products-n-solutions/hardware/docs/>.**
3. **Record the Ethernet addresses of the new network interfaces on the new diskless nodes.**

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

- 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:

```
Netra CP3010, No Keyboard
Copyright 2005 Sun Microsystems, Inc. All rights reserved.
OpenBoot 4.21.0, 2048 MB memory installed, Serial #64803169.
Ethernet address 0:3:ba:dc:d1:61, Host ID: 83dcd161.
```

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.

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

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 as follows:

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.

4. Log in to the master node as superuser.

5. Modify the DHCP configuration for NIC0:

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

The parameters of this command are as follows:

where:

<i>NIC0IP-address</i>	is the IP address of the NIC0 interface
<i>newEthernet-address</i>	is the Ethernet address of the NIC0 interface in DHCP configuration format
<i>subnet1</i>	is the subnet connecting the NIC0 interfaces

6. 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:
where:

<i>NIC1IP-address</i>	is the IP address of the NIC1 interface
<i>newEthernet-address</i>	is the Ethernet address of the NIC1 interface in DHCP configuration format
<i>subnet2</i>	is the subnet connecting the NIC1 interfaces

7. Refresh the DHCP configuration on the master node:

```
# pkill -1 in.dhcpd1
```

8. Reboot the diskless node:

```
ok> boot
```

9. Verify that the node is configured correctly:

```
# nhadm checkm
```

Replacing Ethernet Cards on a Vice-Master or 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 at <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:**

```
# nbadm 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 a Vice-Master or Dataless Node](#)” on page 124. 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 at <http://www.sun.com/products-n-solutions/hardware/docs/>.**
4. **Record the Ethernet addresses of the new network cards.**
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 as follows:

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 NIC0IP-address -i newEthernet-address \-f 'PERMANENT+MANUAL' -  
m NIC0IP-address subnet1
```

The parameters of this command are as follows:

where:

<i>NIC0IP-address</i>	is the IP address of the NIC0 interface
<i>newEthernet-address</i>	is the Ethernet address of the NIC0 interface in DHCP configuration format
<i>subnet1</i>	is 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:
where:

<i>NIC1IP-address</i>	is the IP address of the NIC1 interface
<i>newEthernet-address</i>	is the Ethernet address of the NIC1 interface in DHCP configuration format
<i>subnet2</i>	is 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
```

Replacing the Disk on the Vice-Master Node

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

▼ To Replace the Disk on the Vice-Master Node Using IP Replication

Note – This procedure is supported only for the Solaris OS.

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 at**
<http://www.sun.com/products-n-solutions/hardware/docs/>.
3. **Install the Solaris OS on the vice-master node. Keep the original partitioning configuration (reformat the new disk by recreating the format of the old disk).**
4. **Install the Netra HA Suite software on the vice-master node.**
For information, see the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.
5. **If logical partitioning or IDE disks are used, follow this step and then jump to Step 9. Otherwise, go to Step 6.**
Force a full synchronization:

```
# nhcrfsadm -f all
```

Power on the vice-master node.

The master node will resynchronize the vice-master disk automatically.

6. **If SCSI disks are used and logical partitioning is not used, follow this step and the rest of the procedure.**

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.

7. **Log in to the master node as superuser.**
8. **Accept a replication restart:**

```
# nhcrfsadm -a
```

9. **(All configurations, IDE or SCSI) Verify that the synchronization is complete:**
For versions earlier than the Solaris 10 OS:

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

For the Solaris 10 OS and later:

```
# /usr/sbin/dsstat 1
```

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.

▼ To Replace the Disk on the Vice-Master Node Using Shared Disk

Clusters using shared disk are supported only on the Solaris OS. The information presented in this section does not apply to Linux clusters.

1. **Replace the hardware by using the hardware documentation at**
<http://www.sun.com/products-n-solutions/hardware/docs/>.

2. **Reformat the new disk by recreating the format of the old disk.**
You must restore the local file system from backup.

3. **Restore the disk configuration.**

a. **Install the Solaris Operating System on the vice-master node.**

b. **Install the Netra HA Suite software on the vice-master node.**

For information, see the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

4. **Create the database replicas on the dedicated partition:**

```
# metadb -a -c 3 -f /dev/rdisk/c0t0d0s7
```

5. **Reboot the node in cluster mode.**

The vice-master node joins the cluster.

Note – Reliable NFS will not notice that the disksets on the vice master have not yet been recreated, so a switchover or failover is prohibited.

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

7. **Remove the vice-master node from the diskset node names:**

```
# metaset -s nhas_diskset -d -h netraMEN2-cgtp
```

8. Re-add the vice-master node to the diskset node names:

```
# metaset -s nhas_diskset -a -h netraMEN2-cgtp
```

Replacing Disks on Both Master-Eligible Nodes Without Cluster Shutdown

Disks in master-eligible nodes can be replaced without fully shutting down a cluster. The new disks may have different geometry than the old disks, however, the new disk in one node must be identical to the new disk in the other node, and the disks must be replaced sequentially in both nodes. When you replace disks as described in this section, the cluster is not single-fault tolerant.

This section describes how to replace disks on both master-eligible nodes without fully shutting down the cluster.

▼ To Replace Disks on Both Master-Eligible Nodes Using IP Replication Without Full Cluster Shutdown

Note – For this release of the Netra HA Suite product, this procedure is supported for only master-eligible nodes that are running the Solaris OS.

1. Replace the disk in the vice-master node using the procedure described in the hardware documentation at:

<http://www.sun.com/products-n-solutions/hardware/docs/>

2. Install the Solaris OS on the vice-master node. Keep the original partitioning configuration as much as possible (reformat the new disk by recreating the format of the old disk). The following conditions must be met:

- Device names of the new replicated and bitmap slices/partitions must be preserved.
- Minor and major numbers of these devices must be preserved.
- New replicated slices/partitions must not be smaller than the original slices/partitions.

- The new bitmap partitions must be at least 1 Kbyte + 4 Kbytes per Gbyte of data in the associated new replicated slice/partition.

3. Install the Netra HA Suite software on the vice-master node.

For information, see the *Netra High Availability Suite 3.0 1/08 Foundation Services Manual Installation Guide for the Solaris OS*.

4. If logical partitioning or IDE disks are used, follow this step and then jump to [Step 7](#). Otherwise, go to [Step 5](#).

On the master node, force a full synchronization:

```
master# nhcrfsadm -f all
```

Start the vice-master node with the Foundation Services.

Remove the `/etc/opt/SUNWcgha/not_configured` file, which was created automatically during the installation process and reboot the vice-master node. The master node will resynchronize the vice-master disk automatically.

5. If SCSI disks are used and logical partitioning is not used, follow this step and the rest of the procedure.

Start the vice-master node with the Foundation Services. Remove the `/etc/opt/SUNWcgha/not_configured` file, which was created automatically during the installation process and reboot 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. Accept a replication restart by running the following command on the master node:

```
master# nhcrfsadm -a
```

7. (All configurations, IDE or SCSI) On the master node, verify that the synchronization is complete:

For versions earlier than the Solaris 10 OS:

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

For the Solaris 10 OS and later:

```
master# /usr/sbin/dsstat 1
```

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.

8. Verify that the vice-master node is configured correctly by running the following command on the vice-master node:

```
vice-master# nhadm check
```

9. Abruptly terminate the master node by running the following command on the master node:

```
master# uadmin 1 0
```

Running this command will halt the CPU(s) immediately and failover will occur. The first node with an upgraded disk will take the mastership and the master role. Switchover should not be used, as the vice-master with old disk might be rejected because of insufficient disk space for replication.

10. Repeat steps [Step 1](#) through [Step 8](#) to finish the upgrade of the second node.

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 on the Solaris OS using the `ufsdump` command, use the `ufsrestore` command to restore the file system. If the backup was created on Linux using the `dump` command, use the `restore` command to restore the file system.

5. Reboot the dataless node.

6. Verify that the node is configured correctly:

```
# nhadm check
```

Index

Symbols

- `.rhosts`, summary of, 11, 13
- `/etc/hostname.hme`, network configuration file, 48
- `/etc/hosts` file
 - avoiding naming service conflicts, 61
 - defining loghost, 17
 - names of nodes, 61
 - network configuration file, 49
- `/etc/nodename`, network configuration file, 48
- `/usr` directory, patching for diskless nodes, 85

A

- abrupt halt
 - on Linux, 71
 - on the Solaris OS, 71
- abrupt poweroff, on the Solaris OS, 70, 71
- abrupt reboot
 - on Linux, 70
 - on the Solaris OS, 70
- adding a node, 117
- addressing example
 - class B, 36
 - class C, 36

B

- back up, mirrored data partition
 - See* data partitions, mirrored
- boot command, summary of, 5
- bootparams database, summary of, 10, 12

C

- caching
 - enabling
 - on Linux, 56
 - on the Solaris OS, 55
 - when to use, 55
- CGTP
 - verifying a redundant network, 26
 - viewing the routing table, 50
- class B addressing example, 36
- class C addressing example, 36
- clean halt
 - on Linux, 70
 - on the Solaris OS, 69
- clean poweroff
 - on Linux, 69
 - on the Solaris OS, 68
- clean reboot
 - on Linux, 68
 - on the Solaris OS, 68
- cluster
 - verifying
 - master and vice-master node synchronization, 26
 - presence of master and vice-master node, 24
- Cluster Membership Manager, verifying operation of, 25
- cluster network, mapping, 32
- `cluster_nodes_table`, summary of, 9
- clusters
 - recovering from failure, 77
 - requirements for a valid cluster, 23

- restarting, 74
- shutting down, 73
- shutting down, avoiding during disk replacement, 130
- verifying
 - correct configuration, 28
 - daemon operation, 25
 - network configuration, 25
 - Reliable Boot Service operation, 28
- commands
 - See also* individual command names
 - adding commands to your path, 1
- communication, external network configuration, 38
- configuration files
 - Foundation Services, 8
 - Linux, 11
 - modifying, 9
 - Solaris, 10
- coreadm.conf file, summary of, 10
- crashed daemon, removing links to log files, 20

D

- Daemon Monitor
 - debugging daemons, 53
 - removing links to logfiles, 20
 - verifying that a daemon is running on a node, 25
- data and attribute caching
 - enabling
 - on Linux, 56
 - on Solaris, 55
 - when to use, 55
- data caches, impact to performance, 55
- data partitions, mirrored
 - adding on a virtual disk
 - on Linux, 114
 - on the Solaris OS, 104
 - backing up on a physical disk
 - on Linux, 110
 - on the Solaris OS, 100
 - deleting on Linux, 109
 - increasing size on a physical disk
 - on Linux, 110
 - on the Solaris OS, 99
 - increasing size on a virtual disk
 - on Linux, 111
 - on the Solaris OS, 102
- data replication, viewing routing table, 50

- dataless node
 - patching, 84, 90
 - replacing a disk, 132
 - shutting down, 72
- debugging
 - applications using the `snoop` command, 53
 - diskless nodes, 53
- deleting partitions on Linux, 109
- dfstab file, summary of, 10
- DHCP configuration, modifying, 123
- disk partition
 - change size on the Solaris OS, 97
 - display configuration
 - on Linux, 108
 - on the Solaris OS, 94
 - modify configuration on the Solaris OS, 97
 - on master-eligible nodes
 - on Linux, 107
 - on the Solaris OS, 93
- disk, replacing
 - on a dataless node, 132
 - on a vice-master node, 127
 - on two master-eligible nodes, 130
- diskless nodes
 - debugging, 53
 - examining, 53
 - patching
 - /usr directory, 85
 - root directory, 86
 - shutting down, 72
- Domain Name System (DNS), avoiding naming conflicts, 61
- DRBD metadata, defined, 58
- dump command
 - backing up a mirrored data partition on a physical disk, Linux, 110
 - summary of, 8

E

- errors, checking for, 53
- Ethernet addresses, identifying, 122
- external access, to peer node, 38
- external communication
 - configuring, 38
 - examining network configuration, 38

F

failure, recovering clusters from, 77

files

See also individual file names

managing

nonshared files, 60

shared files, 55

synchronizing, 60

filesystems, growing

on Linux, 112

on the Solaris OS, 103

floating address triplet, examining configuration of, 39

floating external address, examining

ifconfig output, 43

nhfs.conf configuration file, 42

Node State Manager configuration, 41

format command

disk partition configuration

displaying, 94

modifying, 97

summary of, 5

Foundation Services

administration tools, 2

configuration files, 8

modifying configuration files, 9

patching, 82

restarting, 65

stopping, 64

stopping and restarting the Daemon Monitor, 66

G

growing a file system

growfs command on the Solaris OS, 103

resize2fs command on Linux, 112

H

halt, clean node halt

on Linux, 70

on the Solaris OS, 69

highly available services, minimum criteria, 23

hostname.interface-name file, summary of, 10

hosts file, summary of, 10, 12

I

ifconfig command

examining network configuration, 35

summary of, 5, 7

in.dhcp daemon, verifying operation, 28

inetd.conf file, summary of, 10, 12

inittab file, summary of, 10, 12

IP address

example of class B, 36

example of class C, 36

L

Linux

administration tools, 7

files modified to use the Netra HA Suite software, 11

log files

accessing, 15

configuring system log files, 15

logging NFS operations, 20

maintaining, 18

message categories, 15

redirecting log message to files, 16

removing old log files, 20

log messages, redirecting to nonpeer node, 18

loghost, defining in /etc/hosts file, 17

M

man pages, adding to your path, 2

master node

examining Node State Manager configuration, 41

external floating address, 41

floating address triplet, 39

shutting down, 71

verifying the presence of, 24

master-eligible node

disk partitions

on Linux, 107

on the Solaris OS, 93

replacing disks without shutdown, 130

messages

configuring categories, 15

high volume traffic, 17

redirecting to a nonpeer node, 18

mirrored data partition *See* data partition

monitored daemon, verifying daemon is running on a node, 25

mount command, summary of, 6, 7

mount options, using O_SYNC or O_DSYNC, 55

N

names of nodes, avoiding conflicts between, 61

naming services, guidelines for configuration, 61

netmasks

- identifying from the `ifconfig` output, 36

- summary of, 10, 12

netstat command

- examining network configuration, 31

- summary of, 6, 7

- viewing the routing table, 50

netstat r command, description of output, 52

network cards, identifying Ethernet addresses, 122

network configuration

- dataless node, 35

- diskless node, 35

- examining, 31, 32

- floating addresstriplet, 39

- verify for the cluster, 25

- vice-master node, 35

network configuration file

- `/etc/hostname.hmex`, 48

- `/etc/hosts`, 49

- `/etc/nodename`, 48

Network Information Services, avoiding naming conflicts, 61

network interface, determining status of, 35

network topology map, 32

NFS

- logging NFS operations, 20

- traffic, avoid disruption of, 17

nhadm command

- checking

 - cluster startup, 75

 - network configuration, 32

- comparing the content of files, 61

- examining network configuration, 31

- subcommands, 3

- summary of, 3

- verifying

 - cluster configuration, 28

 - peer node communication, 26

nhadmsync.conf file, editing to compare files, 60

nhcmmnd command, verifying operation, 25

nhcmmqualif command, summary of, 3

nhcmmrole command, summary of, 4

nhcmmstat command

- finding the role of a node, 25

- subcommands, 4

- summary of, 4

- triggering switchover, 76

- verifying existence of vice-master node, 29

nhcmmstat command, finding the role of a node, 24

nhcrfsadm command, summary of, 5

nhenablesync command, summary of, 5

nhfs.conf file, summary of, 9

nhpmdadm command, summary of, 5

NIS, avoiding naming conflicts, 61

nma.notifs.txt file, summary of, 9

nma.params.txt file, summary of, 9

nma.properties file, summary of, 9

nma.security file, summary of, 9

nma.targets.txt file, summary of, 9

noac option

- setting

 - on Linux, 56

 - on the Solaris OS, 55

- when to set, 55

nodeid, identifying from the `ifconfig` output, 36

nodename file, summary of, 10

nodes

- abrupt halt

 - on Linux, 71

 - on the Solaris OS, 71

- abrupt poweroff on the Solaris OS, 70

- abrupt reboot

 - on Linux, 70

 - on the Solaris OS, 70

- adding, 117

- clean halt

 - on Linux, 70

 - on the Solaris OS, 69

- clean poweroff

 - on Linux, 69

 - on the Solaris OS, 68

- clean reboot

 - on Linux, 68

 - on the Solaris OS, 68

- configuring for external communication, 38

- debugging diskless nodes, 53

- determining role of, 24

- predefined, 117

- restarting, 72
- shutting down, 67
- verifying a daemon is running on, 25
- nonpeer node, system log messages from peer nodes, 18
- nonshared files, managing, 60
- nonshared packages, patching
 - list of packages to patch, 82
 - on dataless nodes
 - on Linux, 90
 - on the Solaris OS, 84
 - on master-eligible nodes
 - on Linux, 90
 - on the Solaris OS, 83
- not_configured file
 - creating, 64
 - use of, 64
- notrouter file, summary of, 11
- nsswitch.conf file, summary of, 11, 12

O

O_SYNC and O_DSYNC mount options, when to use, 55

P

packages, patching

- on Linux, 89
- on the Solaris OS, 81

parted command

- displaying disk partition configuration, 108
- summary of, 7

partition *See* data partition, 99, 110

patchadd command, summary of, 6

patching

- /usr directory for diskless nodes, 85
- choosing a procedure, 82
- introduction, 81, 89
- nonshared packages on a dataless node
 - on Linux, 90
 - on the Solaris OS, 84
- nonshared packages on master-eligible nodes
 - on Linux, 90
 - on the Solaris OS, 83
- planning, 82
- root directory on diskless nodes, 86
- shared packages, 87

PATH environment variable, adding command and man pages, 1

peer nodes

- accessing externally, 38
- verifying communication of, 26

performance, impact of data caches and shared file systems, 55

pgrep command, summary of, 6, 7

physical disk

- backing up a mirrored data partition
 - on Linux, 110
 - on the Solaris OS, 100
- increasing size of a mirrored data partition
 - on Linux, 110
 - on the Solaris OS, 99

pkill command, summary of, 6, 7

poweroff

- abrupt node halt
 - on Linux, 71
 - on the Solaris OS, 71
- abrupt node poweroff, on the Solaris OS, 70
- clean node poweroff
 - on Linux, 69
 - on the Solaris OS, 68

predefined node

- determining whether a cluster has them, 118
- summary of, 117

profile file, summary of, 11

ps command, summary of, 6

R

rdc.conf, reconfiguring the scoreboard bitmap, 59

reboot

- abrupt node reboot
 - on Linux, 70
 - on the Solaris OS, 70
- clean node reboot
 - on Linux, 68
 - on the Solaris OS, 68

reboot command, summary of, 6, 7

recovering a failed cluster, 77

redundant network, verifying existence of, 26

Reliable NFS

- logging NFS operations, 20
- verifying it is running, 28

ReliableBoot Service, verifying operation of, 28

- replacing a disk
 - on a dataless node, 132
 - on a vice-master node, 127
 - on master-eligible nodes without shutdown, 130
- replacing Ethernet cards
 - on a dataless node, 124
 - on a diskless node
 - with the DHCP dynamic boot policy, 125
 - with the DHCP static boot policy, 125
 - on a vice-master node node, 124
- replication, viewing the routing table, 50
- resize2fs command, growing a file system on
 - Linux, 112
- restarting
 - a node, 73
 - clusters, 74
 - nodes, 72
- restore backup
 - restore command, on Linux, 111
 - ufsrestore command, on the Solaris OS, 100
- restore command
 - restoring a backup on Linux, 111
 - summary of, 8
- role of a node, determining, 24
- route, summary of, 6, 7
- routing table, viewing, 50
- rpm command, summary of, 7

S

- scoreboard bitmap
 - changing location of, 59
 - configuration of, 58
- services
 - restarting, 65
 - stopping, 64
- services file, summary of, 11, 12
- shared file systems, impact to performance, 55
- shared files, managing, 55
- shared packages, patching
 - on Linux, 87
 - on the Solaris OS, 82
- shutting down
 - a cluster, 73
 - diskless node or dataless node, 72
 - master node, 71
 - nodes, 67

- vice-masternode, 72
- snoop command, debugging diskless nodes, 53
- Solaris configuration files modified for Netra HA
 - Suite software, 10
- Solaris Operating System
 - administration tools, 5
 - patching, 82
- strace command, summary of, 8
- switchover
 - nhcmmstat, 76
 - triggering, 76
- switchover, triggering, 76
- synchronization
 - of files, 60
 - verifying of masternode and vice-master
 - node, 26
- syslog.conf
 - configuring, 16
 - summary of, 11, 13
- system file
 - summary of, 11
 - synchronization of, 60
- system log files
 - accessing, 15
 - configuring system log files, 15
 - logging NFS operations, 20
 - maintaining, 18
 - message categories, 15
 - redirecting log message to files, 16
 - removing old log files, 20

T

- tools
 - adding tools to your path, 1
 - examining network configuration, 31
 - Foundation Services administration tools, 2
 - Linux administration tools, 7
 - Solaris administration tools, 5
- truss command, summary of, 6

U

- ufrestore, restoring a backup, 100
- ufsdump command
 - backing up mirrored data partition on physical
 - disk, 100
 - summary of, 6
- ufsrestore command, summary of, 6

V

`vfstab` command, summary of, 11, 12

vice-master node

- replacing a disk, 127

- shutting down, 72

- verifying the presence of, 24

virtual disk

- adding a mirrored data partition

 - on Linux, 114

 - on the Solaris OS, 104

- increasing the size of a mirrored data partition

 - on Linux, 111

