

Sun™ Cluster 3.0 Administration ES-333

Student Guide **With Instructor Notes**



Sun Microsystems, Inc.
UBRM05-104
500 Eldorado Blvd.
Broomfield, CO 80021
U.S.A.

July 2001, Revision B

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303, U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or 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.

iPlanet, JumpStart, OpenBoot, Solaris Operating Environment, Solaris Resource Manager, Solstice DiskSuite, Solaris Sun Enterprise 450, Sun, Sun Cluster, Sun Enterprise, Sun Enterprise 220R, Sun Enterprise 10000, Sun Fire, Sun Fireplane Interconnect, Sun Management Center, Sun Microsystems, the Sun Logo, SunPlex, SunSolve, Sun StorEdge, Sun StorEdge D1000, Sun StorEdge A5000, Sun StorEdge T3, and Sun StorEdge MultiPack, 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.

UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Netscape is a trademark or registered trademark of Netscape Communications Corporation in the United States and other countries.

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 approval might be required when exporting the product.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015 (b)(6/95) and DFAR 227.7202-3(a).

Display PostScript(TM) is a trademark or registered trademark of Adobe Systems, Incorporated. Display PostScript est une marque de fabrique d'Adobe Systems, Incorporated. in the United States and other countries.

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 2001 Sun Microsystems Inc., 901 San Antonio Road, Palo Alto, California 94303, Etats-Unis. Tous droits réservés.

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.

iPlanet, JumpStart, OpenBoot, Solaris Operating Environment, Solaris Resource Manager, Solstice DiskSuite, Solaris Sun Enterprise 450, Sun, Sun Cluster, Sun Enterprise, Sun Enterprise 220R, Sun Enterprise 10000, Sun Fire, Sun Fireplane Interconnect, Sun Management Center, Sun Microsystems, the Sun Logo, SunPlex, SunSolve, Sun StorEdge, Sun StorEdge D1000, Sun StorEdge A5000, Sun StorEdge T3, et Sun StorEdge MultiPack 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 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.

UNIX est une marques déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Netscape est une marque de Netscape Communications Corporation aux Etats-Unis et dans d'autres pays.

L'interfaces 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 licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

L'accord du gouvernement américain est requis avant l'exportation du produit.

PostScript(TM) is a trademark or registered trademark of Adobe Systems, Incorporated, which may be registered in certain jurisdictions. PostScript est une marque de fabrique d'Adobe Systems, Incorporated, laquelle pourrait être déposée dans certaines juridictions. in the United States and other countries.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" 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.



Please
Recycle



Adobe PostScript

Table of Contents

About This Course	xix
Course Goals.....	xix
Course Map.....	xx
Topics Not Covered.....	xxi
How Prepared Are You?.....	xxii
Introductions	xxiii
How to Use Course Materials	xxiv
Conventions.....	xxv
Icons	xxv
Typographical Conventions	xxvi
Sun™ Cluster Overview.....	1-1
Objectives	1-1
Relevance.....	1-2
Additional Resources	1-3
Overview of Sun™ Cluster 3.0 07/01	1-4
Sun Cluster 3.0 7/01 Features	1-4
Software Revisions.....	1-4
Tools.....	1-5
Hardware Support.....	1-6
Data Services.....	1-6
Cluster Hardware Components.....	1-7
Administrative Console	1-8
Terminal Concentrator	1-8
Cluster Host Systems.....	1-8
Cluster Transport Interface.....	1-8
Cluster Disk Storage	1-8
Sun Cluster High-Availability Features	1-9
High-Availability Hardware Design.....	1-9
Sun Cluster High-Availability Software.....	1-9
Software RAID Technology.....	1-9
Controller-based RAID Technology.....	1-9
Sun Cluster Data Service Support	1-10
Highly Available and Scalable Data Service Support	1-10

Parallel Database Support	1-10
High-Availability Strategies	1-11
Redundant Servers.....	1-12
Redundant Data	1-12
Redundant Public Network Interfaces	1-12
Redundant Transport Interface.....	1-12
Domain-based Clusters	1-13
Domain-based Cluster Configurations	1-14
Cluster in a Box	1-14
Sun Cluster High-Availability Failover	1-15
Failover Applications	1-15
Node Fault Monitoring	1-16
Network Fault Monitoring	1-16
Data Service Fault Monitoring.....	1-17
Cluster Configuration Repository	1-18
Sun Cluster Scalable Services	1-19
Disk ID Devices	1-20
Global Devices.....	1-21
Global Device Links.....	1-22
Cluster File Systems.....	1-23
Resource Groups	1-24
Parallel Database Application.....	1-25
Enabling Shared Data.....	1-26
Check Your Progress	1-27
Think Beyond	1-28
Terminal Concentrator	2-1
Objectives	2-1
Relevance.....	2-2
Additional Resources	2-3
Cluster Administration Interface.....	2-4
Cluster Administration Elements.....	2-5
Terminal Concentrator Overview	2-6
Operating System Load.....	2-7
Setup Port.....	2-7
Terminal Concentrator Setup Programs.....	2-7
Setting Up the Terminal Concentrator.....	2-8
Connecting to Port 1	2-8
Enabling Setup Mode	2-8
Setting the Terminal Concentrator IP Address.....	2-9
Setting the Terminal Concentrator Load Source	2-9
Specifying the Operating System Image	2-10
Setting the Serial Port Variables.....	2-11
Disabling Terminal Concentrator Routing.....	2-12
Creating a Terminal Concentrator Default Route	2-13
Using Multiple Terminal Concentrators.....	2-14

Terminal Concentrator Troubleshooting.....	2-15
Using the <code>telnet</code> Command to Manually Connect to a Node	2-15
Using the <code>telnet</code> Command to Abort a Node.....	2-15
Connecting to the Terminal Concentrator Command-Line Interpreter	2-16
Using the Terminal Concentrator <code>help</code> Command	2-16
Identifying and Resetting a Locked Port.....	2-17
Notes: Erasing Terminal Concentrator Settings	2-18
Exercise: Configuring the Terminal Concentrator	2-19
Preparation.....	2-19
Task – Verifying the Network and Host System Cabling.....	2-20
Task – Connecting a Local Terminal	2-21
Task – Connecting Tip Hardwire	2-22
Task – Achieving Setup Mode	2-23
Task – Configuring the IP Address	2-24
Task – Configuring the TC to Self-Load	2-25
Task – Verifying the Self-Load Process.....	2-26
Task – Verifying the TC Port Settings.....	2-27
Task – Terminal Concentrator Troubleshooting	2-28
Exercise Summary.....	2-29
Check Your Progress	2-30
Think Beyond	2-31
Installing the Administrative Console	3-1
Objectives	3-1
Relevance.....	3-2
Additional Resources	3-3
Sun Cluster Console Software.....	3-4
Console Software Installation	3-4
Sun Cluster Console Tools.....	3-5
The Cluster Control Panel	3-5
Cluster Console	3-6
Cluster Console Window Variations.....	3-8
Cluster Console Tools Configuration.....	3-9
Configuring the <code>/etc/clusters</code> File	3-9
Configuring the <code>/etc/serialports</code> File.....	3-10
Multiple Terminal Concentrator Configuration.....	3-11
Exercise: Configuring the Administrative Console	3-12
Preparation.....	3-12
Task – Updating Host Name Resolution.....	3-13
Task – Installing the Cluster Console Software	3-13
Task – Verifying the Administrative Console Environment	3-14
Task – Configuring the <code>/etc/clusters</code> File.....	3-14

Task – Configuring the /etc/serialports File	3-15
Task – Starting the cconsole Tool.....	3-15
Task – Using the ccp Control Panel.....	3-16
Exercise Summary.....	3-17
Check Your Progress	3-18
Think Beyond	3-19
Preinstallation Configuration.....	4-1
Objectives	4-1
Relevance.....	4-2
Additional Resources	4-3
Configuring Cluster Servers.....	4-4
Boot Device Restrictions	4-4
Server Hardware Restrictions	4-5
Configuring Cluster Topologies	4-6
Clustered Pairs Topology	4-6
Pair+N Topology.....	4-8
N+1 Topology.....	4-9
Sun Fire System Configurations	4-10
Sun Fire 4800 and Sun Fire 4810 Configuration	4-10
Sun Fire 6800 Configuration.....	4-11
Configuring Storage	4-12
Sun StorEdge MultiPack Configuration	4-12
Sun StorEdge D1000 System Configuration.....	4-13
Sun StorEdge A3500 System Configuration.....	4-14
Sun StorEdge A3500FC System Configuration	4-15
Sun StorEdge A5x00 System Configurations.....	4-16
Sun StorEdge T3 System Configurations.....	4-19
Cluster Interconnect Configuration	4-21
Point-to-Point Cluster Interconnect.....	4-21
Junction-based Cluster Interconnect	4-21
Cluster Transport Interface Addresses	4-22
Identifying Cluster Transport Interfaces	4-22
Eliminating Single Points of Failure.....	4-24
Cluster Quorum Device Configuration	4-25
Quorum Device Rules	4-26
Two-Node Cluster Quorum Devices	4-26
Clustered-Pair Quorum Disks.....	4-27
Pair+N Quorum Disks	4-28
N+1 Quorum Disks.....	4-29
Public Network Configuration	4-30
Identifying Storage Array Firmware.....	4-31
Identifying Attached Sun StorEdge A5x00 Storage Arrays.....	4-31
Identifying Host Bus Adapter Firmware.....	4-32

Identifying Sun StorEdge A5x00 Interface	
Board Firmware.....	4-33
Identifying Sun StorEdge T3 Array Firmware	4-34
Exercise: Preinstallation Preparation	4-35
Preparation.....	4-35
Task – Verifying the Solaris Operating Environment	4-36
Task – Identifying a Cluster Topology	4-36
Task – Selecting Quorum Devices	4-37
Task – Verifying the Cluster Interconnect	
Configuration	4-37
Task – Verifying Storage Array Firmware Revisions.....	4-39
Task – Selecting Public Network Interfaces	4-40
Exercise Summary.....	4-41
Check Your Progress	4-42
Think Beyond	4-43
Installing the Cluster Host Software.....	5-1
Objectives	5-1
Relevance.....	5-2
Additional Resources	5-3
Sun Cluster Software Summary.....	5-4
Sun Cluster Software Distribution	5-5
Sun Cluster Framework Software	5-6
Sun™ Cluster 3.0 07/01 Agents.....	5-7
Virtual Volume Management Software.....	5-7
Sun™ Cluster 3.0 07/01 Licensing	5-8
Sun Cluster Basic Installation Process	5-8
Sun Cluster Alternative Installation Processes.....	5-9
Configuring the Sun Cluster Node Environment	5-10
Configuring the User root Environment.....	5-10
Configuring Network Name Resolution	5-10
Installing Sun Cluster Node Patches.....	5-11
Patch Installation Warnings	5-11
Obtaining Patch Information.....	5-11
Installing the Sun™ Cluster 3.0 07/01 Software	5-12
Installing Sun Cluster Interactively.....	5-12
Postinstallation Configuration	5-19
Resetting Install Mode of Operation	5-19
Configuring Network Time Protocol.....	5-21
Postinstallation Verification	5-22
Verifying DID Devices	5-22
Verifying General Cluster Status.....	5-23
Verifying Cluster Configuration Information	5-24
Correcting Minor Configuration Errors	5-26
Exercise: Installing the Sun Cluster Server Software.....	5-27
Preparation.....	5-27

Task – Verifying the Boot Disk	5-27
Task – Verifying the Environment	5-28
Task – Updating the Name Service	5-28
Task – Establishing a New Cluster	5-29
Task – Adding a Second Cluster Node	5-31
Task – Configuring a Quorum Device	5-32
Task – Configuring the Network Time Protocol	5-33
Task – Configuring Host Name Resolution	5-33
Testing Basic Cluster Operation	5-34
Exercise Summary	5-35
Check Your Progress	5-36
Think Beyond	5-37
Basic Cluster Administration.....	6-1
Objectives	6-1
Relevance.....	6-2
Additional Resources	6-3
Cluster Status Commands	6-4
Checking Status Using the <code>scstat</code> Command.....	6-4
Checking Status Using the <code>sccheck</code> Command.....	6-4
Checking Status Using the <code>scinstall</code> Utility.....	6-5
Cluster Control.....	6-6
Starting and Stopping Cluster Nodes	6-6
Booting Nodes in Non-Cluster Mode	6-7
Placing Nodes in Maintenance Mode	6-8
Cluster Amnesia.....	6-8
Monitoring With the Sun Management Center	6-10
Sun MC Server Software	6-11
Sun MC Console Software	6-11
Sun MC Agent Software	6-11
Exercise: Performing Basic Cluster Administration	6-12
Preparation.....	6-12
Task – Verifying Basic Cluster Status.....	6-13
Task – Starting and Stopping Cluster Nodes.....	6-13
Task – Placing a Node in Maintenance State	6-14
Task – Recovering from Cluster Amnesia.....	6-15
Task – Booting Nodes in Non-cluster Mode.....	6-15
Exercise Summary.....	6-16
Check Your Progress	6-17
Think Beyond	6-18
Volume Management Using VERITAS Volume Manager.....	7-1
Objectives	7-1
Relevance.....	7-2
Additional Resources	7-3
Disk Space Management.....	7-4
VERITAS Volume Manager Disk Initialization.....	7-5

Private Region Contents	7-6
Private and Public Region Format.....	7-7
Initialized Disk Types.....	7-7
VERITAS Volume Manager Disk Groups	7-8
VERITAS Volume Manager Status Commands	7-9
Checking Volume Status.....	7-9
Checking Disk Group Status	7-10
Checking Disk Status.....	7-10
Saving Configuration Information	7-10
Optimizing Recovery Times.....	7-11
Dirty Region Logging.....	7-11
The Solaris Operating Environment UFS Logging	7-12
VERITAS Volume Manager Installation Overview	7-13
VERITAS Volume Manager Dynamic Multipathing.....	7-13
Installing the VERITAS Volume Manager Software.....	7-14
Verifying the vxio Driver Major Numbers.....	7-15
Setting Unique rootdg Minor Device Numbers.....	7-16
Initializing the rootdg Disk Group.....	7-17
Creating a Simple rootdg Disk Group.....	7-17
Encapsulating the System Boot Disk.....	7-18
Sun Cluster Boot Disk Encapsulation Process.....	7-21
Sun Cluster Manual Encapsulation Process.....	7-21
Sun Cluster Automated Encapsulation Process	7-23
Registering VERITAS Volume Manager Disk Groups.....	7-25
Device Group Policies	7-26
Exercise: Configuring Volume Management.....	7-27
Preparation.....	7-27
Task – Selecting Disk Drives	7-28
Task – Using scvxinstall to Install and Initialize VERITAS Volume Manager Software.....	7-28
Task – Manually Installing and Initializing VERITAS Volume Manager Software.....	7-30
Task – Optional rootdg Disk Group Initialization.....	7-34
Task – Configuring Demonstration Volumes.....	7-35
Task – Registering Demonstration Disk Groups.....	7-37
Task – Creating a Global nfs File System	7-38
Task – Creating a Global web File System	7-39
Task – Testing Global File Systems	7-40
Task – Managing Disk Device Groups	7-40
Exercise Summary.....	7-43
Check Your Progress	7-44
Think Beyond	7-45

Volume Management Using	
Solstice DiskSuite™	8-1
Objectives	8-1
Relevance	8-2
Additional Resources	8-3
Disk Space Management	8-4
Solstice DiskSuite Disk Space Management	8-4
Solstice DiskSuite Initialization	8-5
Replica Configuration Guidelines	8-6
Solstice DiskSuite Disk Grouping	8-7
Adding Disks to a Diskset	8-8
Dual-String Mediators	8-9
The Solaris Operating Environment UFS Logging	8-10
Solstice DiskSuite Status	8-11
Checking Volume Status	8-11
Checking Mediator Status	8-11
Checking Replica Status	8-12
Recording Solstice DiskSuite Configuration	
Information	8-13
Solstice DiskSuite Installation Overview	8-14
Solstice DiskSuite Postinstallation	8-15
Modifying the <code>md.conf</code> File	8-15
Enabling Solstice DiskSuite Node Access	8-16
Initializing Local State Database Replicas	8-16
Creating Disksets for the Data Services	8-16
Adding Disks to a Diskset	8-16
Configuring Metadevices	8-17
Configuring Dual-String Mediators	8-18
Exercise: Configuring Solstice DiskSuite	8-19
Preparation	8-19
Task – Installing the Solstice DiskSuite Software	8-20
Task – Initializing the Solstice DiskSuite State	
Databases	8-21
Task – Selecting the Solstice DiskSuite Demo	
Volume Disk Drives	8-22
Task – Configuring the Solstice DiskSuite	
Demonstration Disksets	8-23
Task – Configuring Solstice DiskSuite	
Demonstration Volumes	8-24
Task – Configuring Dual-String Mediators	8-25
Task – Creating a Global <code>nfs</code> File System	8-26
Task – Creating a Global <code>web</code> File System	8-27
Task – Testing Global File Systems	8-28
Task – Managing Disk Device Groups	8-29
Exercise Summary	8-30
Check Your Progress	8-31

Think Beyond	8-32
Public Network Management.....	9-1
Objectives	9-1
Relevance.....	9-2
Additional Resources	9-3
Public Network Management	9-4
Supported Public Network Interface Types.....	9-5
Global Interface Support.....	9-5
Configuring NAFO Groups.....	9-6
Pre-Configuration Requirements.....	9-6
Configuring Backup Groups	9-7
Modifying Existing PNM Configurations.....	9-8
PNM Status Commands.....	9-9
The <code>pnmstat</code> Command.....	9-9
The <code>pnmptor</code> Command.....	9-9
The <code>pnmrtop</code> Command.....	9-9
The PNM Monitoring Process.....	9-10
PNM Monitoring Routines	9-11
NAFO Group Status	9-11
PNM Parameters	9-12
Exercise: Configuring the NAFO Groups	9-13
Preparation.....	9-13
Task – Verifying EEPROM Status.....	9-14
Task – Creating a NAFO Group	9-14
Exercise Summary.....	9-15
Check Your Progress	9-16
Think Beyond	9-17
Resource Groups	10-1
Objectives	10-1
Relevance.....	10-2
Additional Resources	10-3
Resource Group Manager.....	10-4
Resource Types.....	10-5
Failover Data Service Resources	10-7
Scalable Data Service Resources	10-8
Sun Cluster Resource Groups	10-9
Configuring a Resource Group.....	10-9
Resource Group Components.....	10-10
Resource Group Administration	10-11
Resource Properties	10-12
Standard Resource Type Properties.....	10-13
Extended Resource Type Properties	10-13
Resource Properties	10-13
Resource Group Properties.....	10-14
Creating Resource Groups Using the <code>scsetup</code> Utility...	10-15

Check Your Progress	10-16
Think Beyond	10-17
Data Services Configuration	11-1
Objectives	11-1
Relevance.....	11-2
Additional Resources	11-3
Sun Cluster Data Service Methods.....	11-4
Data Service Methods.....	11-4
Data Service Fault Monitors	11-5
Sun Cluster High Availability for NFS Methods	11-5
Disk Device Group Considerations.....	11-6
The SUNW.HAStorage Resource Type.....	11-6
Guidelines for SUNW.HAStorage	11-7
Overview of Data Service Installation	11-8
Preparing for Data Service Installation.....	11-8
Installing and Configuring the Application Software	11-9
Installing the Sun Cluster Data Service	
Software Packages	11-9
Registering and Configuring a Data Service.....	11-9
Installing Sun Cluster HA for NFS.....	11-10
Preparing for Installation.....	11-10
Installing and Configuring the Application	
Software.....	11-11
Installing the Sun Cluster Data Service	
Software Packages	11-12
Registering and Configuring the Data Service	11-13
Testing NFS Failover	11-16
Installing Sun Cluster Scalable Service for Apache	11-18
Preparing for Installation.....	11-18
Installing and Configuring the Application	
Software.....	11-20
Testing the Application Software Installation	11-22
Installing the Sun Cluster Data Service	
Software Packages	11-24
Registering and Configuring the Data Service	11-25
Advanced Resource Commands.....	11-28
Advanced Resource Group Operations.....	11-28
Advanced Resource Operations	11-28
Advanced Fault Monitor Operations.....	11-29
Exercise: Installing and Configuring Sun Cluster HA	
for NFS.....	11-30
Preparation.....	11-30
Task – Preparing for Sun Cluster HA for NFS	
Data Service Configuration	11-31

Task – Registering and Configuring the Sun Cluster HA for NFS Data Service	11-33
Task – Verifying Access by NFS Clients	11-34
Task – Observing Sun Cluster HA for NFS Failover Behavior	11-35
Task – Removing the <code>nfs-rg</code> Resource Group	11-36
Task – Creating a Resource Group Using the <code>scsetup</code> Utility	11-37
Exercise: Installing and Configuring Sun Cluster Scalable Service for Apache	11-40
Preparation	11-40
Task – Preparing for HA-Apache Data Service Configuration	11-41
Task – Registering and Configuring the Sun Cluster HA for Apache Data Service	11-43
Task – Verifying Apache Web Server Access and Scalable Functionality	11-44
Task – Optional Resource Group Exercise	11-44
Exercise Summary	11-46
Check Your Progress	11-47
Think Beyond	11-48
Using SunPlex™ Manager	12-1
Objectives	12-1
Relevance	12-2
Additional Resources	12-3
SunPlex Manager Introduction	12-4
SunPlex Manager Configuration	12-5
SunPlex Manager Installation Overview	12-5
Logging In to SunPlex Manager	12-6
SunPlex Manager Initial Display	12-7
SunPlex Manager Device Tree	12-8
SunPlex Manager Resource Groups Actions	12-9
SunPlex Manager Administration Summary	12-10
Nodes Summary	12-10
Resource Groups Summary	12-10
Transports Summary	12-10
Global Devices	12-11
Quorum	12-11
Exercise: Configuring SunPlex Manager	12-12
Preparation	12-12
Task – Installing SunPlex Manager	12-12
Task – Navigating the Nodes Device Tree	12-13
Task – Navigating the Resource Groups Device Tree	12-13
Task – Navigating the Transports Device Tree	12-14
Task – Navigating the Global Devices Device Tree	12-14

Task – Navigating the Quorum Device Tree	12-14
Task – Removing a Resource Group	12-15
Task – Creating a Resource Group	12-16
Exercise Summary	12-18
Check Your Progress	12-19
Think Beyond	12-20
Sun Cluster Administration Workshop	13-1
Objectives	13-1
Relevance	13-2
Additional Resources	13-3
Sun Cluster Administration Workshop Introduction	13-4
System Preparation	13-4
Configuration Steps	13-5
Configuration Solutions	13-10
Exercise Summary	13-19
Check Your Progress	13-20
Think Beyond	13-21
Cluster Configuration Forms	A-1
Cluster and Node Names Worksheet	A-2
Cluster Interconnect Worksheet	A-3
Public Networks Worksheet	A-4
Local Devices Worksheet	A-5
Local File System Layout Worksheet	A-6
Disk Device Group Configurations Worksheet	A-7
Volume Manager Configurations Worksheet	A-8
Metadevices Worksheet	A-9
Failover Resource Types Worksheet	A-10
Failover Resource Groups Worksheet	A-11
Network Resource Worksheet	A-12
HA Storage Application Resources Worksheet	A-13
NFS Application Resources Worksheet	A-14
Scalable Resource Types Worksheet	A-15
Scalable Resource Groups Worksheet	A-16
Shared Address Resource Worksheet	A-17
Scalable Application Resource Worksheet	A-18
Configuring Multi-Initiator SCSI	B-1
Multi-Initiator Overview	B-2
Installing a Sun StorEdge MultiPack Enclosure	B-3
Installing a Sun StorEdge D1000 Disk Array	B-7
NVRAMRC Editor and NVEDIT Keystroke Commands	B-11

Sun Cluster Administrative Command Summary	C-1
Command Overview	C-2
The <code>scinstall</code> Utility.....	C-3
Command Formats	C-3
The <code>scconf</code> Command.....	C-5
Command Formats	C-5
Command Example	C-6
The <code>scsetup</code> Utility	C-7
Command Example	C-7
The <code>sccheck</code> Utility	C-8
Command Format.....	C-8
Command Example	C-8
The <code>scstat</code> Command.....	C-9
Command Format.....	C-9
Command Example	C-9
The <code>scgdevs</code> Utility	C-11
Command Example	C-11
The <code>scdidadm</code> Command	C-12
Command Formats	C-12
Command Example	C-13
The <code>scswitch</code> Command	C-14
Command Formats	C-14
The <code>scshutdown</code> Command.....	C-17
Command Format.....	C-17
Command Example	C-17
The <code>scrgadm</code> Command.....	C-18
Command Formats	C-18
The <code>prnmsset</code> Utility	C-20
Command Formats	C-20
The <code>prnmstat</code> Command.....	C-22
Command Formats	C-22
Command Examples	C-23
The <code>prnmptor</code> and <code>prnmrtop</code> Commands.....	C-24
Command Formats and Examples	C-24
Sun Cluster Node Replacement	D-1
Node Replacement Overview	D-1
Replacement Preparation.....	D-2
Logically Removing a Failed Node	D-2
Physically Replacing a Failed Node	D-6
Logically Adding a Replacement Node.....	D-7
Sun Cluster HA for Oracle Installation	E-1
Installation Process	E-1
Glossary	Glossary-1

About This Course

Course Goals

Upon completion of this course, you should be able to:

- Describe the major Sun™ Cluster components and functions
- Perform preinstallation configuration verification
- Configure the terminal concentrator
- Configure a cluster administrative console
- Install the Sun Cluster 3.0 07/01 software
- Configure Sun Cluster 3.0 07/01 quorum devices
- Create network adapter failover (NAFO) groups
- Install Sun Cluster 3.0 07/01 data service software
- Configure global file systems
- Configure a failover data service resource group
- Configure a scalable data service resource group
- Configure the Sun Cluster 3.0 07/01 High-Availability (HA) for NFS failover data service
- Configure the Sun Cluster 3.0 07/01 HA for Apache scalable data service
- Use Sun Cluster administration tools

Course Map

The following course map enables you to see what you have accomplished and where you are going in reference to the course goals.

Product Introduction

Sun™ Cluster
Overview

Installation

Terminal
Concentrator

Installing the
Administrative
Console

Preinstallation
Configuration

Installing the
Cluster Host
Software

Operation

Basic
Cluster
Administration

Customization

Volume
Management
Using VERITAS™
Volume Manager

Volume
Management
Using Solstice
DiskSuite™

Public Network
Management

Resource
Groups

Data
Services
Configuration

Using
SunPlex™
Manager

Workshop

Sun Cluster
Administration
Workshop

Topics Not Covered

This course does not cover the topics shown on the overhead. Many of the topics listed on the overhead are covered in other courses offered by Sun Educational Services:

- Database installation and management – Covered in database vendor courses
- Network administration – Covered in SA-389: *Solaris™ 8 Operating Environment – TCP/IP Network Administration*
- Solaris™ Operating Environment administration – Covered in SA-238: *Solaris™ 8 Operating Environment System Administration I* and SA-288: *Solaris™ 8 Operating Environment System Administration II*
- Disk storage management – Covered in ES-220: *Disk Management With DiskSuite* and ES-310: *Volume Manager with Sun StorEdge™*

Refer to the Sun Educational Services catalog for specific information and registration.

How Prepared Are You?

To be sure you are prepared to take this course, can you answer yes to the following questions?

- Can you explain virtual volume management terminology, such as mirroring, striping, concatenation, volumes, and mirror synchronization?
- Can you perform basic Solaris™ Operating Environment administration tasks, such as using `tar` and `ufsdump` commands, creating user accounts, formatting disk drives, using `vi`, installing the Solaris Operating Environment, installing patches, and adding packages?
- Do you have prior experience with Sun hardware and the OpenBoot™ programmable read-only memory (PROM) technology?
- Are you familiar with general computer hardware, electrostatic precautions, and safe handling practices?

Introductions

Now that you have been introduced to the course, introduce yourself to each other and the instructor, addressing the item shown in the bullets below.

- Name
- Company affiliation
- Title, function, and job responsibility
- Experience related to topics presented in this course
- Reasons for enrolling in this course
- Expectations for this course

How to Use Course Materials

To enable you to succeed in this course, these course materials use a learning model that is composed of the following components:

- **Goals** – You should be able to accomplish the goals after finishing this course and meeting all of its objectives.
- **Objectives** – You should be able to accomplish the objectives after completing a portion of instructional content. Objectives support goals and can support other higher-level objectives.
- **Lecture** – The instructor will present information specific to the objective of the module. This information should help you learn the knowledge and skills necessary to succeed with the activities.
- **Activities** – The activities take on various forms, such as an exercise, self-check, discussion, and demonstration. Activities help to facilitate mastery of an objective.
- **Visual aids** – The instructor might use several visual aids to convey a concept, such as a process, in a visual form. Visual aids commonly contain graphics, animation, and video.

Conventions

The following conventions are used in this course to represent various training elements and alternative learning resources.

Icons



Additional resources – Indicates other references that provide additional information on the topics described in the module.



Discussion – Indicates a small-group or class discussion on the current topic is recommended at this time.



Note – Indicates additional information that can help students but is not crucial to their understanding of the concept being described. Students should be able to understand the concept or complete the task without this information. Examples of notational information include keyword shortcuts and minor system adjustments.



Caution – Indicates that there is a risk of personal injury from a nonelectrical hazard, or risk of irreversible damage to data, software, or the operating system. A caution indicates that the possibility of a hazard (as opposed to certainty) might happen, depending on the action of the user.



Warning – Indicates that either personal injury or irreversible damage of data, software, or the operating system will occur if the user performs this action. A warning does not indicate potential events; if the action is performed, catastrophic events will occur.

Typographical Conventions

Courier is used for the names of commands, files, directories, programming code, and on-screen computer output; for example:

```
Use ls -al to list all files.  
system% You have mail.
```

Courier is also used to indicate programming constructs, such as class names, methods, and keywords; for example:

```
The getServletInfo method is used to get author information.  
The java.awt.Dialog class contains Dialog constructor.
```

Courier **bold** is used for characters and numbers that you type; for example:

```
To list the files in this directory, type:  
# ls
```

Courier **bold** is also used for each line of programming code that is referenced in a textual description; for example:

```
1 import java.io.*;  
2 import javax.servlet.*;  
3 import javax.servlet.http.*;  
Notice the javax.servlet interface is imported to allow access to its  
life cycle methods (Line 2).
```

Courier italic is used for variables and command-line placeholders that are replaced with a real name or value; for example:

```
To delete a file, use the rm filename command.
```

Courier italic **bold** is used to represent variables whose values are to be entered by the student as part of an activity; for example:

```
Type chmod a+rw filename to grant read, write, and execute  
rights for filename to world, group, and users.
```

Palatino italic is used for book titles, new words or terms, or words that you want to emphasize; for example:

```
Read Chapter 6 in the User's Guide.  
These are called class options.
```

Sun™ Cluster Overview

Objectives

Upon completion of this module, you should be able to:

- List the new Sun™ Cluster 3.0 07/01 features
- List the hardware elements that constitute a basic Sun Cluster system
- List the hardware and software components that contribute to the availability of a Sun Cluster system
- List the types of redundancy that contribute to the availability of a Sun Cluster system
- Identify the functional differences between failover, scalable, and parallel database cluster applications
- List the supported Sun™ Cluster 3.0 07/01 data services
- Explain the purpose of Disk ID (DID) devices
- Describe the relationship between system devices and the cluster global namespace
- Explain the purpose of resource groups in the Sun Cluster environment
- Describe the purpose of the cluster configuration repository
- Explain the purpose of each of the Sun Cluster fault monitoring mechanisms

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- Which can contribute more to system availability, hardware or software?
- Why is a highly available system usually more practical than a fault-tolerant system?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Overview of Sun™ Cluster 3.0 07/01

This section describes the Sun™ Cluster 3.0 07/01 software release.

Sun Cluster 3.0 7/01 Features

The following describes the features included in Sun™ Cluster 3.0 07/01:

- VERITAS™ Volume Manager with Solstice DiskSuite™ root mirroring

You can now use VERITAS Volume Manager and Solstice DiskSuite on the same cluster nodes. You can use Solstice DiskSuite to mirror the cluster boot disk and use VERITAS Volume Manager for application data volumes.

- Solaris Resource Manager™ 1.2

Solaris Resource Manager 1.2 can now coexist with the Sun Cluster 3.0 framework.

- Improved Cluster File System performance

The Cluster File System input/output (I/O) performance has been significantly improved.

Software Revisions

Currently, the following software versions are supported:

- All Solaris 8 Operating Environment releases up to Update 4
The latest release has fewer software patch requirements.
- Sun™ Management Center 3.0
- VERITAS Volume Manager 3.1 and 3.1.1 support

Note – Consult the current Sun Cluster release notes for more up-to-date product support information.



Tools

The following describes the additional tools included in Sun™ Cluster 3.0 07/01:

- SunPlex™ Manager

SunPlex Manager is a comprehensive Sun Cluster graphical interface. You can use it to install the Sun Cluster software, Solstice DiskSuite software, and High Availability (HA) for NFS and HA for Apache data services. After the initial Sun Cluster installation, you use SunPlex Manager to perform general cluster administration.

- The `scvxinstall` script

You can use the `scvxinstall` script to install VERITAS Volume Manager and encapsulate the system boot disk. You can choose installation only or installation plus boot disk encapsulation. However, there are a number of prerequisites that you must do before running the script.

- `scsetup` utility enhancements

The `scsetup` utility is a menu-based interface that automates the process of configuring VERITAS Volume Manager disk groups and quorum disks in the Sun Cluster 3.0 environment. With the Sun™ Cluster 3.0 07/01 release, a set of extensions have been added to the `scsetup` utility to allow you to create and administer data service resource groups.

The following subjects are not presented in this course. They are advanced subjects that are available only in internal training courses.

- Sun Remote Support Interface - The Sun Remote Support Interface furnishes remote support capabilities for cluster nodes. You must contract for the remote support service. The Sun Remote Support interface is implemented as a Sun Management Center 3.0 agent.
- Custom Data Service Library routines - The custom data service library routines are used by the new SunPlex agent building software when creating custom data service agents.
- SunPlex Agent Builder - The agent builder is a tool that automates the creation of custom data services that run under the Resource Group Manager framework of the Sun Cluster software. The agent builder relies on the custom data service library routines.

Hardware Support

The following lists the additional hardware supported by Sun™ Cluster 3.0 07/01:

- Sun Fire™ systems

Support has been added for the Sun Fire 4800, Sun Fire 4810, and Sun Fire 6800 systems. Clustering within a single box is supported using these systems.

- Gigabit Ethernet

Currently, Gigabit Ethernet is supported for use in the cluster interconnect but not for public networks.

- Sun StorEdge™ A3500FC arrays

- Sun StorEdge™ T3 array support

Because Sun StorEdge T3 arrays have only a single host connection, you must use a minimum of two Sun StorEdge T3 arrays connected through fiber-channel hubs or switches. The Sun StorEdge Network Fibre-Channel switch is supported running in transparent mode. Data must be mirrored across two arrays using a software-based redundant array of independent disks (RAID) manager, such as VERITAS Volume Manager or Solstice DiskSuite.



Note – Currently, Sun StorEdge T3 partner-pair configuration are supported in the Sun™ Cluster 3.0 07/01 environment but only for specific combinations of system models, host bus adapters, virtual volume managers, software, and firmware. Consult the most current Sun Cluster release notes for up-to-date product support information.

Data Services

The following lists the additional data services supported by Sun™ Cluster 3.0 07/01:

- HA-SAP
- HA-Sybase

Cluster Hardware Components

The minimum hardware components that are necessary for most cluster configurations include:

- One administrative console
- One terminal concentrator
- Two hosts (eight maximum for most configurations)
- One or more public network interfaces per system (not shown)
- A private cluster transport interface
- Dual hosted, mirrored disk storage

Figure 1-1 shows the physical structure of a minimum cluster.

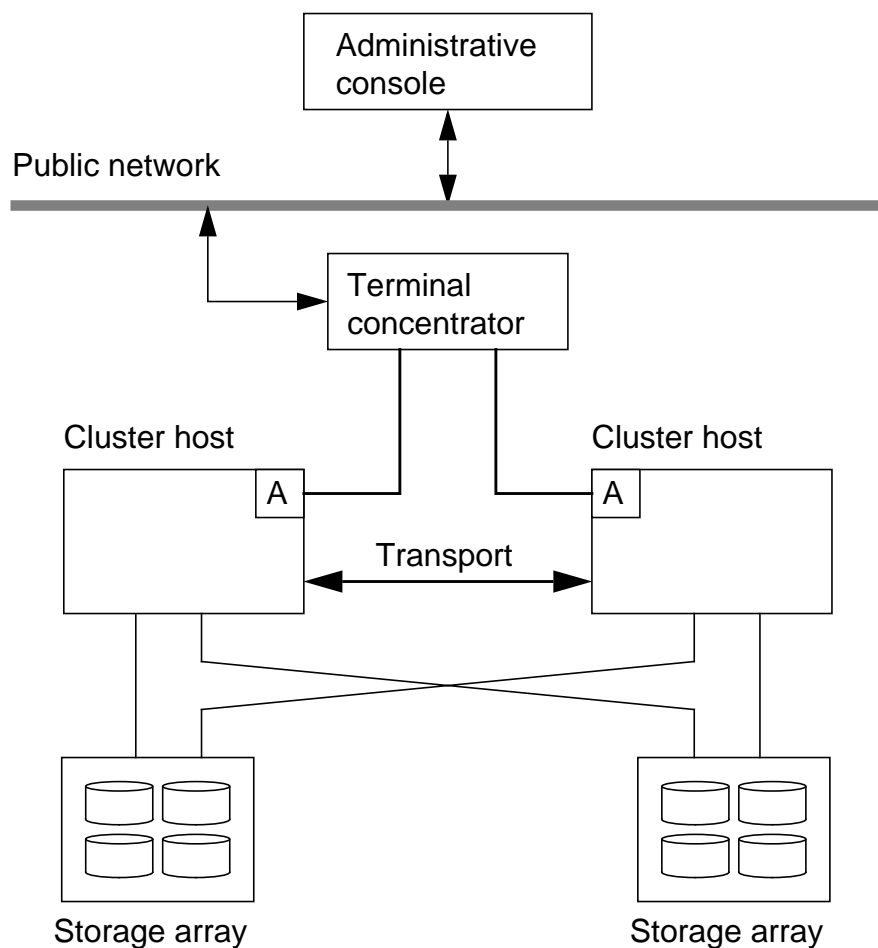


Figure 1-1 Cluster Physical Connections

Administrative Console

The administrative console can be any Sun workstation, providing it has adequate resources to support graphics and compute-intensive applications. You can use cluster administration tools to monitor many clusters from one administrative console.

Terminal Concentrator

The Sun terminal concentrator (TC) provides data translation from the network to serial port interfaces. Each of the serial port outputs connects to a separate node in the cluster through serial port A. Because the nodes commonly do not have frame buffers, this is the only access path when the operating system is down.

Cluster Host Systems

A wide range of Sun hardware platforms are supported for use in the clustered environment. Mixed platform clusters are not supported.

Cluster Transport Interface

All nodes in a cluster are linked by a private cluster transport. The transport is redundant and can be used for the following purposes:

- Cluster-wide monitoring and recovery
- Parallel database lock and query information
- Global data access

Cluster Disk Storage

The Sun Cluster environment can use several Sun storage models. They must all accept dual-host connections. The Sun StorEdge T3 arrays have a single connection, so they must be used with hubs or switches.

Note – Although some storage array models can physically accept more than two host system connections, the Sun™ Cluster 3.0 07/01 release supports a maximum of two host connections to a storage array.



Sun Cluster High-Availability Features

The Sun Cluster system focuses on providing reliability, availability, and scalability. Part of the reliability and availability is inherent in the cluster hardware and software.

High-Availability Hardware Design

Many of the supported cluster hardware platforms have the following features that contribute to maximum uptime:

- Hardware that is interchangeable between models
- Redundant system board power and cooling modules
- Systems that contain automatic system reconfiguration; failed components, such as the central processing unit (CPU), memory, and I/O, can be disabled at reboot
- Several disk storage options support hot-swapping of disks

Sun Cluster High-Availability Software

The Sun Cluster software has monitoring and control mechanisms that can initiate various levels of cluster reconfiguration to help maximize application availability.

Software RAID Technology

The VERITAS Volume Manager and Sun Solstice DiskSuite software provide RAID protection in redundant mirrored volumes.

Controller-based RAID Technology

The Sun StorEdge T3, Sun StorEdge A3500 and Sun StorEdge A3500FC models use controller-based RAID technology that is sometimes referred to as hardware RAID.

The Sun StorEdge T3 management and configuration programs are different than those used by the StorEdge A3500 and Sun StorEdge A3500FC models.

Sun Cluster Data Service Support

Each of the Sun Cluster data services provides a control and monitoring framework that enables a standard application to be highly available or scalable.

You can configure some of the data services for either failover or scalable operation.

Highly Available and Scalable Data Service Support

The Sun Cluster software provides preconfigured components that support the following HA data services:

- Sun Cluster HA for Oracle (failover)
- Sun Cluster HA for iPlanet™ (failover or scalable)
- Sun Cluster HA for Netscape Directory Server (failover)
- Sun Cluster HA for Apache (failover or scalable)
- Sun Cluster HA for Domain Name Service (DNS, failover)
- Sun Cluster HA for NFS (failover)
- Sun Cluster HA for SAP (failover)
- Sun Cluster HA for Sybase (failover)

Parallel Database Support

There is also Sun Cluster data service software that provides support for the Oracle Parallel Server (OPS) database application.

High-Availability Strategies

To help provide the level of system availability required by many customers, the Sun Cluster system uses the following strategies:

- Redundant servers
- Redundant data
- Redundant public network access
- Redundant private communications (transport)
- Multi-host storage access

Figure 1-2 shows the location and relationship of the high-availability strategies.

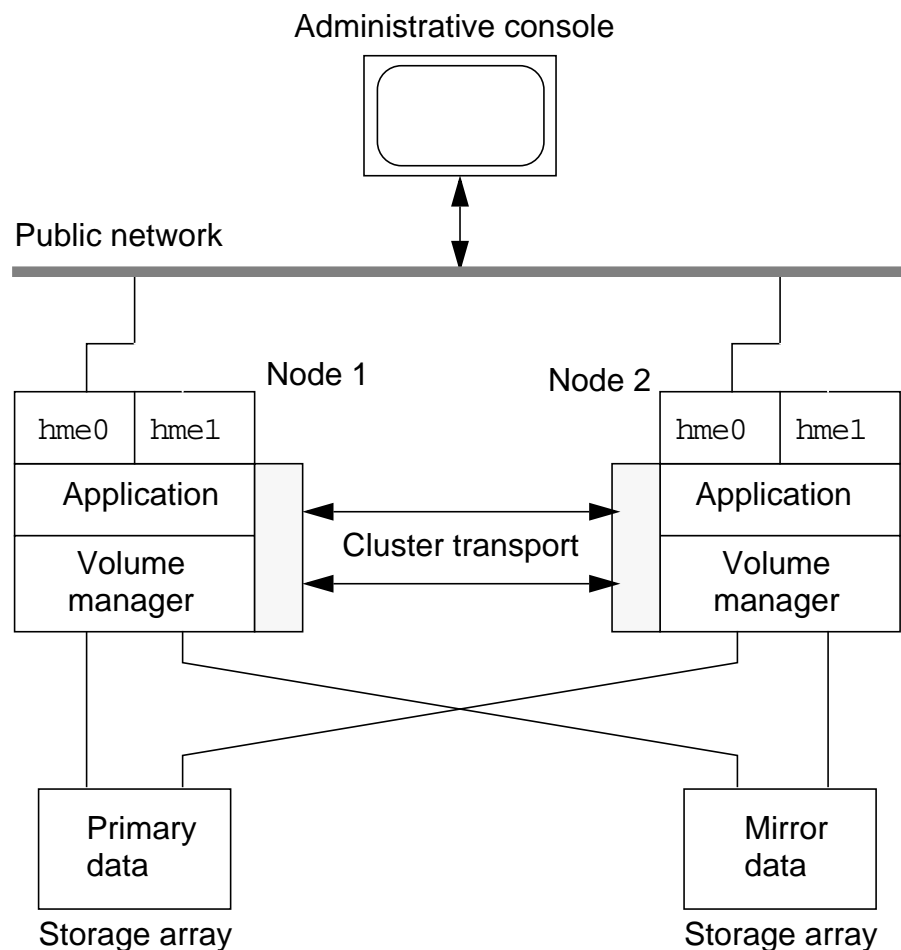


Figure 1-2 Cluster Availability Strategies

Redundant Servers

The Sun Cluster system consists of two to eight interconnected systems that are referred to as cluster host systems or nodes. The systems can be any of a range of Sun platforms and use off-the-shelf non-proprietary hardware.

The N+1 cluster configuration supports a maximum of four nodes.



Note – You cannot mix systems that use peripheral component interconnect (PCI) bus technology, such as the Sun Enterprise™ 450 server, with SBus technology systems, such as the Sun Enterprise 3500 server.

Redundant Data

A Sun Cluster system can use either VERITAS Volume Manager or Solstice DiskSuite to provide data redundancy. The use of data mirroring provides a backup in the event of a disk drive or storage array failure.

Redundant Public Network Interfaces

The Sun Cluster system provides a proprietary feature, public network management (PNM), that can transfer user I/O from a failed network interface to a predefined backup interface. The switch to the backup interface is transparent to cluster applications and users.

Redundant Transport Interface

The cluster transport interface consists of two to six high-speed private node-to-node communication interfaces. The cluster software uses all of the interfaces concurrently except for Oracle Parallel Server traffic.

If an interface fails, this is transparent to cluster applications.

Domain-based Clusters

You can structure the Sun Enterprise™ 10000 and Sun Fire lines of systems into separate resource domains within a single system. As shown in Figure 1-3, each domain is constructed using a portion of the system's available resources. Typically, a domain consists of at least one system processor board and one I/O board. The domains function independently of one another. Each domain run its own Solaris Operating Environment.

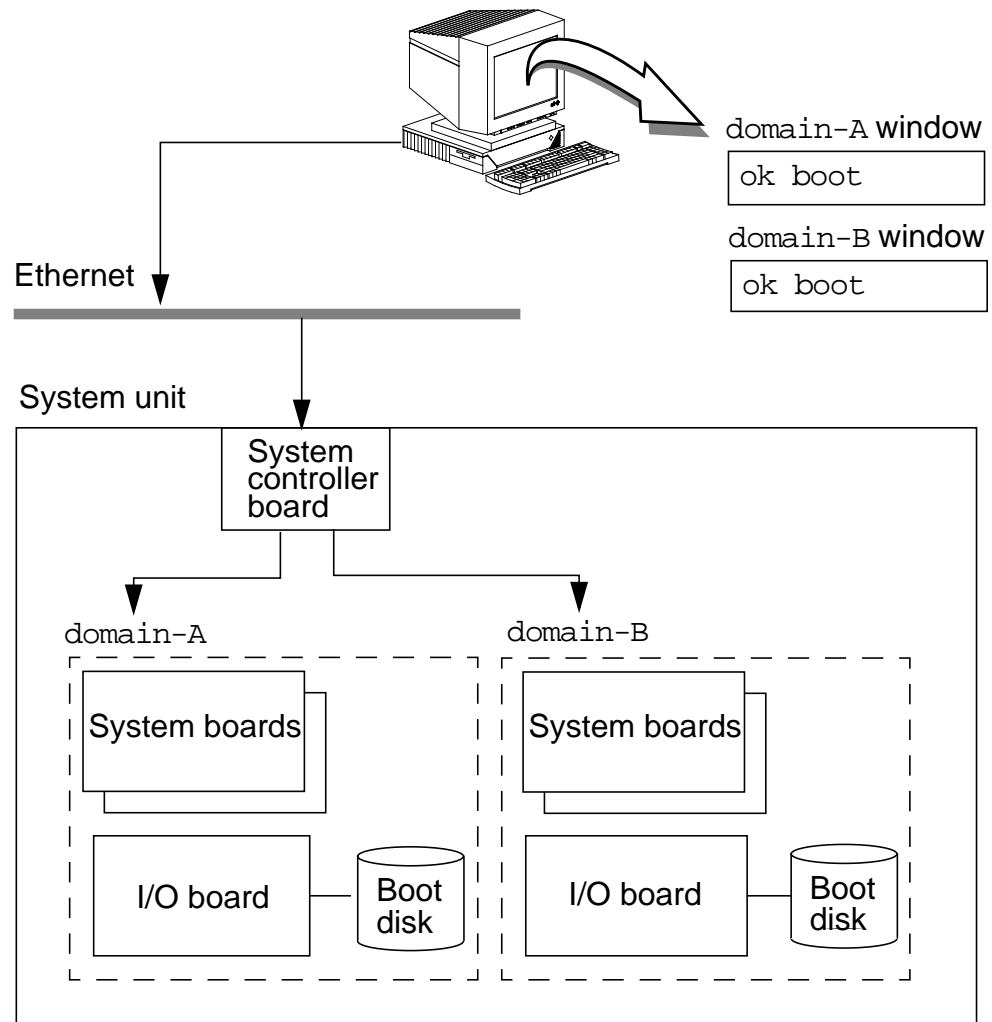


Figure 1-3 Sun Cluster Domain Strategy

Domain-based Cluster Configurations

Sun Cluster software can create and manage clusters that are constructed from domains. As shown in Figure 1-4, if possible, you should create clusters from domains that are in physically separate systems. If a cluster is created using domains that reside in a single system, commonly referred to as a *cluster-in-a-box* configuration, the system becomes a single point of failure that can cause the entire cluster to become unavailable.

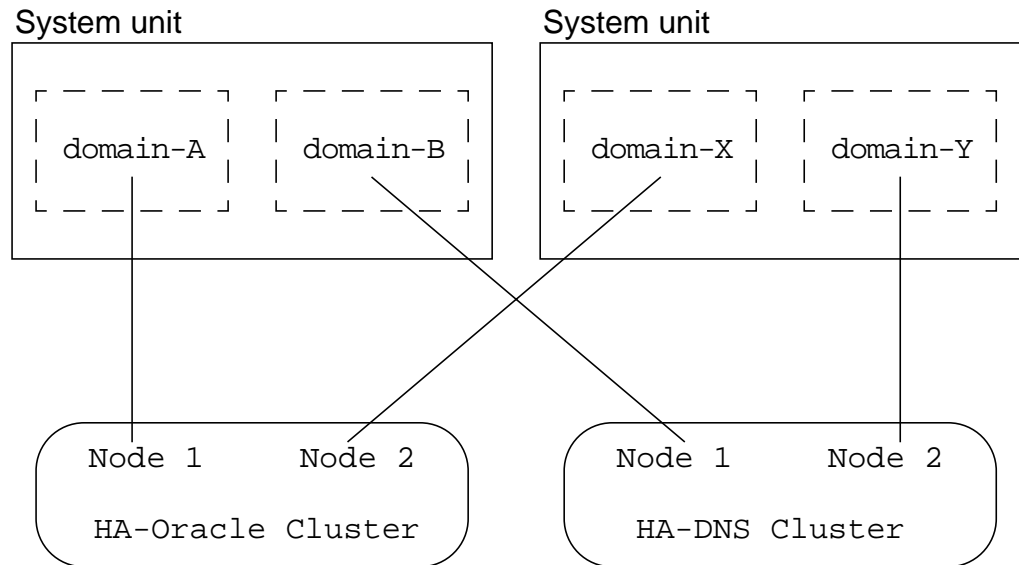


Figure 1-4 Domain Clustering

Cluster in a Box

Clustering in a box is supported using the Sun Enterprise 10000 or Sun Fire 4800, Sun Fire 4810, or Sun Fire 6800 systems. The Sun Fire 4800 and Sun Fire 4810 models have a single power grid that furnishes power to all boards. The single power grid is a single point of failure.

The Sun Fire 6800 system is an ideal candidate for a cluster-in-a-box configuration, because it has separate power grids for the even-numbered and odd-numbered board slots. Each half of the board slots have a unique power path, including separate power cords. This provides a higher level of reliability.

Note – Check current Sun Cluster product release notes for up-to-date system model support.



Sun Cluster High-Availability Failover

The Sun Cluster high-availability failover features include the following:

- Failover applications
- Node fault monitoring
- Network fault monitoring
- Data service fault monitoring

Failover Applications

As shown in Figure 1-5, a failover application (data service) runs on a single cluster node. If there is a node failure, a designated backup node takes over the application that was running on the failed node.

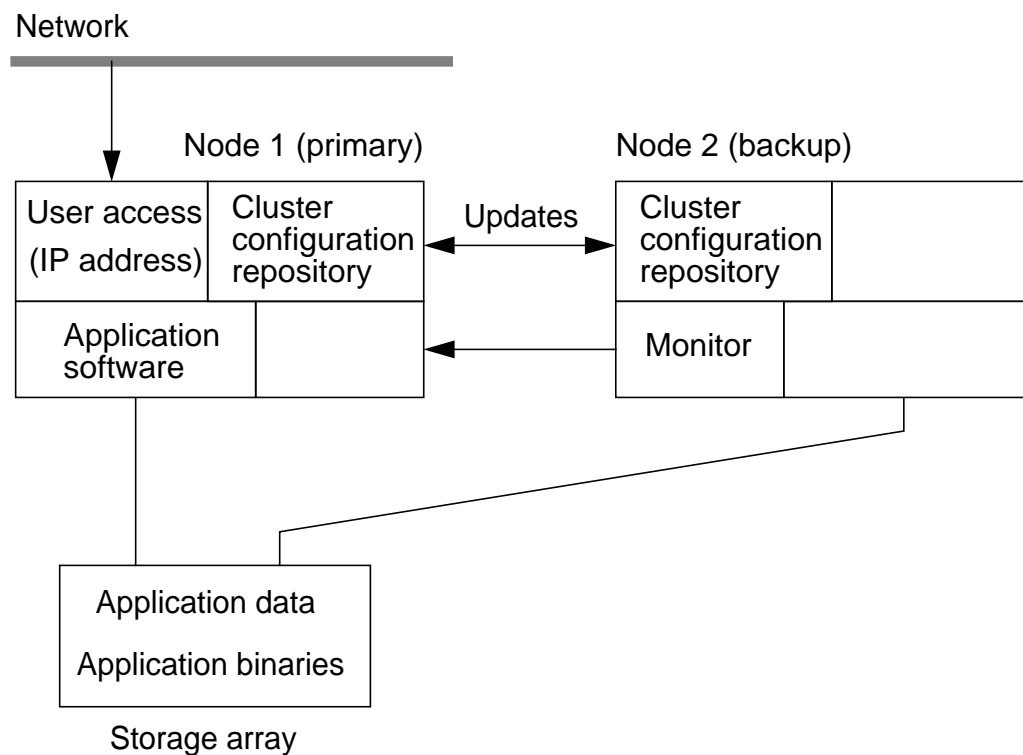


Figure 1-5 Failover Data Service Features

Node Fault Monitoring

The cluster membership monitor (CMM) is kernel-resident on each node and detects major cluster status changes, such as loss of communication between one or more nodes. The CMM instances communicate with one another across the private high-speed interconnect network interfaces by sending regular heartbeat messages. If the heartbeat from any node is not detected within a defined time-out period, it is considered as having failed and a cluster reconfiguration is initiated to renegotiate cluster membership.



Note – The cluster membership negotiation process is relatively complex and is described in Module 4, “Preinstallation Configuration.”

Network Fault Monitoring

Both the public network adapter failover (NAFO) interfaces and the cluster transport interfaces are monitored for potential failures.

Public Network Management

The PNM daemon, `pnmd`, monitors the functionality of NAFO group network interfaces and can transparently switch to backup interfaces in the event of a failure.

Cluster Transport Monitoring

The cluster transport interfaces are monitored on each node. If an active cluster transport interface on any node is determined to be inoperative, all nodes route interconnect traffic to functional transport interfaces. The failure is transparent to Sun Cluster applications.

Data Service Fault Monitoring

Each data service supplied by Sun, such as HA for NFS, has predefined fault monitoring routines associated with it. When a resource group is brought online with its associated resources, the resource group management (RGM) software automatically starts the appropriate fault monitoring processes. The data service fault monitors are referred to as *probes*.

The fault monitor probes verify that the data service is functioning correctly and providing its intended service. Typically, data service fault monitors perform two functions:

- Monitoring for the abnormal exit of data service processes
- Checking the health of the data service

Data Service Process Monitoring

The Process Monitor Facility (PMF) monitors the data service process. When an abnormal exit occurs, the PMF invokes an action script supplied by the data service to communicate the failure to the data service fault monitor. The probe then updates the status of the data service as “Service daemon not running” and takes action. The action can involve just restarting the data service locally or failing over the data service to a secondary cluster node.

Data Service Health Monitoring

A health monitoring probe typically behaves as a data service client and performs regular tests. Each data service requires different types of testing. For example, to test the health of the HA for NFS data service, its health probes check to ensure that the exported file systems are available and functional. The HA for DNS health monitor regularly uses the `nslookup` command to query the naming service for host or domain information.

Cluster Configuration Repository

General cluster configuration information is stored in global configuration files collectively referred to as the cluster configuration repository (CCR). The CCR must be kept consistent between all nodes and is a critical element that enables each node to be aware of its potential role as a designated backup system.



Caution – Never attempt to modify any of the CCR-related files. The files contain timestamp and checksum information that is critical to the operation of the cluster software. The CCR information is automatically modified as the result of administrative command execution and cluster status changes.

The CCR structures contain the following types of information:

- Cluster and node names
- Cluster transport configuration
- The names of VERITAS disk groups
- A list of nodes that can master each disk group
- Data service operational parameter values (timeouts)
- Paths to data service control routines
- Disk ID device configuration
- Current cluster status

The CCR is accessed when error or recovery situations occur or when there has been a general cluster status change, such as a node leaving or joining the cluster.

Sun Cluster Scalable Services

The Sun Cluster scalable services rely on the following components:

- Disk ID (DID) devices
- Global devices
- Global device links
- Cluster file systems (global file services)

A scalable data service application is designed to distribute an application workload between two or more cluster nodes. As shown in Figure 1-6, the same Web page server software is running on all nodes in a cluster.

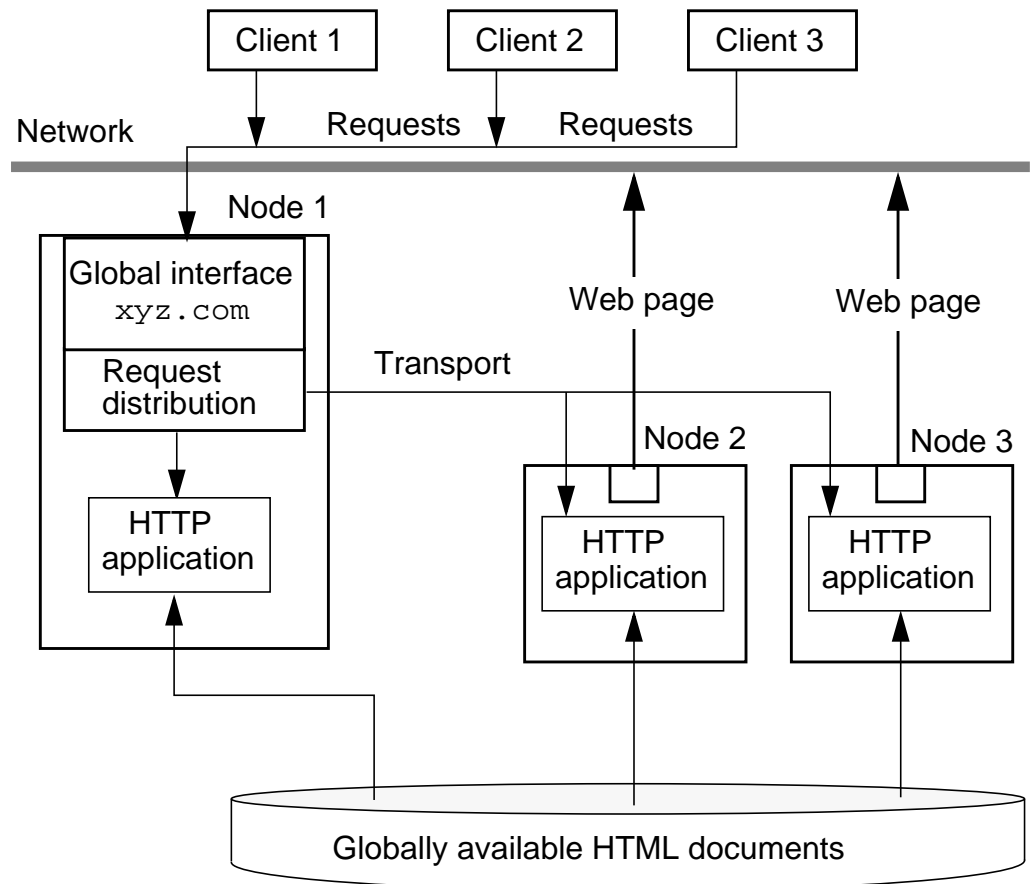


Figure 1-6 Scalable Data Service Features

A scalable data service configuration operates as follows:

- A designated node receives all requests and distributes them among the other nodes using the private cluster transport.
- The cluster nodes answer incoming Web page requests in parallel.
- If a node crashes, the cluster framework keeps the data service running with the remaining nodes.
- A single copy of the Hypertext Markup Language (HTML) documents can be placed on a globally accessible cluster file system.

Disk ID Devices

An important component of the Sun Cluster global device technology is the Disk ID (DID) pseudo driver. During the Sun Cluster installation, the DID driver probes devices on each node and creates a unique DID device name for each disk or tape device.

As shown in Figure 1-7, a disk drive in a storage array can have a different logical access path from attached nodes but is globally known by a single DID device number. A third node that has no array storage still has unique DID devices assigned for its boot disk and CD-ROM.

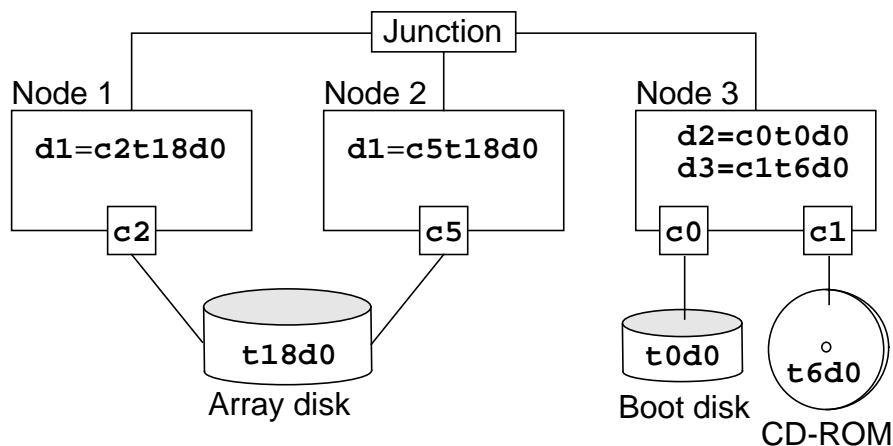


Figure 1-7 DID Driver Devices

The DID device driver creates the following DID instance names for the devices shown in Figure 1-7:

- /dev/did/rdisk/d1
- /dev/did/rdisk/d2
- /dev/did/rdisk/d3

Global Devices

Sun™ Cluster 3.0 07/01 uses *global devices* to provide cluster-wide, highly available access to any device in a cluster, from any node, without regard to where the device is physically attached. In general, if a node fails while providing access to a global device, Sun™ Cluster 3.0 07/01 automatically discovers another path to the device and redirects the access to that path. Sun™ Cluster 3.0 07/01 global devices include disks, CD-ROMs, and tapes. However, disks are the only supported multiported global devices.

The Sun™ Cluster 3.0 07/01 mechanism that enables global devices is the *global namespace*. The global namespace includes the `/dev/global/` hierarchy as well as the volume manager namespace. The global namespace reflects both multihost disks and local disks (and any other cluster device, such as CD-ROMs and tapes), and provides multiple failover paths to the multihost disks. Each node physically connected to multihost disks provides a path to the storage for any node in the cluster.

In Sun™ Cluster 3.0 07/01, each of the local device nodes, including volume manager namespace, are replaced by symbolic links to device nodes in the `/global/.devices/node@nodeID` file system, where *nodeID* is an integer that represents a node in the cluster (*node1*, *node2*, *node3*). Sun™ Cluster 3.0 07/01 continues to present the volume manager devices as symbolic links in their standard locations. The global namespace is available from any cluster node.

Typical global namespace relationships for a *nodeID* of 1 are shown in Table 1-1. They include a standard disk device, a DID disk device, a VERITAS Volume Manager volume and a Solstice DiskSuite metadvice.

Table 1-1 Global Namespace

Local Node Namespace	Global Namespace
<code>/dev/dsk/c0t0d0s0</code>	<code>/global/.devices/node@1/dev/dsk/c0t0d0s0</code>
<code>/dev/did/dsk/d0s0</code>	<code>/global/.devices/node@1/dev/did/dsk/d0s0</code>
<code>/dev/md/nfs/dsk/d0</code>	<code>/global/.devices/node@1/dev/md/nfs/dsk/d0</code>
<code>/dev/vx/dsk/nfs/v0</code>	<code>/global/.devices/node@1/dev/vx/dsk/nfs-dg/v0</code>

Global Device Links

In the Sun Cluster global device environment, each node can essentially access all of the devices attached to any other cluster node. Even a node with no local data storage can access another node's storage through the cluster interconnect.

- All nodes can view the `/global/.devices` file system mounts of other nodes.

```
# mount |grep /global/.devices
/global/.devices/node@1 on /dev/vx/dsk/rootdisk14vol
/global/.devices/node@2 on /dev/vx/dsk/rootdisk24vol
/global/.devices/node@3 on /dev/vx/dsk/rootdisk34vol
...
...
/global/.devices/node@8 on /dev/vx/dsk/rootdisk84vol
```

- The local `/dev` and `/devices` structures on each node are linked or copied into that node's global structure.

Figure 1-8 shows the basic global device structure. The paths shown do not represent the actual path details but accurately depict the concepts.

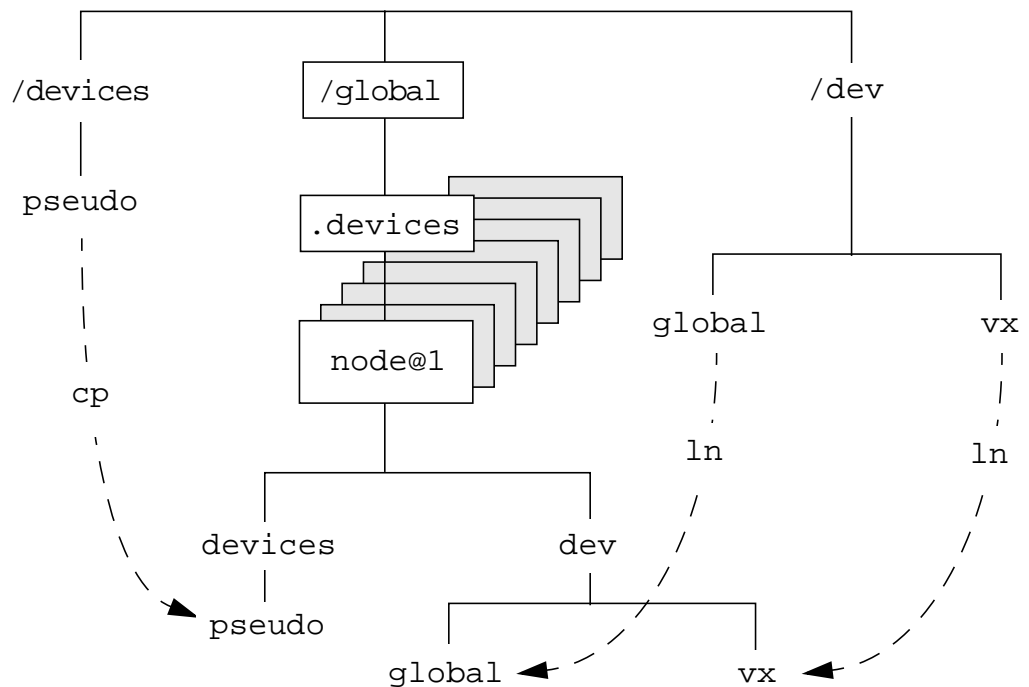


Figure 1-8 Global Device Links

Cluster File Systems

Cluster file systems depend on global devices (disks, tapes, CD-ROMs) with physical connections to one or more nodes.

To make a globally available file system, create a standard file system on a virtual volume (VERITAS or Solstice DiskSuite), and mount the volume on a mount point in the `/global` directory using special global mount options. A typical mount command is as follows:

```
# mount -g dev/vx/dsk/nfs-dg/vol-01 /global/nfs
```

The equivalent mount entry in the `/etc/vfstab` file is:

```
/dev/vx/dsk/nfs-dg/vol-01 /dev/vx/rdisk/nfs-dg/vol-01 \
/global/nfs ufs 2 yes global,logging
```

After the file system is mounted, it is available on all nodes in the cluster.

Note – You must register VERITAS Volume Manager disk groups with the cluster framework software before you can make any disk group structures globally available.



Proxy File System

The cluster file system is based on the proxy file system (PXFS), which has the following features:

- PXFS makes file access locations transparent. A process can open a file located anywhere in the global structure, and processes on all nodes can use the same path name to locate a file.
- PXFS uses coherency protocols to preserve the UNIX[®] file access semantics even if the file is accessed concurrently from multiple nodes.
- PXFS provides continuous access to data, even when failures occur. Applications do not detect failures as long as a path to the disks is still operational. This guarantee is maintained for raw disk access and all file system operations.

Note – PXFS is not a distinct file system type. That is, clients see the underlying file system type (for example, `ufs`).



Resource Groups

At the heart of any Sun Cluster highly available data service is the concept of a resource group. The cluster administrator creates resource group definitions. Each resource group is associated with a particular data service, such as Sun Cluster HA for NFS.

A resource group definition provides all of the necessary information for a designated backup system to take over the data services of a failed node. This includes the following:

- The Internet Protocol (IP) address or host name (logical host name) that users use to access the data service application
- The path to data and administrative resources
- The data service type that is to be started
- A list of participating nodes (primary or backup or scalable group)

Figure 1-9 shows the major components of a resource group definition.

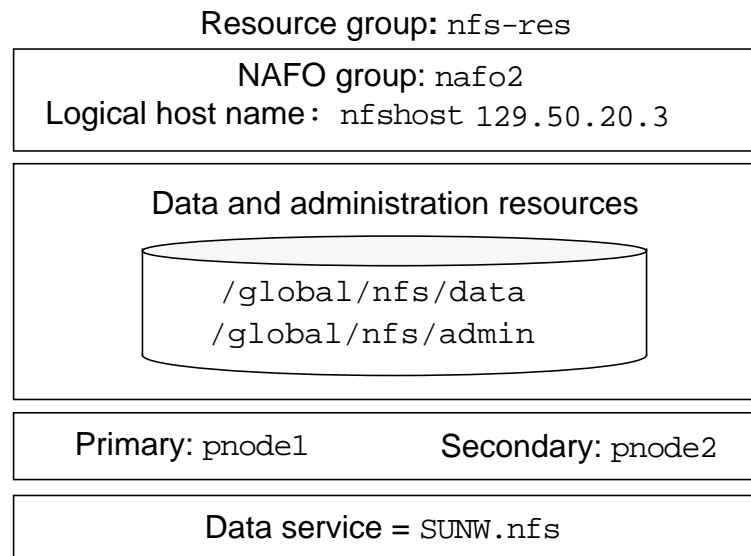


Figure 1-9 Resource Group Components

Note – Resource group configuration information is maintained in the globally available CCR database.



Parallel Database Application

The OPS configuration is characterized by multiple nodes that access a single database image. OPS configurations are throughput applications. OPS configurations are similar to a scalable data service: when a node fails, an application does not move to a backup system.

OPS uses a global lock management (GLM) scheme to prevent simultaneous data modification by two hosts. The lock ownership information is transferred between cluster hosts across the cluster transport system.

When a failure occurs, most of the recovery work is performed by the OPS software, which updates the lock information and resolves incomplete database transactions using the redo logs.

As shown in Figure 1-10, the OPS redo logs must also be available to both hosts.

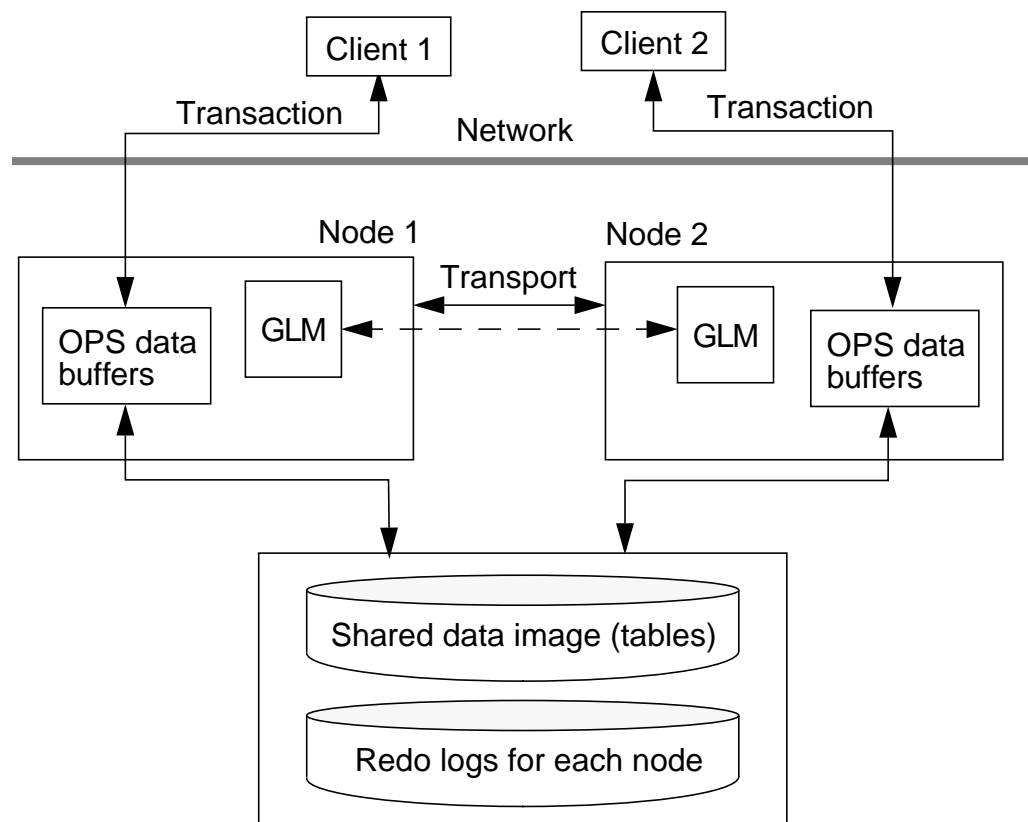


Figure 1-10 Parallel Database Features

Enabling Shared Data

The OPS database tables must be simultaneously available to both attached nodes. This is called shared access. Shared access can be achieved by either of two methods:

- Dual-host access of a raw Sun StorEdge A3500 logical unit (LUN)

A software volume manager is not used. The LUN is mirrored internally using hardware RAID.

- Dual-host shared access of a raw-mirrored VERITAS Volume Manager volume

The VERITAS Volume Manager cluster feature must be licensed separately. This allows you to create shared disk groups, which are owned by the cluster and not by a single host system.

The volumes must be mirrored between different storage arrays. The Sun StorEdge T3 storage arrays are supported in both single brick and partner pair configuration. Because these storage arrays have only a single data interface connection, they must be connected to multiple hosts using either fiber-channel hubs or switches.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ List the new Sun™ Cluster 3.0 07/01 features
- ☐ List the hardware elements that constitute a basic Sun Cluster system
- ☐ List the hardware and software components that contribute to the availability of a Sun Cluster system
- ☐ List the types of redundancy that contribute to the availability of a Sun Cluster system
- ☐ Identify the functional differences between failover, scalable, and parallel database cluster applications
- ☐ List the supported Sun™ Cluster 3.0 07/01 data services
- ☐ Explain the purpose of Disk ID (DID) devices
- ☐ Describe the relationship between system devices and the cluster global namespace
- ☐ Explain the purpose of resource groups in the Sun Cluster environment
- ☐ Describe the purpose of the cluster configuration repository
- ☐ Explain the purpose of each of the Sun Cluster fault monitoring mechanisms

Think Beyond

What are some of the most common problems encountered during cluster installation?

How do you install a cluster? What do you need to do first?

Do you need to be a database expert to administer a Sun Cluster system?

Terminal Concentrator

Objectives

Upon completion of this module, you should be able to:

- Describe the main features of the Sun Cluster administrative interface
- List the main functions of the terminal concentrator (TC) operating system
- Verify the correct TC cabling
- Configure the TC Internet Protocol (IP) address
- Configure the TC to self-load
- Verify the TC port settings
- Configure a TC default router, if necessary
- Verify that the TC is functional
- Use the TC `help`, `who`, and `hangup` commands
- Describe the purpose of the `telnet send brk` command

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- Why is this hardware covered so early in the course?
- Does this information apply to domain-based clusters?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *Sun™ Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *Sun™ Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *Sun™ Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *Sun™ Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *Sun™ Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *Sun™ Cluster 3.0 07/01 Concepts*, part number 806-7074
- *Sun™ Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *Sun™ Cluster 3.0 07/01 Release Notes*, part number 806-7078

Cluster Administration Interface

The TC is a hardware interface that provides the only access path to *headless* cluster host systems when these systems are halted or before any operating system software is installed.



Note – If the cluster host systems do not have a keyboard and frame buffer, they are said to be *headless*.

As shown in Figure 2-1, the cluster administration interface is a combination of hardware and software components that enables you to monitor and control one or more clusters from a remote location.

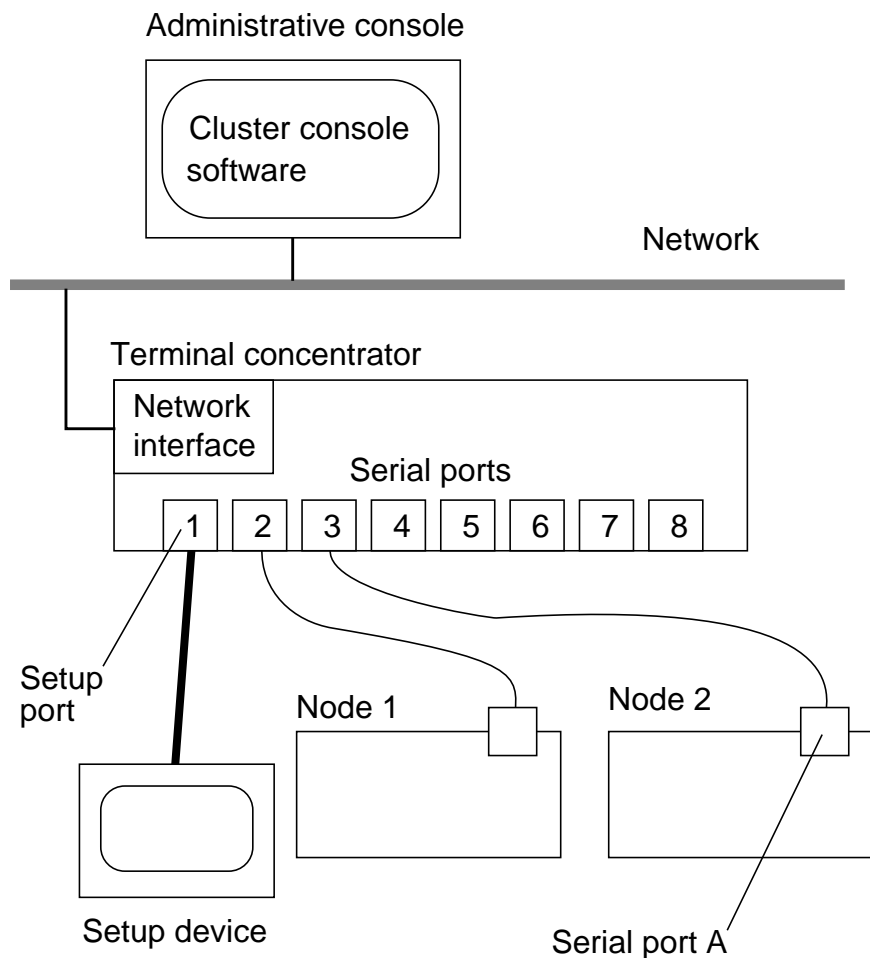


Figure 2-1 Cluster Administration Interface

Cluster Administration Elements

The relationship of the following elements is shown in Figure 2-1 on page 2-4.

Administrative Console

The administrative console can be any Sun workstation, providing it has adequate resources to support graphics and computer-intensive applications, such as the Sun Management Center software.

Administration Tools

There are several administration tools but only one of them, the Cluster Console `cconsole` tool, can access the cluster host systems when they are at the OpenBoot `ok` prompt.

The `cconsole` tool automatically links to each node in a cluster through the TC. A text window is provided for each node. The connection is functional even when the headless nodes are at the `ok` prompt level. This is the only path available to boot the cluster nodes or initially load the operating system software when the systems are headless.

Terminal Concentrator

The TC provides translation between the local area network (LAN) environment and the serial port interfaces on each node. The nodes do not have display monitors, so the serial ports are the only means of accessing each node to run local commands.

Cluster Host Serial Port Connections

If the cluster host systems do not have a display monitor or keyboard, the system firmware senses this when power is turned on and directs all system output through serial port A. This is a standard feature on Sun systems.

Note – The Sun Enterprise 10000 and Sun Fire systems do not need a TC for the administration interface.



Terminal Concentrator Overview

The TC used in the Sun Cluster systems has its own internal operating system and resident administration programs. The TC firmware is specially modified for Sun Cluster installation.



Note – If any other TC is substituted, it *must not* send an abort signal to the attached host systems when it is powered on.

As shown in Figure 2-2, the TC is a self-contained unit with its own operating system.

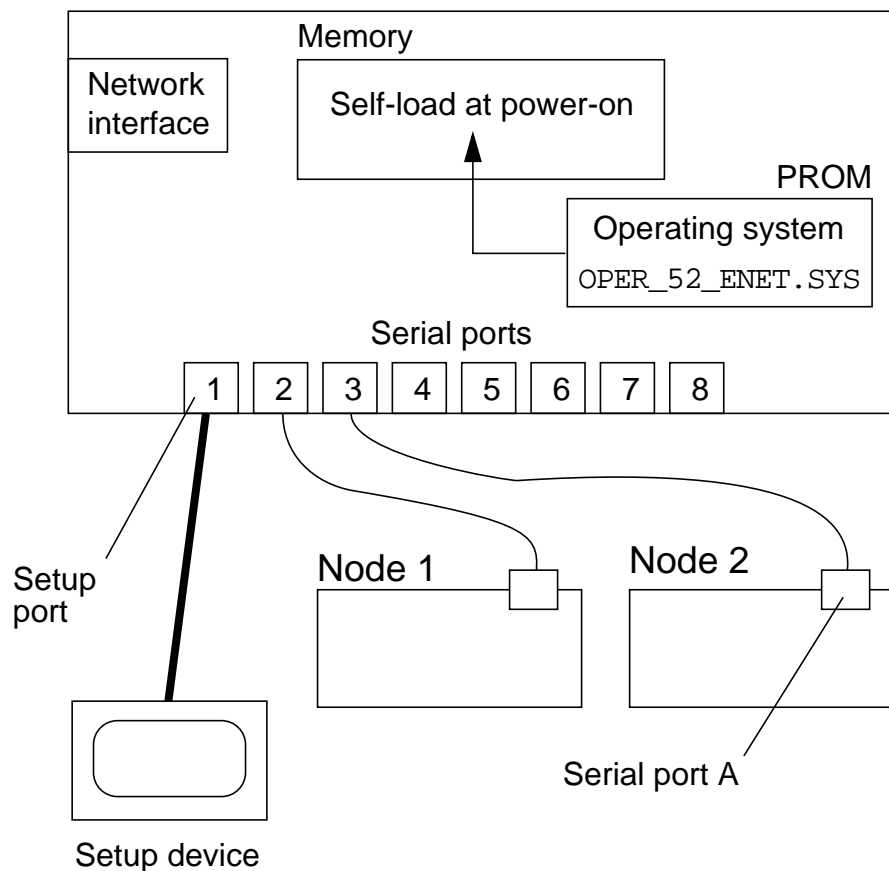


Figure 2-2 Terminal Concentrator Functional Diagram



Caution – If the programmable read-only memory (PROM) operating system is older than version 52, you must upgrade it.

Operating System Load

You can set up the TC to load its operating system either internally from the resident PROM or externally from a server. In the cluster application, it is always set to load internally. Placing the operating system on an external server can decrease the reliability of the terminal server.

When power is first applied to the TC, it performs the following steps:

1. It runs a PROM-based self-test and displays error codes.
2. It loads a resident PROM-based operating system into the TC memory.

Setup Port

Serial port 1 on the TC is a special purpose port that is used only during initial setup. It is used primarily to set up the IP address and load sequence of the TC. You can access port 1 from either a `tip` connection or from a locally connected terminal.

Terminal Concentrator Setup Programs

You must configure the TC nonvolatile random access memory (NVRAM) with the appropriate IP address, boot path, and serial port information. You use the following resident programs to specify this information:

- `addr`
- `seq`
- `image`
- `admin`

Setting Up the Terminal Concentrator

The TC must be configured for proper operation. Although the TC setup menus seem simple, they can be confusing, and it is easy to make a mistake. You can use the default values for many of the prompts.

Connecting to Port 1

To perform basic TC setup, you must connect to its setup port. Figure 2-3 shows a tip hardwire connection from the administrative console, but you can also connect an American Standard for Information Interchange (ASCII) terminal to the setup port.

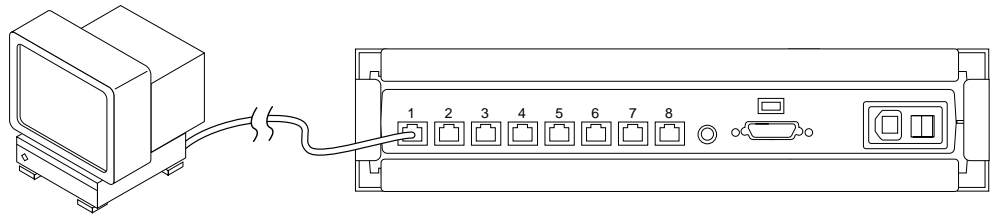


Figure 2-3 Setup Connection to Port 1

Enabling Setup Mode

To enable Setup mode, press the TC Test button shown in Figure 2-4 until the TC power indicator begins to blink rapidly, then release the Test button and press it again briefly.

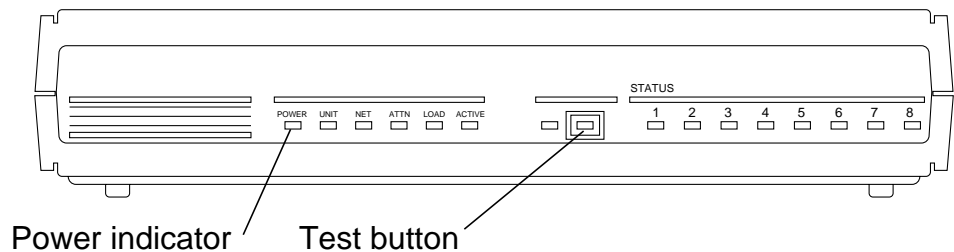


Figure 2-4 Terminal Concentrator Test Button

After you have enabled Setup mode, a `monitor::` prompt should appear on the setup device. Use the `addr`, `seq`, and `image` commands to complete the configuration.

Setting the Terminal Concentrator IP Address

The following example shows how to use the `addr` program to set the IP address of the TC. Usually this is set correctly when your cluster arrives, but you should always verify that it is correct.

```
monitor:: addr

Enter Internet address [192.9.22.98]::
129.150.182.100
Enter Subnet mask [255.255.255.0]::
Enter Preferred load host Internet address
[192.9.22.98]:: 129.150.182.100
Enter Broadcast address [0.0.0.0]::
129.150.182.255
Enter Preferred dump address [192.9.22.98]::
129.150.182.100
Select type of IP packet encapsulation
(ieee802/ethernet) [<ethernet>]::
    Type of IP packet encapsulation: <ethernet>

Load Broadcast Y/N [Y]:: y
```

Setting the Terminal Concentrator Load Source

The following example shows how to use the `seq` program to specify the type of loading mechanism to be used.

```
monitor:: seq

Enter a list of 1 to 4 interfaces to attempt to use
for downloading code or upline dumping. Enter them
in the order they should be tried, separated by
commas or spaces. Possible interfaces are:

    Ethernet: net
    SELF: self

Enter interface sequence [self]::
```

The `self` response configures the TC to load its operating system internally from the PROM when you turn on the power. The PROM image is currently called `OPER_52_ENET.SYS`.

Enabling the self-load feature negates other setup parameters that refer to an external load host and dump host, but you must still define these parameters during the initial setup sequence.



Note – Although you can load the TC's operating system from an external server, this introduces an additional layer of complexity that is prone to failure.

Specifying the Operating System Image

Even though the self-load mode of operation negates the use of an external load and dump device, you should still verify the operating system image name as shown by the following:

```
monitor:: image
```

```
Enter Image name ["oper_52_enet"]::  
Enter TFTP Load Directory ["9.2.7/"]::  
Enter TFTP Dump path/filename  
["dump.129.150.182.100"]::
```

```
monitor::
```



Note – Do not define a dump or load address that is on another network because you receive additional questions about a gateway address. If you make a mistake, you can press Control-C to abort the setup and start again.

Setting the Serial Port Variables

The TC port settings must be correct for proper cluster operation. This includes the type and mode port settings. Port 1 requires different type and mode settings. You should verify the port settings before installing the cluster host software. The following is an example of the entire procedure:

```
admin-ws# telnet terminal_concentrator_name
Trying terminal concentrator IP address
Connected to sec-tc.
Escape character is '^]'.
Rotaries Defined:
    cli
Enter Annex port name or number: cli
Annex Command Line Interpreter * Copyright 1991
Xylogics, Inc.
annex: su
Password: type the password
annex# admin
Annex administration MICRO-XL-UX R7.0.1, 8 ports
admin: show port=1 type mode
Port 1:
type: hardwired      mode: cli
admin:set port=1 type hardwired mode cli
admin:set port=2-8 type dial_in mode slave
admin:set port=1-8 imask_7bits Y
admin: quit
annex# boot
bootfile: <CR>
warning: <CR>.
```



Note – Do not perform this procedure through the special setup port but through public network access.

The default TC password is its IP address, including the periods.

The `imask_7bits` variable masks out everything but the standard 7-bit character set.

Disabling Terminal Concentrator Routing

If you access a TC from a host that is on a different network, the TC's internal routing table can overflow. If the TC routing table overflows, network connections can be intermittent or lost completely.

As shown in Figure 2-5, you can correct this problem by setting a default route within the TC configuration file `config.annex`.

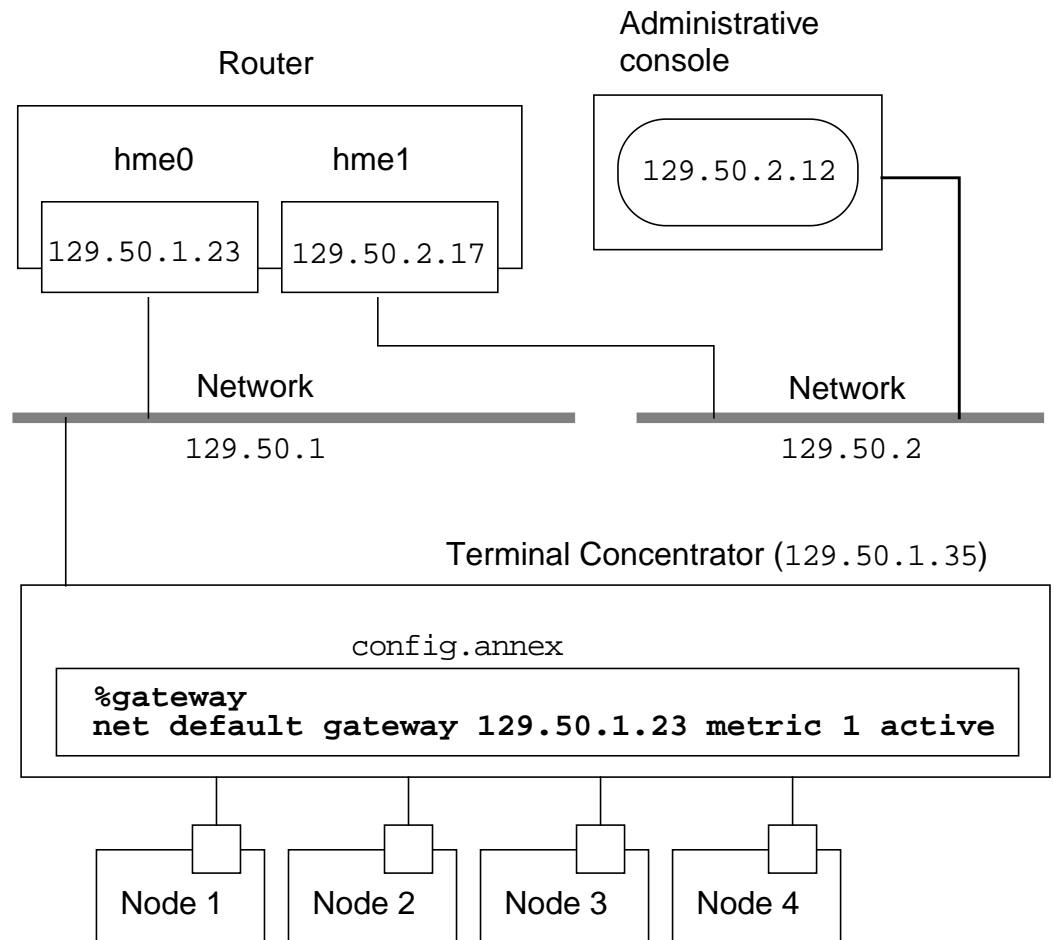


Figure 2-5 Terminal Concentrator Routing

Creating a Terminal Concentrator Default Route

To create a default route for the TC, you must edit a electrically erasable programmable read-only memory (EEPROM) file in the TC named `config.annex`. You must also disable the TC routing function. The following is a summary of the general process.

```
admin-ws# telnet tcl.central
Trying 192.9.200.1 ...
Connected to 192.9.200.1.
Escape character is '^]'.
[Return] [Return]
Enter Annex port name or number: cli
...
annex: su
Password: root_password
annex: edit config.annex
(Editor starts)
Ctrl-W:save and exit Ctrl-X:exit Ctrl-F:page down
Ctrl-B:page up
%gateway
net default gateway 192.9.200.2 metric 1 active ^W
annex# admin set annex routed n
You may need to reset the appropriate port, Annex
subsystem or
reboot the Annex for changes to take effect.
annex# boot
```



Note – You must enter an IP routing address appropriate for your site. While the TC is rebooting, the node console connections are not available.

Using Multiple Terminal Concentrators

A single TC can provide serial port service to a maximum of seven systems. If you have eight nodes, you must use two TCs for a single cluster.

The maximum length for a TC serial port cable is approximately 348 feet. As shown in Figure 2-6, it might be necessary to have cluster host systems separated by more than the serial port cable limit. You might need a dedicated TC for each node in a cluster.

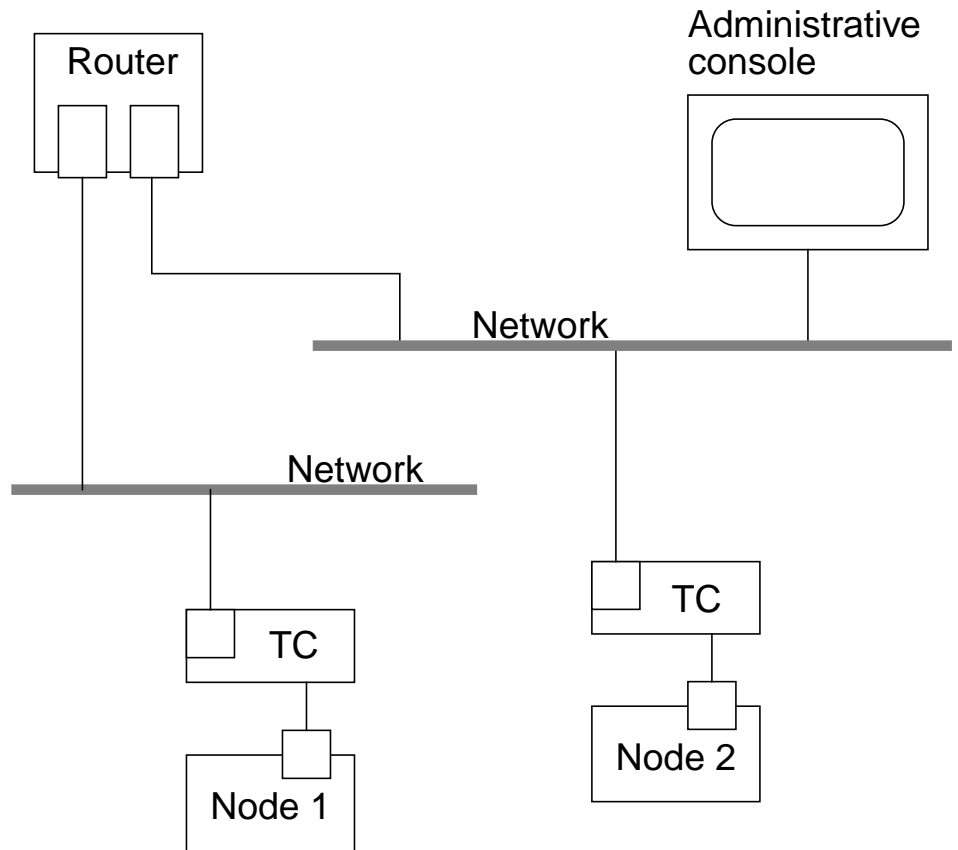


Figure 2-6 Multiple Terminal Concentrators

Terminal Concentrator Troubleshooting

Occasionally, it is useful to be able to manually manipulate the TC. The commands to do this are not well documented in the cluster manuals.

Using the `telnet` Command to Manually Connect to a Node

If the `cconsole` tool is not using the TC serial ports, you can use the `telnet` command to connect to a specific serial port as follows:

```
# telnet tc_name 5002
```

You can then log in to the node attached to port 5002. After you have finished and logged out of the node, you must break the `telnet` connection with the Control-] keyboard sequence and then type **quit**. If you do not, the serial port remains locked and cannot be used by other applications, such as the `cconsole` tool.

Using the `telnet` Command to Abort a Node

If you have to abort a cluster node, you can either use the `telnet` command to connect directly to the node and use the Control-] keyboard sequence, or you can use the Control-] keyboard sequence in a cluster console window. When you have the `telnet` prompt, you can abort the node with the following command:

```
telnet > send brk  
ok
```

Note – You might have to repeat the command multiple times.



Connecting to the Terminal Concentrator Command-Line Interpreter

You can use the `telnet` command to connect directly to the TC, and then use the resident command-line interpreter (CLI) to perform status and administration procedures.

```
# telnet IPaddress
Trying 129.146.241.135...
Connected to 129.146.241.135
Escape character is '^]'.

Enter Annex port name or number: cli

Annex Command Line Interpreter * Copyright 1991
Xylogics, Inc.
annex:
```

Using the Terminal Concentrator `help` Command

After you connect directly into a terminal concentrator, you can get online help as follows:

```
annex: help
annex: help hangup
```

Identifying and Resetting a Locked Port

If a node crashes, it can leave a telnet session active that effectively locks the port from further use. You can use the `who` command to identify which port is locked, and then use the `admin` program to reset the locked port. The command sequence is as follows:

```
annex: who
annex: su
Password:
annex# admin
Annex administration MICRO-XL-UX R7.0.1, 8 ports
admin : reset 6
admin : quit
annex# hangup
```

Notes: Erasing Terminal Concentrator Settings

This section is for instructor information only. It is to be used in case a student changes the default TC superuser password.

Using the TC `erase` command can be dangerous. Use it only when you have forgotten the superuser password. It sets the password to its default setting, which is the terminal concentrator's IP address. It also returns all other settings to their default values. For security reasons, the `erase` command is available only through the port 1 interface. A typical procedure is as follows:

```
monitor :: erase
```

- 1) EEPROM(i.e. Configuration information)
- 2) FLASH(i.e. Self boot image)

```
Enter 1 or 2 :: 1
```



Warning – Do not use option 2 of the `erase` command; it destroys the TC boot PROM-resident operating system.

Exercise: Configuring the Terminal Concentrator

In this exercise, you complete the following tasks:

- Verify the correct TC cabling
- Configure the TC IP address
- Configure the TC to self-load
- Verify the TC port settings
- Verify that the TC is functional

Preparation

Before starting this lab, record the name and IP address assignment for your TC. Ask your instructor for assistance.

TC Name:_____

TC IP address:_____

If there are special root passwords set on the lab TCs, tell students now.

Stress that this is typically the first step in a real installation.

If they are to use the tip hardwire method and the administrative consoles in the lab have a shared A/B serial port connector, you must tell the students the correct port setting to use in the “Connecting Tip Hardwire” section of this exercise.



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Verifying the Network and Host System Cabling

A TC can have one or two Ethernet connections, depending on its age. All TC generations have the same serial port connections.

Before you begin to configure the TC, you must verify that the network and cluster host connections are correct.

1. Inspect the rear of the TC, and make sure it is connected to a public network.

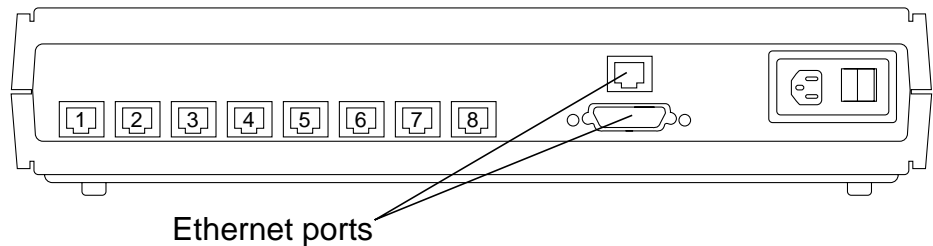


Figure 2-7 Terminal Concentrator Network Connection

2. Verify that the serial ports are properly connected to the cluster host systems. Each output should go to serial port A on the primary system board of each cluster host system.

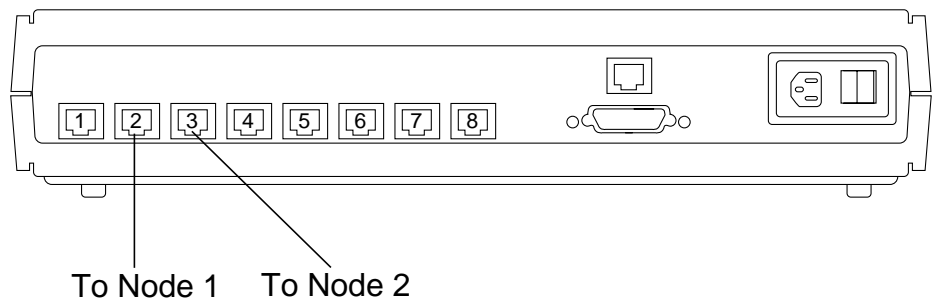


Figure 2-8 Concentrator Serial Port Connections



Note – In clusters with more than two nodes, there are additional serial port connections to the TC.

To set up the TC, you can either connect a “dumb” terminal to serial port 1 or use the `tip` hardwire command from a shell on the administrative console.

If you are using a local terminal connection, continue with the next section, “Task – Connecting a Local Terminal.” If you are using the administrative console, proceed to the “Task – Connecting Tip Hardwire” on page 2-22.

Task – Connecting a Local Terminal

Perform the following procedure only if you are connecting to the TC with a dumb terminal.

Perform the following steps to connect a local terminal:

1. Connect the local terminal to serial port 1 on the back of the TC using the cable supplied.

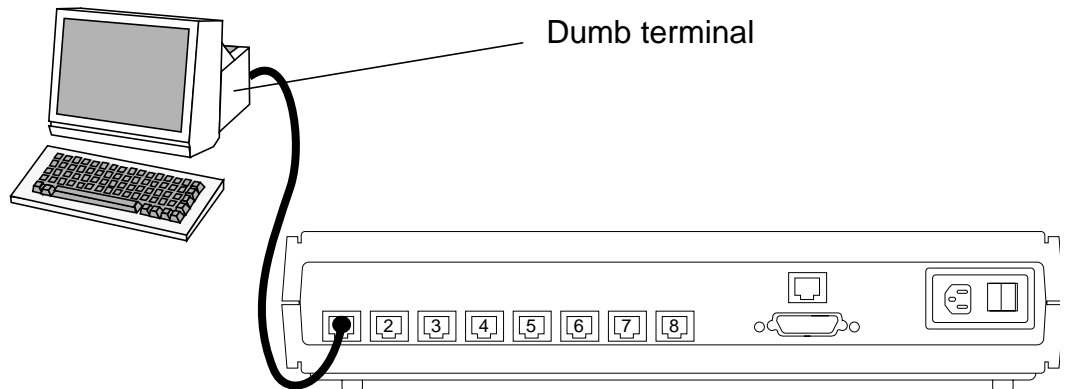


Figure 2-9 Concentrator Local Terminal Connection

Note – Do not use a cable length of more than 500 feet. Use a null modem cable.

2. Verify that the local terminal operating parameters are set to 9600 baud, 7 data bits, no parity, and 1 stop bit.
3. Proceed to the “Task – Achieving Setup Mode” on page 2-23.



Task – Connecting Tip Hardware

Perform the following procedure only if you are using the `tip` connection method to configure the TC. If you have already connected to the TC with a dumb terminal, skip this procedure.

Perform the following steps to connect the `tip` hardware:

1. Connect serial port B on the administrative console to serial port 1 on the back of the TC using the cable supplied.

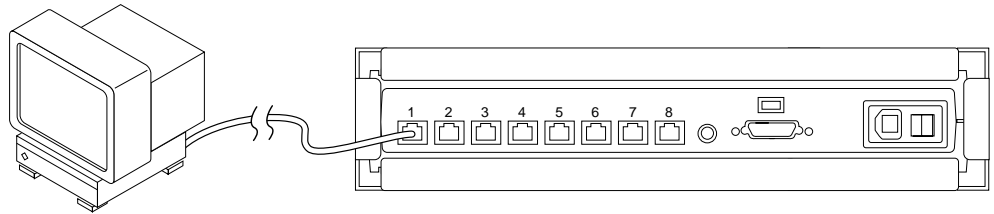


Figure 2-10 Concentrator to Tip Hardware Connection

Note – Do not use a cable length over 500 feet. Use a null modem cable.



2. Verify that the `hardwire` entry in the `/etc/remote` file matches the serial port you are using (see figure 2-10):

```
hardwire:\  
:dv=/dev/term/b:br#9600:el=^C^S^Q^U^D:ie=%$:oe=^D:
```

The serial port designator must match
the serial port you are using.

The baud rate must be 9600.

3. Open a shell window on the administrative console, figure 2-10 verifying the `hardwire` entry and make the `tip` connection by typing the following command:

```
# tip hardwire
```

Task – Achieving Setup Mode

Before the TC configuration can proceed, you must first place it in its Setup mode of operation. Once in Setup mode, the TC accepts configuration commands from a serial device connected to port 1.

Perform the following steps to enable Setup mode:

1. Press and hold the TC Test button until the TC power indicator begins to blink rapidly, then release the Test button, and press it again briefly.

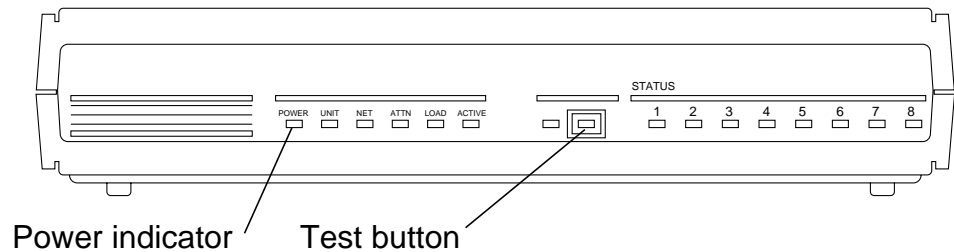


Figure 2-11 Enabling Setup Mode on the Terminal Concentrator

2. After the TC completes its power-on self-test, you should see the following prompt on the shell window or on the local terminal:

```
monitor::
```

Note – It can take a minute or more for the self-test process to complete.



Task – Configuring the IP Address

Verify that the TC IP address and preferred load host address are set to your assigned value. This address must not conflict with any other network systems or devices.

Perform the following step to configure the TC IP address using the `addr` command:

1. Type **addr** at the `monitor::` prompt.

```
monitor:: addr
```

```
Enter Internet address [192.9.22.98]::  
129.150.182.101
```

```
Enter Subnet mask [255.255.255.0]::
```

```
Enter Preferred load host Internet address  
[192.9.22.98]::  
129.150.182.101
```

```
***Warning: Local host and Internet host are the  
same***
```

```
Enter Broadcast address [0.0.0.0]::  
129.150.182.255
```

```
Enter Preferred dump address [192.9.22.98]::  
129.150.182.101
```

```
Select type of IP packet encapsulation  
(ieee802/ethernet) [<ethernet>]:  
Type of IP packet encapsulation: <ethernet>
```

```
Load Broadcast Y/N [Y]:: n
```

```
monitor::
```

Task – Configuring the TC to Self-Load

When the TC is turned on, you must configure it to load a small operating system. You can use the `seq` command to define the location of the operating system and the `image` command to define its name.

Perform the following steps to configure the TC to load from itself instead of trying to load from a network host,:

1. Type the **seq** command at the `monitor::` prompt.

```
monitor:: seq
```

Enter a list of one to four interfaces to attempt to use for downloading code or upline dumping. Enter them in the order they should be tried, separated by commas or spaces. Possible interfaces are:

```
Ethernet: net  
SELF: self
```

```
Enter interface sequence [self]::
```

2. To configure the TC to load the correct operating system image, type the **image** command at the `monitor::` prompt.

```
monitor:: image  
Enter Image name ["oper.52.enet"]::  
Enter TFTP Load Directory ["9.2.7/"]::
```

```
Enter TFTP Dump path/filename  
["dump.129.150.182.101"]::
```

3. If you used a direct terminal connection, disconnect it from the TC when finished.
4. If you used the `tip` hardware method, break the `tip` connection by typing the `~.` sequence in the shell window.

Task – Verifying the Self-Load Process

Before proceeding, you must verify that the TC can complete its self-load process and that it answers to its assigned IP address by performing the following steps:

1. Turn off the TC power for at least 10 second, and then turn it on again.
2. Observe the light-emitting diodes (LEDs) on the TC front panel. After the TC completes its power-on self-test and load routine, the front panel LEDs should have the settings shown in Table 2-1.

Table 2-1 LED Front Panel Settings

Power (Green)	Unit (Green)	Net (Green)	Attn (Amber)	Load (Green)	Active (Green)
On	On	On	Off	Off	Intermittent blinking



Note – It takes at least one minute for the process to complete. The Load LED extinguishes after the internal load sequence is complete.

Verifying the Terminal Concentrator Pathway

Complete the following steps on the administrative console from a shell or command tool window:

1. Test the network path to the TC using the following command:

```
# ping IPaddress
```



Note – Substitute the IP address of your TC for *IPaddress*.

Task – Verifying the TC Port Settings

You must set the TC port variable *type* to *dial_in* for each of the eight TC serial ports. If it is set to *hardwired*, the cluster console might be unable to detect when a port is already in use. There is also a related variable called *imask_7bits* that you must set to *Y*.

You can verify and, if necessary, modify the *type*, *mode*, and *imask_7bits* variable port settings with the following procedure:

1. On the administrative console, use the `telnet` command to connect to the TC. Do not use a port number.

```
# telnet IPaddress
Trying 129.146.241.135...
Connected to 129.146.241.135
Escape character is '^]'.
```

2. Enable the command-line interpreter, use the `su` command to get to the root account, and start the `admin` program.

```
Enter Annex port name or number: cli

Annex Command Line Interpreter * Copyright 1991
Xylogics, Inc.

annex: su
Password:
annex# admin
Annex administration MICRO-XL-UX R7.0.1, 8 ports

admin :
```



Note – By default, the superuser password is the TC IP address. This includes the periods in the IP address.

3. Use the `show` command to examine the current setting of all ports.

```
admin : show port=1-8 type mode
```

4. Perform the following procedure to change the port settings and to end the TC session.

```
admin:set port=1 type hardwired mode cli
admin:set port=1-8 imask_7bits Y
admin:set port=2-8 type dial_in mode slave
admin: quit
annex# boot
bootfile: <CR>
warning: <CR>
Connection closed by foreign host.
```



Note – It takes at least one minute for the process to complete. The Load LED extinguishes after the internal load sequence is complete.

Task – Terminal Concentrator Troubleshooting

Perform the following steps to troubleshoot the TC:

1. On the administrative console, use the `telnet` command to connect to the TC. Do not use a port number.

```
# telnet IPaddress
Trying 129.146.241.135...
Connected to 129.146.241.135
Escape character is '^]'.
```

2. Enable the command line interpreter.

```
Enter Annex port name or number: cli

Annex Command Line Interpreter * Copyright 1991
Xylogics, Inc.

annex:
```

3. Practice using the `help` and `who` commands.
4. End the session with the `hangup` command.

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Describe the main features of the Sun Cluster administrative interface
- ☐ List the main functions of the TC operating system
- ☐ Verify the correct TC cabling
- ☐ Configure the TC IP address
- ☐ Configure the TC to self-load
- ☐ Verify the TC port settings
- ☐ Configure a TC default router, if necessary
- ☐ Verify that the TC is functional
- ☐ Use the TC `help`, `who`, and `hangup` commands
- ☐ Describe the purpose of the `telnet send brk` command

Think Beyond

Is there a significant danger if the TC port variables are not set correctly?

Is the TC a single point of failure? What would happen if it failed?

Installing the Administrative Console

Objectives

Upon completion of this module, you should be able to:

- List the Sun Cluster administrative console functions
- Install the Sun Cluster console software on the administrative console
- Set up the administrative console environment
- Configure the Sun Cluster console software

Relevance

Present the following question to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answer to this question, the answer should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following question is relevant to understanding the content of this module:

- How important is the administrative console during the configuration of the cluster host systems?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Sun Cluster Console Software

As shown in Figure 3-1, the Sun Cluster console software is installed on the administrative console. The Sun Cluster framework and data service software is installed on each of the cluster host systems along with appropriate virtual volume management software.

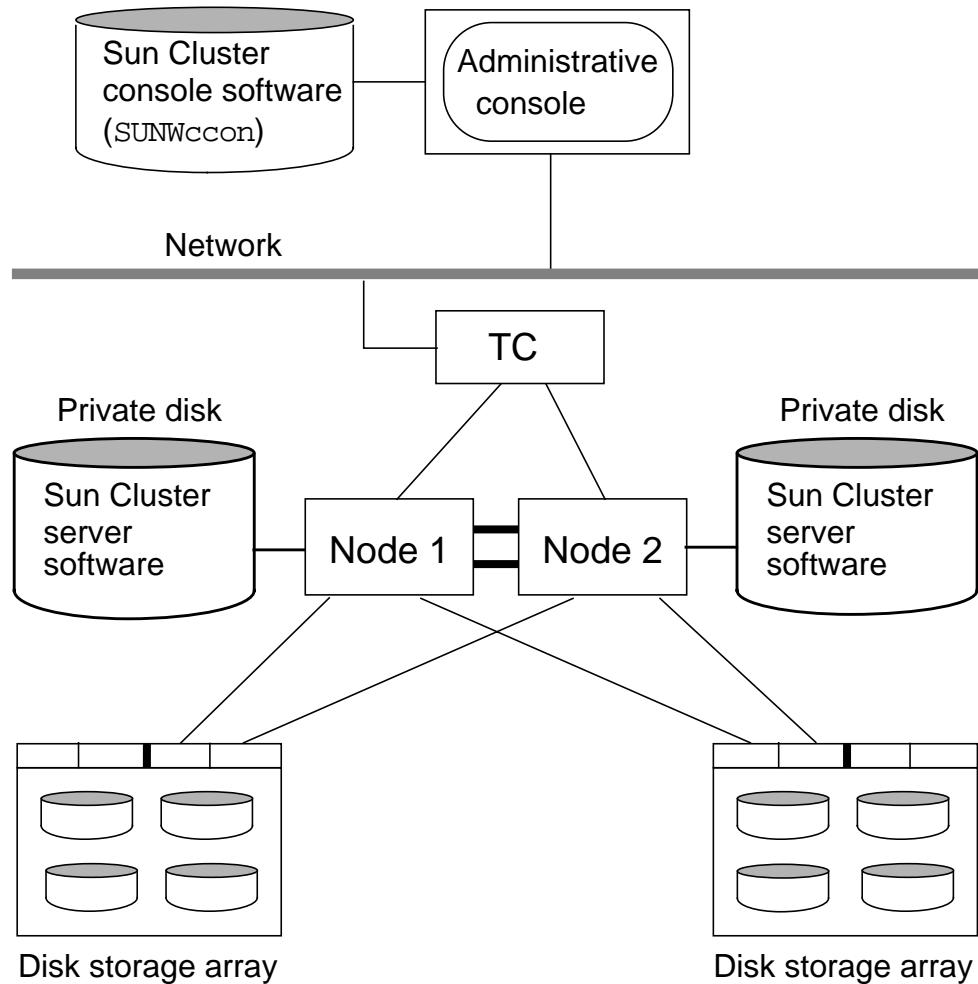


Figure 3-1 Sun Cluster Console Software

Console Software Installation

The administrative console software is contained in a single package: SUNWcccon. The SUNWcccon package is installed manually from the Sun™ Cluster 3.0 07/01 software distribution CD-ROM.

Sun Cluster Console Tools

Use the cluster administration tools to manage a cluster. They provide many useful features, including a:

- Centralized tool bar
- Command-line interface to each cluster host

The console programs are `ccp`, `cconsole`, `crlogin`, and `ctelnet`.

You can start the cluster administration tools manually or by using the `ccp` program tool bar.

The Cluster Control Panel

As shown in Figure 3-2, the Cluster Control Panel provides centralized access to three variations of the cluster console tool.

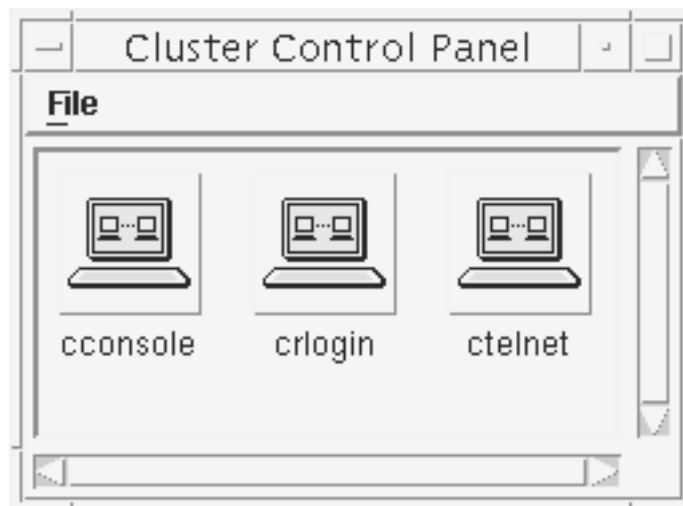


Figure 3-2 Cluster Control Panel

Starting the Cluster Control Panel

To start the Cluster Control Panel, type the following command:

```
# /opt/SUNWcluster/bin/ccp [clustername] &
```

Cluster Console

The cluster console tool uses the terminal concentrator (TC) to access the cluster host systems through serial port interfaces. The advantage of this is that you can connect to the cluster host systems even when they are halted. This is essential when booting headless systems and can be useful during initial cluster host configuration.

As shown in Figure 3-3, the cluster console tool uses `xterm` windows to connect to each of the cluster host systems.

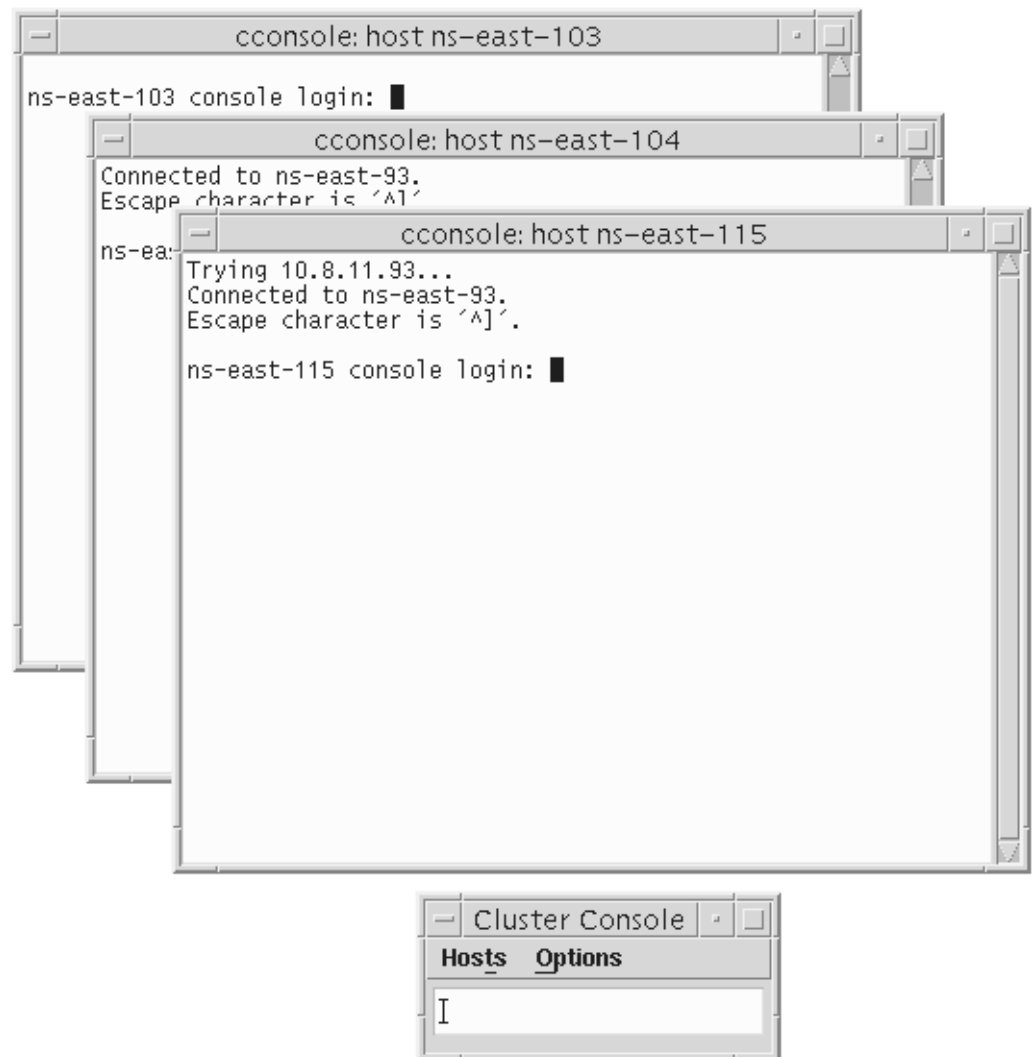


Figure 3-3 Cluster Console Windows

Manually Starting the `cconsole` Tool

As shown, you can use the `cconsole` tool manually to connect to a single cluster node or to the entire cluster.

```
# /opt/SUNWcluster/bin/cconsole node1 &  
# /opt/SUNWcluster/bin/cconsole my-cluster &  
# /opt/SUNWcluster/bin/cconsole node3 &
```

Cluster Console Host Windows

There is a host window for each node in the cluster. You can enter commands in each host window separately.

Set the `TERM` environment variable to `dterm` for best operation.

Cluster Console Common Window

The common window shown in Figure 3-4 enables you to enter commands to all host system windows at the same time. All of the windows are tied together, so when you move the common window, the host windows follow. The Options menu allows you to ungroup the windows, move them into a new arrangement, and group them again.

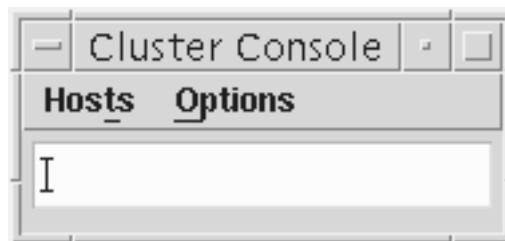


Figure 3-4 Cluster Console Common Window

Cluster Console Window Variations

There are three variations of the cluster console tool that each use a different method to access the cluster hosts. They all look and behave the same way.

- Cluster console (`cconsole`)

The `cconsole` program accesses the host systems through the TC interface. The Solaris Operating Environment does not have to be running on the cluster host systems.

Only one connection at a time can be made through the TC to serial port A of a cluster node. You cannot start a second instance of `cconsole` for the same cluster.

For Sun Enterprise 10000 domains, a `telnet` connection is made to the `ssp` account on the domain's system service processor (SSP), and then a `netcon` session is established.

- Cluster console (`crlogin`)

The `crlogin` program accesses the host systems through the public network using the `rlogin` command. The Solaris Operating Environment must be running on the cluster host systems.

- Cluster console (`ctelnet`)

The `ctelnet` program accesses through the public network using the `telnet` command. The Solaris Operating Environment must be running on the cluster host systems.

Cluster Console Tools Configuration

All of the necessary information needed for the cluster administration tools that run on the administrative console is configured in two files. The files are:

- `/etc/clusters`
- `/etc/serialports`

When you install the Sun Cluster console software on the administrative console, you must manually create the `clusters` and `serialports` files and populate them with the necessary information.

Configuring the `/etc/clusters` File

The `/etc/clusters` file contains the name of a cluster followed by the names of node that are part of that cluster.

The following is a typical entry in the `/etc/clusters` file:

```
sc-cluster sc-node1 sc-node2
```

The single-line entry defines a cluster named `sc-cluster` that has two nodes named `sc-node1` and `sc-node2`.



Note – The cluster name is purely arbitrary, but it should agree with the name you use when you install the server software on each of the cluster host systems.

You can define many different clusters in a single `/etc/clusters` file, so you can administer several clusters from a single administrative console.

Configuring the /etc/serialports File

The `/etc/serialports` file defines the TC path to each node defined in the `/etc/clusters` file. You must enter the paths to all nodes in all of your clusters in this file.

The following are typical entries in the `/etc/serialports` file:

```
sc-node1 sc-tc 5002
sc-node2 sc-tc 5003
```

There is a line for each cluster host that describes the name of each host, the name of the TC, and the TC port to which each host is attached.

For the Sun Enterprise 10000 server, the `/etc/serialports` entries for each cluster domain are configured with the domain name, the SSP name, and (always) the number 23, which represents the telnet port.

```
sc-10knode1 sc10k-ssp 23
sc-10knode2 sc10k-ssp 23
```

Note – A TC is not needed in Sun Enterprise 10000 cluster installations.



For the Sun Fire systems, the `/etc/serialports` entries for each cluster domain are configured with the domain name, the Sun Fire system control board name, and a telnet port number similar to those used with a terminal concentrator.

```
sf1_node1 sf1_ctrlb 5001
sf1_node2 sf1_ctrlb 5002
```

Note – A TC is not needed in Sun Fire cluster installations.



Caution – When upgrading the cluster software, the `/etc/serialports` and `/etc/clusters` files are overwritten. Make a backup copy before starting an upgrade.

Multiple Terminal Concentrator Configuration

If you have widely separated nodes or an eight-node cluster, you might need more than one TC to manage a single cluster. The following examples demonstrate how the `/etc/clusters` and `/etc/serialports` files might appear for an eight-node cluster.

The following entry in the `/etc/clusters` file represents the nodes in an eight-node cluster:

```
sc-cluster sc-node1 sc-node2 sc-node3 sc-node4 \  
sc-node5 sc-node6 sc-node7 sc-node8
```

The single-line entry defines a cluster named `sc-cluster` that has eight nodes.

The following entries in the `/etc/serialports` file define the TC paths to each node in an eight-node cluster:

```
sc-node1 sc-tc1 5002  
sc-node2 sc-tc1 5003  
sc-node3 sc-tc1 5004  
sc-node4 sc-tc1 5005  
sc-node5 sc-tc2 5002  
sc-node6 sc-tc2 5003  
sc-node7 sc-tc2 5004  
sc-node8 sc-tc2 5005
```

There is a line for each cluster host that describes the name of the host, the name of the TC, and the TC port to which the host is attached.

Exercise: Configuring the Administrative Console

In this exercise, you complete the following tasks:

- Install and configure the Sun Cluster console software on an administrative console
- Configure the Sun Cluster administrative console environment for correct Sun Cluster console software operation
- Start and use the basic features of the Cluster Control Panel and the Cluster Console

Preparation

This lab assumes that the Solaris 8 Operating Environment software is already installed on all of the cluster systems. Perform the following steps to prepare for the lab:

1. Ask your instructor for the name assigned to your cluster.

Cluster name: _____

2. Record the information in Table 3-1 about your assigned cluster before proceeding with this exercise:

Table 3-1 Cluster Names and Addresses

System	Name	IP Address
Administrative console		
TC		
Node 1		
Node 2		

3. Ask your instructor for the location of the Sun Cluster software.

Software location: _____



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Updating Host Name Resolution

Even though your site might use Network Information Service (NIS) or Domain Name Service (DNS) to resolve host names, it can be beneficial to resolve the names locally on the administrative console and cluster hosts. This can be valuable in the case of naming service failures. The `cconsole` program does not start unless it can first resolve the host names in the `/etc/clusters` file.

1. If necessary, edit the `/etc/hosts` file on your administrative console and add the IP addresses and names of the TC and the host systems in your cluster.
2. Verify that the `/etc/nsswitch.conf` file entry for `hosts` has `files` listed first.

```
hosts:      files nis
```

Task – Installing the Cluster Console Software

Perform the following steps to install the cluster console software:

1. Log in to your administrative console as user `root`.
2. Move to the Sun™ Cluster 3.0 07/01 packages directory.

Note – Either load the Sun™ Cluster 3.0 07/01 CD or move to the location provided by your instructor.



3. Verify that you are in the correct location.

```
# ls
SUNWcccon  SUNWscman  SUNWscsch  SUNWscva
SUNWmdm    SUNWscr    SUNWscshl  SUNWscvm
SUNWscdev  SUNWscsal  SUNWscssv  SUNWscvr
SUNWscfab  SUNWscsam  SUNWscu    SUNWscvw
```

4. Install the cluster console software package.

```
# pkgadd -d . SUNWcccon
```

Task – Verifying the Administrative Console Environment

Perform the following steps to verify the Administrative Console:

1. Verify that the following search paths and variables are present in the `/.profile` file:

```
PATH=$PATH:/opt/SUNWcluster/bin
MANPATH=$MANPATH:/opt/SUNWcluster/man
EDITOR=/usr/bin/vi
export PATH MANPATH EDITOR
```



Note – Create the `.profile` file in the `/` directory if necessary, and add the changes.

2. Execute the `.profile` file to verify changes that have been made.

```
# ./profile
```



Note – It is best to log out and log in again to set new variables.

Task – Configuring the `/etc/clusters` File

The `/etc/clusters` file has a single line entry for each cluster you intend to monitor. The entries are in the form:

```
clustername host1name host2name host3name host4name
```

Sample `/etc/clusters` File

```
sc-cluster pnode1 pnode2
```

Perform the following steps to configure the `/etc/clusters` file:

1. Edit the `/etc/clusters` file, and add a line using the cluster and node names assigned to your system.



Note – Your console system must be able to resolve the host names.

Task – Configuring the `/etc/serialports` File

The `/etc/serialports` file has an entry for each cluster host describing the connection path. The entries are in the form:

```
hostname      tcname      tcport
```

Sample `/etc/serialports` File

```
pnode1      cluster-tc      5002
pnode2      cluster-tc      5003
pnode3      cluster-tc      5004
```

Perform the following steps to configure the `/etc/serialports` file:

1. Edit the `/etc/serialports` file and add lines using the node and TC names assigned to your system.



Note – Make a backup of the `/etc/serialports` and `/etc/clusters` files before starting a Sun Cluster software upgrade.

Task – Starting the `cconsole` Tool

This section provides a good functional verification of the TC in addition to the environment configuration. Perform the following steps to start the `cconsole` tool:

1. Make sure power is on for the TC and all of the cluster hosts.
2. Start the `cconsole` tool on the administrative console.

```
# cconsole clustername &
```



Note – Substitute the name of your cluster for *clustername*.

3. Place the cursor in the `cconsole` Common window, and press Return several times. You should see a response on all of the cluster host windows. If not, ask your instructor for assistance.



Note – The `cconsole` Common window is useful for simultaneously loading the Sun Cluster software on all of the cluster host systems.

4. If the cluster host systems are not booted, boot them now.

```
ok boot
```

5. After all cluster host systems have completed their boot, log in as user `root`.
6. Practice using the Common window Group Term Windows feature under the Options menu. You can ungroup the `cconsole` windows, rearrange them, and then group them together again.

Task – Using the `ccp` Control Panel

The `ccp` control panel can be useful if you need to use the console tool variations `crlogin` and `ctelnet`. Perform the following steps to use the `ccp` control panel:

1. Start the `ccp` tool (`# ccp clustername &`).
2. Practice using the `crlogin` and `ctelnet` console tool variations.
3. Quit the `crlogin`, `ctelnet`, and `ccp` tools.

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ List the Sun Cluster administrative console functions
- ☐ Install the Sun Cluster console software on the administrative console
- ☐ Set up the administrative console environment
- ☐ Configure the Sun Cluster console software

Think Beyond

What is the advantage of the `/etc/clusters` and `/etc/serialports` files?

What is the impact on the cluster if the administrative console is not available? What alternatives could you use?

Preinstallation Configuration

Objectives

Upon completion of this module, you should be able to:

- List the Sun Cluster boot disk requirements
- Physically configure a cluster topology
- Configure a supported cluster interconnect system
- Identify single points of failure in a cluster configuration
- Identify the quorum devices needed for selected cluster topologies
- Verify storage firmware revisions
- Physically configure a public network group

Relevance

Present the following question to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answer to this questions, the answer should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following question is relevant to understanding the content this module:

- Why is there so much preinstallation planning required for an initial software installation?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Configuring Cluster Servers

The servers you use in a Sun™ Cluster 3.0 07/01 configuration must conform to a number of general software and hardware requirements to qualify for support. The requirements include both hardware and software in the following areas:

- Boot device restrictions
- Server hardware restrictions

Boot Device Restrictions

With the Sun™ Cluster 3.0 07/01 release, there are several restrictions on boot devices including:

- You cannot use a shared storage device as a boot device. If a storage device is connected to more than one host, it is shared.
- The Solaris 8 Operating Environment is supported with the following restrictions:
 - Sun™ Cluster 3.0 07/01 requires a minimum of End User level support. Data service applications might require a higher level of support, such as Developer or Entire Distribution.
 - Alternate Pathing is not supported.
 - The use of Dynamic Reconfiguration is not supported.
 - The use of VERITAS Dynamic Multipathing is not supported.
- The boot disk partitions have several requirements:
 - Swap space must be twice the size of memory.
 - There must be a 100-Mbyte `/globaldevices` file system.
 - The Solstice DiskSuite application requires a 10-Mbyte partition for its Meta State Databases.



Note – The 100-Mbyte `/globaldevices` file system is modified during the Sun™ Cluster 3.0 07/01 installation. It is automatically renamed to `/global/.devices`.

Boot Disk JumpStart Profile

If you decide to configure your cluster servers using the JumpStart utility, the following JumpStart profile represents a starting point for boot disk configuration:

install_type	initial_install
system_type	standalone
partitioning	explicit
cluster	SUNWCall
usedisk	c0t0d0
filesys	c0t0d0s1 1024 swap
filesys	c0t0d0s4 100 /globaldevices
filesys	c0t0d0s7 10
filesys	c0t0d0s0 free /

Server Hardware Restrictions

All cluster servers must meet the following hardware requirements:

- Each server must have a minimum of 512 Mbytes of memory.
- Each server must have a minimum of two Central processing unit (CPUs).
- Servers in a cluster can be heterogeneous with the following restrictions:
 - You can mix only Sun Enterprise™ 220R, Sun Enterprise 250, and Sun Enterprise 450 systems.
 - You can mix only Sun Enterprise 3500, Sun Enterprise 4500, Sun Enterprise 5500, and Sun Enterprise 6500 systems.
 - Sun Enterprise 10000 servers and Sun Fire systems should have a minimum of two system boards in each domain if possible.



Note – The Sun Fire 4800 and Sun Fire 4810 systems can have a maximum of three system boards, so there are not enough boards to meet the minimum two boards per domain recommendation.

Configuring Cluster Topologies

You can configure a Sun Cluster system in several ways. These different configurations are called topologies. Topology configurations are determined by the types of disk storage devices used in a cluster and the method by which these devices are physically connected to the cluster host systems.

Sun™ Cluster 3.0 07/01 supports the following topologies:

- Clustered pairs topology
- Pair+N topology
- N+1 topology

Clustered Pairs Topology

As shown in Figure 4-1, a clustered pairs topology is two or more pairs of nodes operating under a single cluster administrative framework. The nodes in a pair are backups for one another. In this configuration, failover occurs only between a pair. However, all nodes are connected by the private networks and operate under the control of the Sun Cluster software.

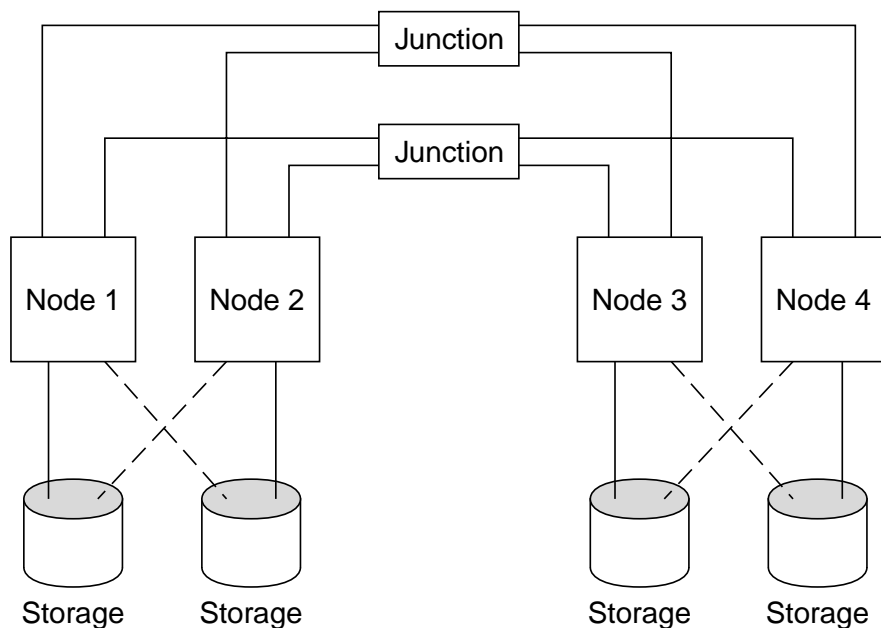


Figure 4-1 Clustered Pairs Topology Configuration

The features of clustered pairs configurations are:

- Nodes are configured in pairs. Possible configurations include either two, four, six, or eight nodes.
- Each pair of nodes shares storage. Storage is connected to both nodes in the pair.
- All nodes are part of the same cluster, simplifying administration.
- Because each pair has its own storage, no one node must have a significantly higher storage capacity than the others.
- The cost of the cluster interconnect is spread across all the nodes.
- This configuration is well suited for failover data services.

The limitation of the clustered pairs configuration is that each node in a pair cannot run at maximum capacity because they cannot handle the additional load of a failover.

Pair+N Topology

As shown in Figure 4-2, the Pair+N topology includes a pair of nodes directly connected to shared storage and nodes that must use the cluster interconnect to access shared storage because they have no direct connection themselves.

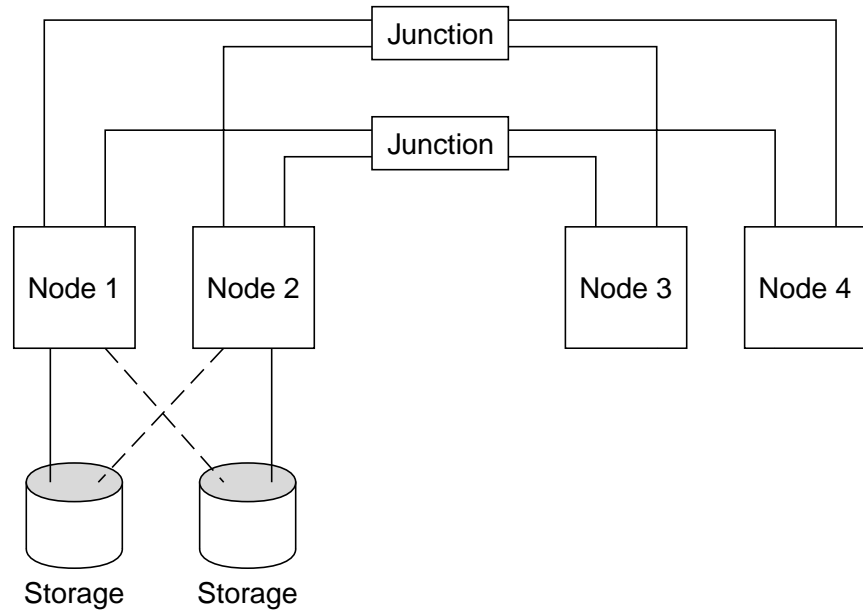


Figure 4-2 Pair+N Topology

The features of the Pair+N configurations are:

- All shared storage is connected to a single pair.
- Additional cluster nodes support scalable data services.
- A maximum of eight nodes is supported.
- There are common redundant interconnects between all nodes.
- The Pair+N configuration is well suited for scalable data services.

The limitations of a Pair+N configuration is that there can be heavy data traffic on the cluster interconnects.

Nodes without storage connections access data through the cluster interconnect using the global device feature. You can expand the number of cluster interconnect paths to increase bandwidth.

N+1 Topology

The N+1 topology, shown in Figure 4-3, enables one system to act as the backup for every other system in the cluster. All of the secondary paths to the storage devices are connected to the redundant or secondary system, which can be running a normal workload of its own.

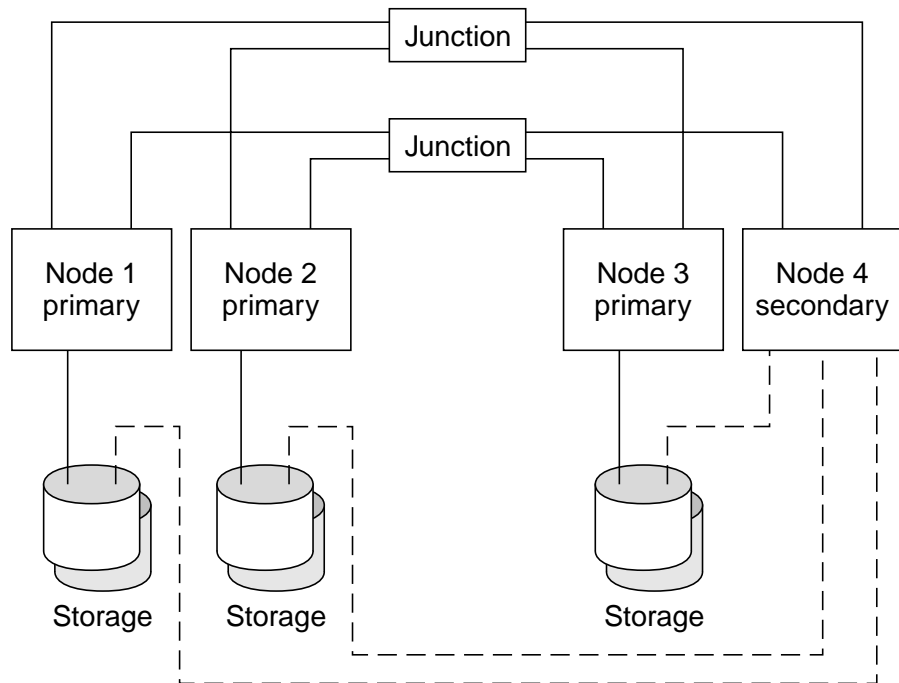


Figure 4-3 N+1 Topology Configuration

The features of the N+1 configurations are:

- The secondary node is the only node in the configuration that is physically connected to all the multihost storage
- The backup node can take over without any performance degradation.
- The backup node is more cost effective because it does not require additional data storage.
- This configuration is best suited for failover data services.

A limitation of a N+1 configuration is that if there is more than one primary node failure, you can overload the secondary node. Currently, N+1 configurations are limited to a maximum of four nodes.

Sun Fire System Configurations

The Sun Fire systems can be used in the Sun Cluster environment in several basic configuration. Clustering within a single box is supported in some Sunfire systems, but it is important to follow configuration recommendations and requirements.

Sun Fire 4800 and Sun Fire 4810 Configuration

Sun Fire 4800 and Sun Fire 4810 models have a single power grid for all the system and I/O boards. The single power grid is potentially a single point of failure when clustering within a single box. The Sun™ Fireplane interconnect system, as shown in Figure 4-4, must be segmented along domain boundaries.

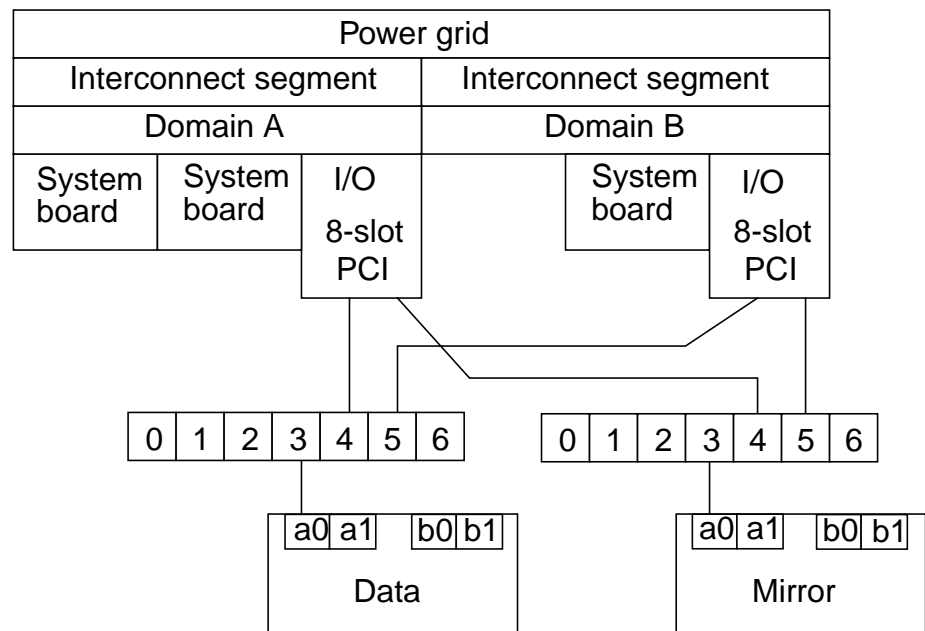


Figure 4-4 Sun Fire 4800 and Sun Fire 4810 Cluster in a Box

The Sun Fire 4800 and Sun Fire 4810 systems have the following features:

- A maximum of three system boards maximum
- Two peripheral component interface (PCI) I/O units (eight slots each)
- A maximum of two interconnect segments
- A maximum of two domains (one domain per segment)

Sun Fire 6800 Configuration

The Sun Fire 6800 is a good candidate for a cluster in a box because it has separate power grid for the odd-numbered and even-numbered board slots.

The Sun Fireplane interconnect system must be segmented along domain boundaries, as shown in Figure 4-5.

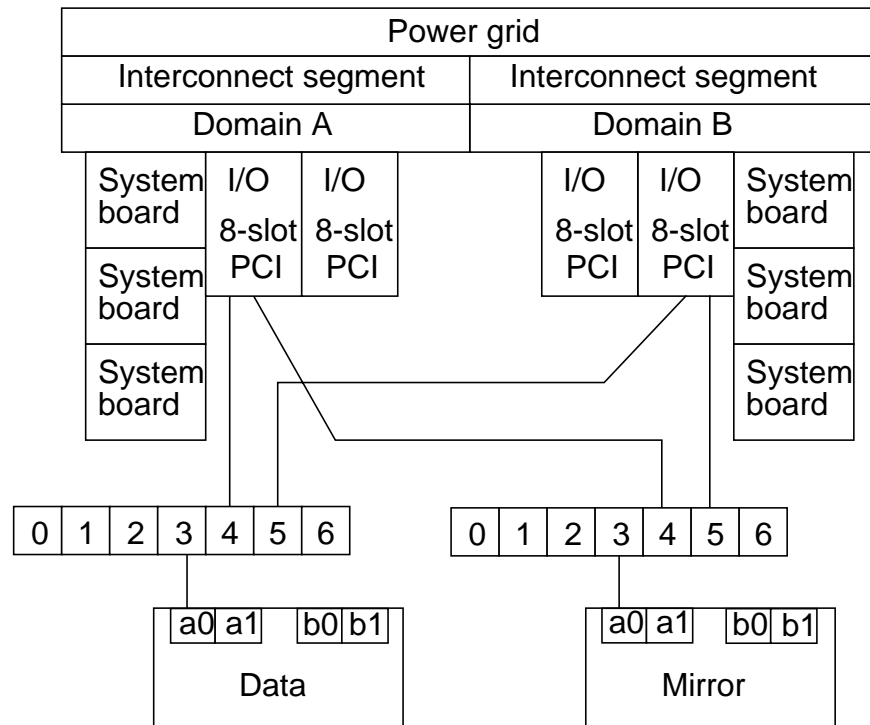


Figure 4-5 Sun Fire 6810 Cluster in a Box

The Sun Fire 6800 system has the following configuration capacity:

- A maximum of six system boards
- Four PCI I/O units (eight slots each)
- A maximum of two interconnect segments
- A maximum of four domains (two domains per segment)

Note – Each I/O unit can be associated with a domain. Multiple I/O units can be associated with one domain.



Configuring Storage

Each of the supported storage devices have configuration rules that you must follow to qualify for support. Some of the rules are common to any installation of the storage device and others are unique to the Sun™ Cluster 3.0 07/01 environment.

Sun StorEdge MultiPack Configuration

Sun StorEdge MultiPack configurations are relatively simple. The main configuration limitation is that they cannot be daisy chained. Figure 4-6 illustrates a typical Sun™ Cluster 3.0 07/01 Sun StorEdge MultiPack configuration.

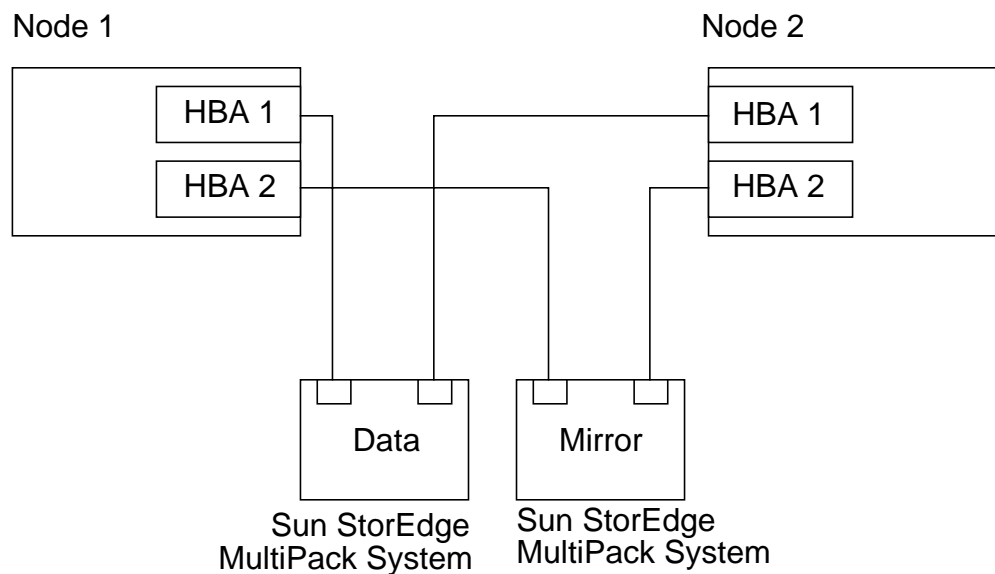


Figure 4-6 Sun StorEdge MultiPack Configuration



Note – You must change the Small Computer System Interface (SCSI) initiator ID on one of the host bus adapters (HBAs) connected to a MultiPack. The process is complex and should be performed by someone who is familiar with OpenBoot programmable read-only memory (PROM) `nvrAmrc` programming.

Infodoc 20704 has good procedures to configure multi-initiator installations.

Sun StorEdge D1000 System Configuration

Sun StorEdge D1000 system configurations have configuration restrictions similar to the MultiPack storage configuration restrictions:

- Daisy chaining is not supported.
- A single Sun StorEdge D1000 system, in a split-bus configuration, is not supported.

Figure 4-7 illustrates a typical Sun™ Cluster 3.0 07/01 Sun StorEdge D1000 system configuration.

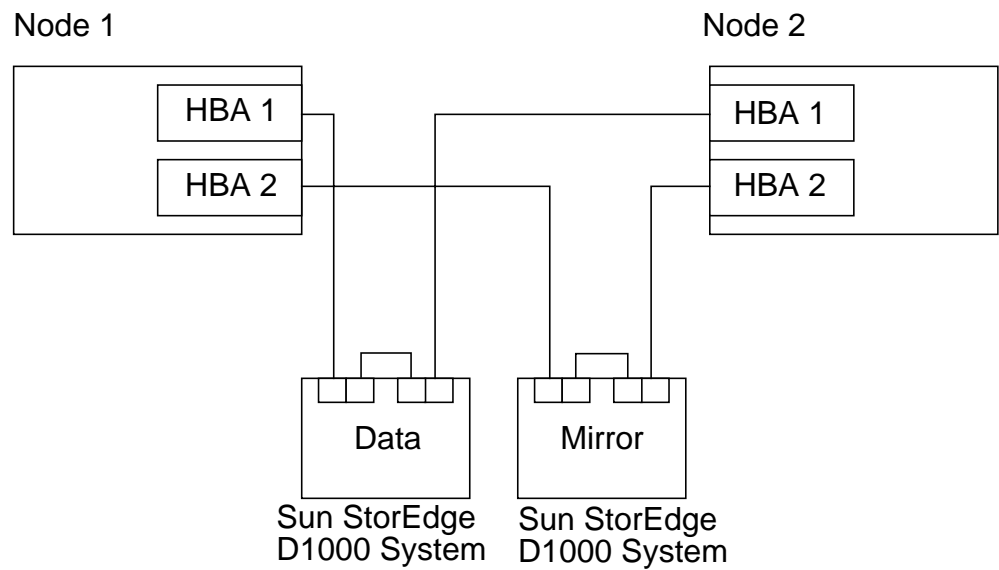


Figure 4-7 Sun StorEdge D1000 System Configuration

Sun StorEdge A3500 System Configuration

The configuration rules for using the Sun StorEdge A3500 storage array in the Sun™ Cluster 3.0 07/01 environment are as follows:

- Daisy chaining of the controller modules is not supported.
- A Sun StorEdge A3500 storage array with a single controller module is supported.
- The Sun StorEdge A3500 Lite system is supported.
- You must connect the two ports of a controller module to two different hosts.
- You cannot use the Sun StorEdge A3500 system disks as quorum devices.
- Only SBus-based host bus adapters are supported.

Figure 4-8 illustrates a typical Sun™ Cluster 3.0 07/01 Sun StorEdge A3500 system configuration.

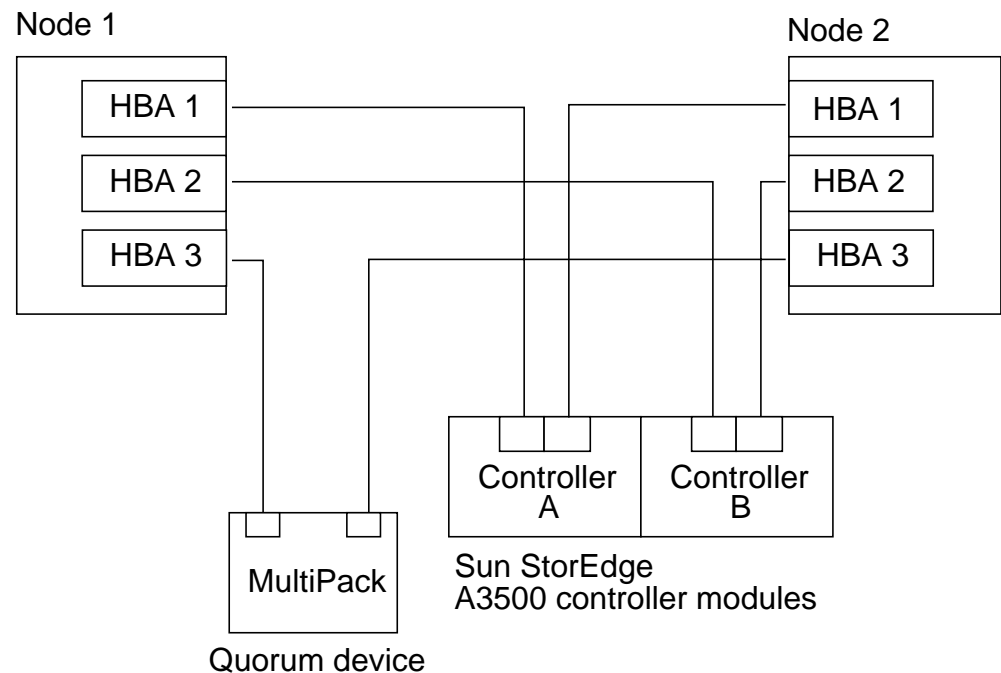


Figure 4-8 Sun StorEdge A3500 System Configuration

Sun StorEdge A3500FC System Configuration

The configuration rules for using the Sun StorEdge A3500FC storage array in the Sun[™] Cluster 3.0 07/01 environment are as follows:

- Daisy chaining of the controller modules is not supported.
- A Sun StorEdge A3500FC storage array with a single controller module is supported.
- A controller module must be connected to two different hosts.
- The Sun StorEdge A3500FC system disks cannot be used as a quorum device.
- Fibre-Channel hubs are required in the cluster environment.
- Currently, only Sbus-based fibre-Channel host bus adapters are supported for use with the Sun StorEdge A3500FC array.

Figure 4-8 illustrates a typical Sun[™] Cluster 3.0 07/01 Sun StorEdge A3500FC system configuration.

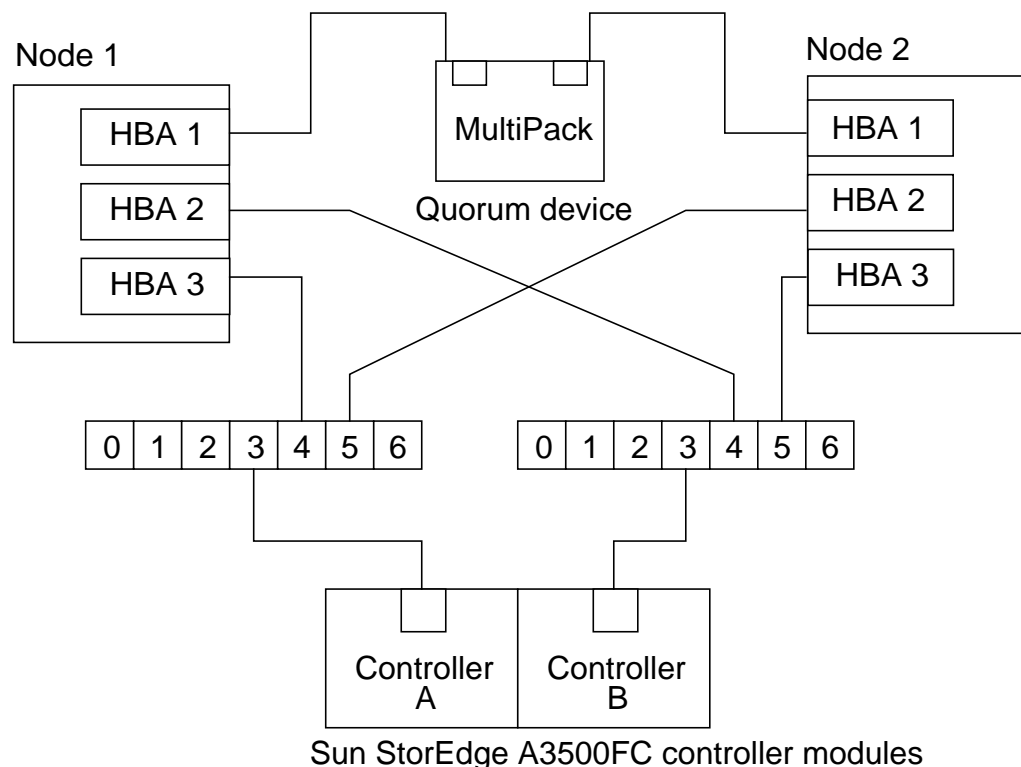


Figure 4-9 Sun StorEdge A3500FC System Configuration

Sun StorEdge A5x00 System Configurations

The Sun StorEdge A5000, Sun StorEdge 5100, Sun StorEdge 5200 storage arrays have two primary restrictions when used in the Sun™ Cluster 3.0 07/01 environment:

- A maximum of two host connections per loop.

A full-loop configured Sun StorEdge A5x00 array can have only two host system connections and each connection must be made through a different Sun StorEdge A5x00 system interface board (IB).

A split-loop configured Sun StorEdge A5x00 system can have two host system connections to each IB.

- PCI-based Fibre Channel-100 interface boards must be connected to Sun StorEdge A5x00 storage arrays through Fibre Channel-Arbitrated Loop (FC-AL) hubs (hub-attached).

SBus-based Fibre Channel-100 (FC-100) interface boards are attached directly to Sun StorEdge A5x00 storage arrays (direct-attached).

- Daisy chaining of Sun StorEdge A5x00 storage arrays is not supported.

Direct-Attached Full-Loop Sun StorEdge A5x00 Configuration

The configuration shown in Figure 4-10 is typical of a two-node failover cluster with two hosts attached to each storage array.

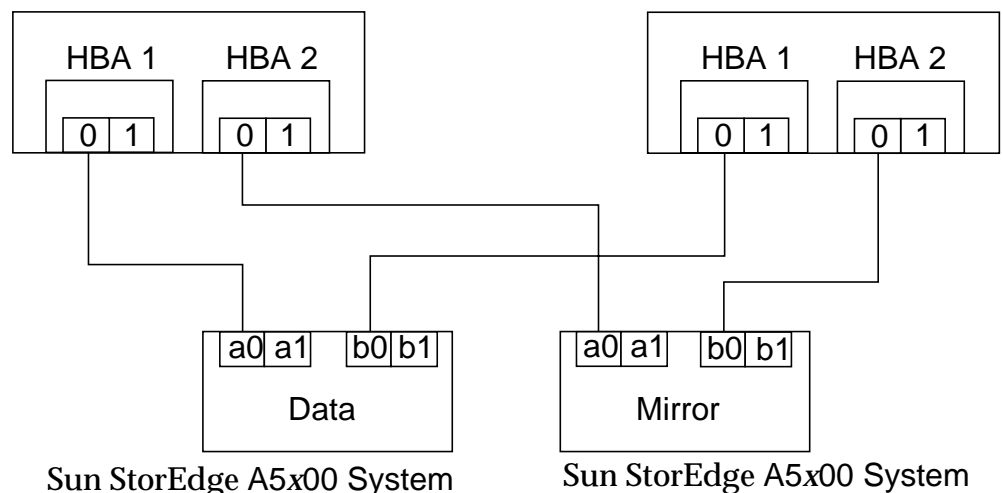


Figure 4-10 Direct-Attached Full-Loop Sun StorEdge A5x00 System

Direct-Attached Split-Loop Sun StorEdge A5x00 Configuration

The direct-attached split-loop Sun StorEdge A5x00 system configuration shown in Figure 4-11 is useful when you need a smaller amount of storage. Do not mirror data within a storage array.

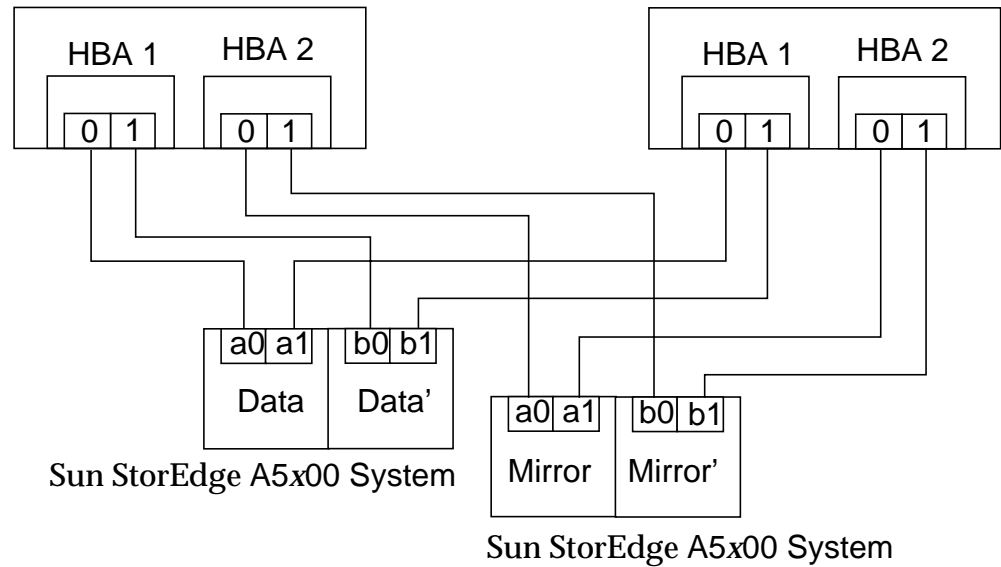


Figure 4-11 Direct-Attached Split-Loop Sun StorEdge A5x00 System



Note – Currently, PCI-based Fibre Channel host bus adapters cannot be directly attached to the Sun StorEdge A5x00 storage arrays. You must use Fibre Channel hubs.

Hub-Attached Full-Loop Sun StorEdge A5x00 Configuration

The hub-attached full-loop Sun StorEdge A5x00 system configuration is shown in Figure 4-12. Fibre Channel hubs are required when connecting PCI-based host has adapters to Sun StorEdge A5x00 storage arrays.

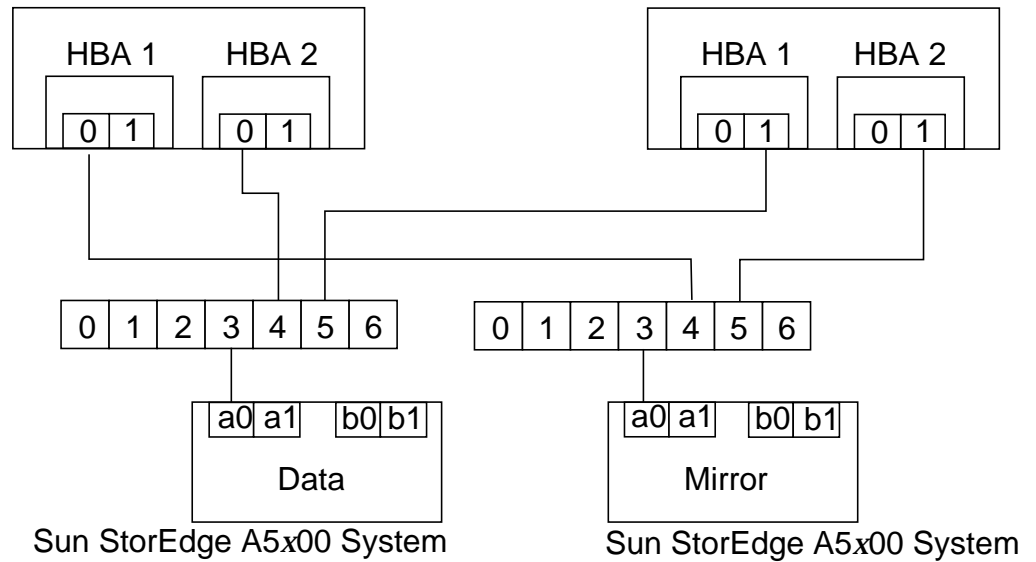


Figure 4-12 Hub-Attached Full-Loop Sun StorEdge A5x00 System

Sun StorEdge T3 System Configurations

The Sun StorEdge T3 storage can be used with most supported Sun Cluster systems that uses PCI bus Fibre-Channel host bus adapters.

Because the Sun StorEdge T3 storage arrays have only a single host connection, they must be used in pairs with either Fibre-Channel hubs or the Sun StorEdge Network Fibre-Channel switch. Data must be mirrored across arrays to preserve cluster availability. Currently, the switches must be used in a fashion similar to the hubs.

Single-Brick Configuration

As shown in Figure 4-13, the Sun StorEdge T3 storage arrays can be configured as two single-single brick units using hubs or switches.

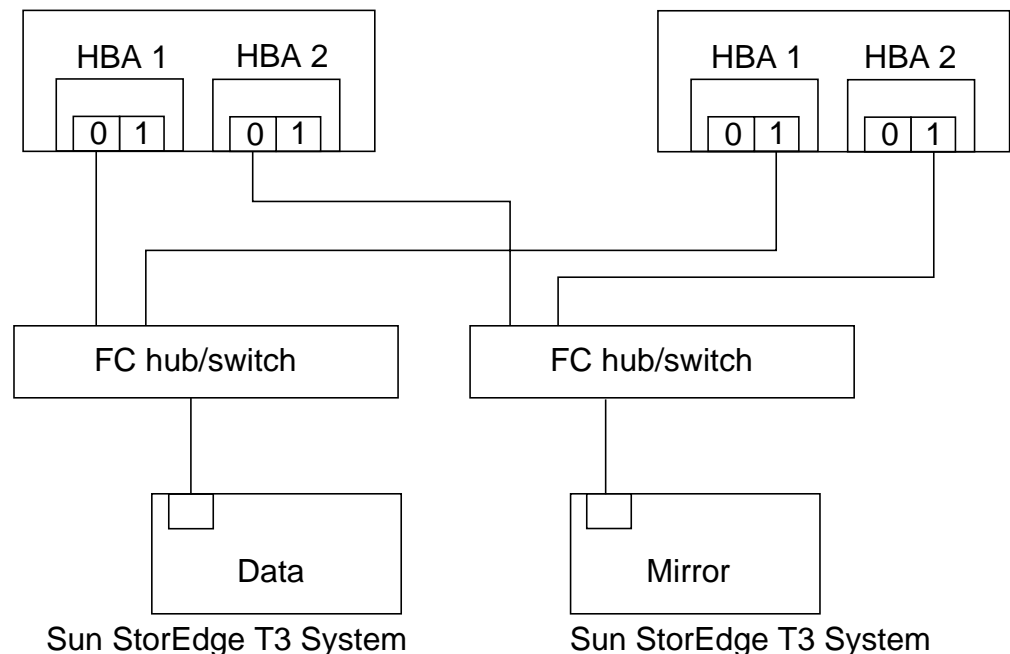


Figure 4-13 Sun StorEdge T3 System Configuration



Note – Check the current Sun Cluster release note addendum for up-to-date information about supported Sun StorEdge T3 array configurations.

Partner-Pair Configuration

As shown in Figure 4-14, a Sun StorEdge T3 partner-pair configuration uses a special interconnect cable to provide data cache mirroring and, in some failures, can provide an alternate controller path.

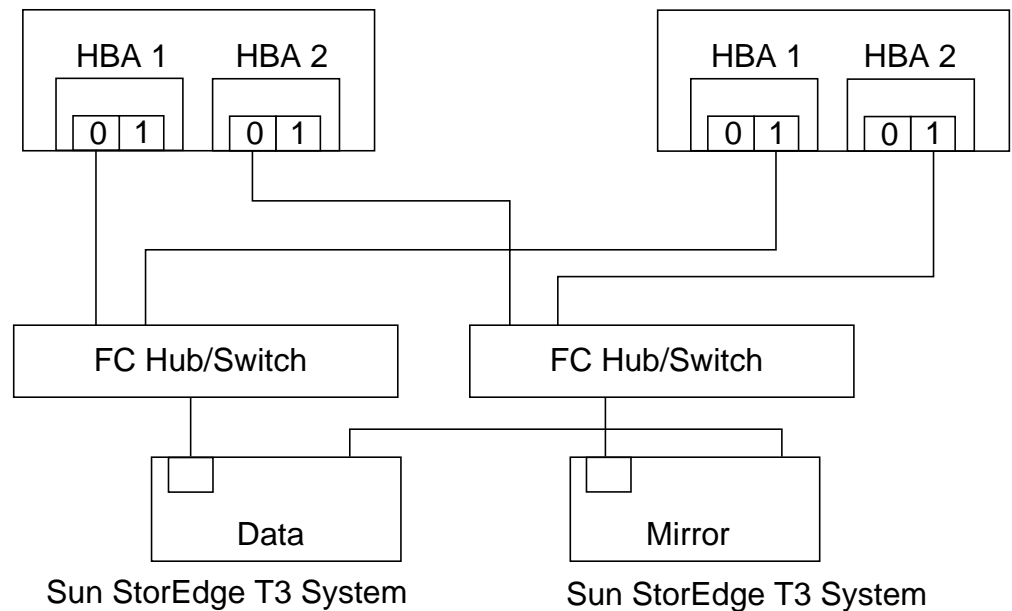


Figure 4-14 Sun StorEdge T3 Partner Pair-Configuration

Currently, the following limitations apply to partner-pair configurations when used in the Sun Cluster environment:

- Use is limited to selected PCI-based servers in two-node configurations
- Host bus adapter cards are limited to X6727A or X6799A options
- Fibre-Channel switches must operate in transparent mode
- A minimum of a Solaris 8 Operating Environment update 4 release
- Solstice DiskSuite 4.2.1
- VERITAS Volume Manager 3.2
- Storage Area Network (SAN) foundation software and Multiplexed I/O (MPxIO) software must be installed
- Storage array firmware restrictions

Note – Check the current Sun Cluster release notes for up-to-date configuration support.



Cluster Interconnect Configuration

There are two variations of cluster interconnects: point-to-point and junction-based. In a junction-based interconnect, the junctions must be switches and not hubs.

Point-to-Point Cluster Interconnect

In a two-node cluster, you can directly connect interconnect interfaces using crossover cables. A point-to-point interconnect configuration using 100BASE-T interfaces is illustrated in Figure 4-15.

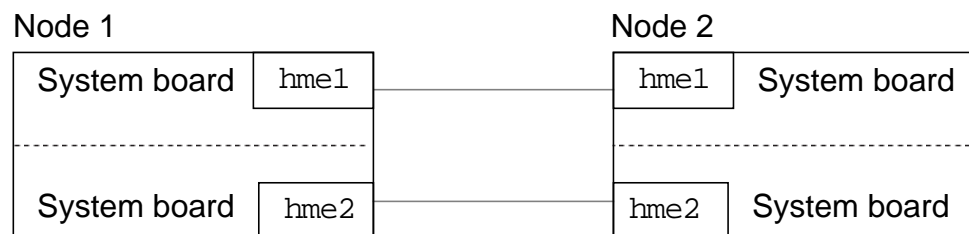


Figure 4-15 Point-to-Point Cluster Interconnect

During the Sun™ Cluster 3.0 07/01 installation, you must provide the names of the end-point interfaces for each cable.



Caution – If you provide the wrong interconnect interface names during the initial Sun Cluster installation, the first node is installed without errors, but when you try to install the second node, the installation hangs. You have to correct the cluster configuration error on the first node and then restart the installation on the second node.

Junction-based Cluster Interconnect

In cluster configurations greater than two nodes, you must join the interconnect interfaces using switches. You can also use switches to join two-node cluster interconnects to prepare for the expansion of the number of nodes at a later time. A typical junction-based interconnect is illustrated in Figure 4-16 on page 4-22.

During the Sun™ Cluster 3.0 07/01 software installation, you are asked whether the interconnect system uses junctions. If you answer yes, you must provide names for each of the switches.

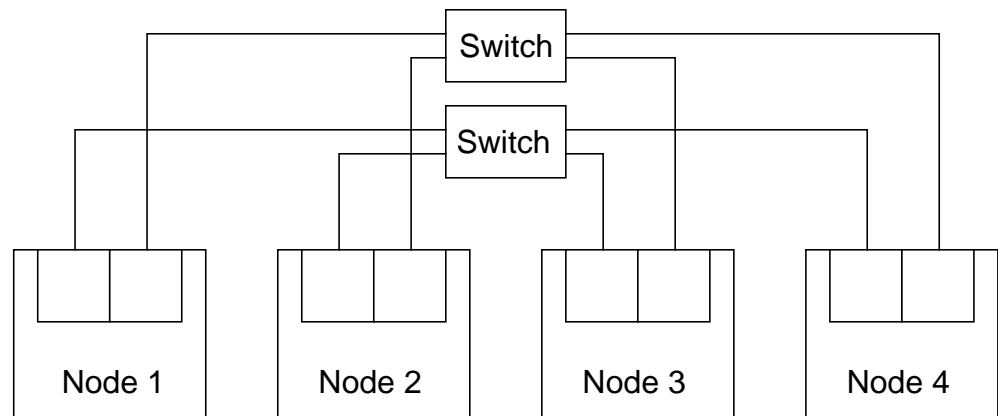


Figure 4-16 Junction-based Cluster Interconnect



Note – If you specify more than two nodes during the initial portion of the Sun Cluster software installation, the use of junctions is assumed.

Cluster Transport Interface Addresses

During the Sun Cluster software installation, the cluster interconnect are assigned Internet Protocol (IP) addresses based on a base address of 172.16.0.0. If necessary, you can override the default address, but this is not recommended. Uniform addresses can be a benefit during problem isolation.

Identifying Cluster Transport Interfaces

Identifying network interfaces is not an easy task. To accurately determine the logical name of each interface on a system, use the `prtconf` command to complete the following steps:

1. Look for network interfaces in the `prtconf` command output.

Typical instances you might see are network instance #0, `SUNW,hme instance #1`, and `SUNW,hme instance #2`.

2. Verify which interfaces are already up (plumbed).

```
# ifconfig -a
lo0:
flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4>
mtu 8232 index 1 inet 127.0.0.1 netmask ff000000
hme0:
flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
mtu 1500 index 2 inet 129.200.9.2 netmask ffffffff0
broadcast 129.200.9.255 ether 8:0:20:96:3:86
```

3. Bring up the unplumbed interfaces for testing.

```
# ifconfig hme1 plumb
# ifconfig hme1 up
# ifconfig hme2 plumb
# ifconfig hme2 up
```

4. Verify that the new interfaces are up.

```
# ifconfig -a
```

5. Test each of the new interfaces while plugging and unplugging them from an active network.

```
# ping -i hme1 pnode2
pnode2 is alive
# ping -i hme2 pnode2
pnode2 is alive
```

Note – You should create configuration worksheets and drawings of your cluster configuration.



6. After you have identified the new network interfaces, bring them down again.

```
# ifconfig hme1 down
# ifconfig hme2 down
# ifconfig hme1 unplumb
# ifconfig hme2 unplumb
```

Caution – Be careful to not bring down the primary system interface.



Eliminating Single Points of Failure

A single point of failure is any hardware or software configuration item that can completely eliminate access to data if it fails.

An example of a software-oriented single point of failure is creating redundant array of independent disks (RAID)-1 mirroring within a single storage array. If the array has a major failure, you lose access to the data.

There are also practices that increase data availability but are not related to single points of failure.

Each of the following rules describe a best practice that you should use whenever possible in a clustered configuration:

- RAID-1 mirrors should reside in different storage arrays.
If an array fails, one of the mirrors is still available.
- Host bus adapters should be distributed across system boards.
A single system board failure should not disable access to both copies of mirrored data.
- Order equipment with optional redundancy, if possible.
Many system and storage array models have optional redundant power supplies and cooling. Attach each power supply to a different power circuit.
- Redundant cluster interconnects are *required*, not optional.
- Uninterruptable power supply (UPS) systems and/or local power generators can increase overall cluster availability.

Although expensive, UPS systems, local power generators, both can be worthwhile in critical cluster applications.

Cluster Quorum Device Configuration

Because cluster nodes share data and resources, the cluster must take steps to maintain data and resource integrity. The concept of quorum voting controls cluster membership.

Each node in a cluster is assigned a vote. To form a cluster and for the cluster to remain up, a majority of votes must be present. To form a two-node cluster, for example, a majority of votes would be two. Without some modification to a two-node cluster, both nodes would have to be booted before a cluster could form. An additional problem is that the cluster cannot continue if a single node fails. This is a single point of failure that defeats the high-availability requirement.

This problem is resolved by assigning a vote to a disk drive called a *quorum device*, which is assigned a single vote. Now when a single node tries to come up, it reserves the quorum device. There is now a majority of two votes out of three possible.

Quorum devices are also used for another purpose: *failure fencing*. As shown in Figure 4-17, if interconnect communication between nodes ceases, either because of a complete interconnect failure or a node crashing, each node must assume the other is still functional. This is called *split-brain* operation. Two separate clusters cannot be allowed to exist because of the potential for data corruption. Each node tries to establish a cluster by gaining another quorum vote. Both nodes attempt to reserve the designated quorum device. The first node to reserve the quorum disk establishes a majority and remains as a cluster member. The node that fails the race to reserve the quorum device aborts the Sun Cluster software because it does not have a majority of votes.

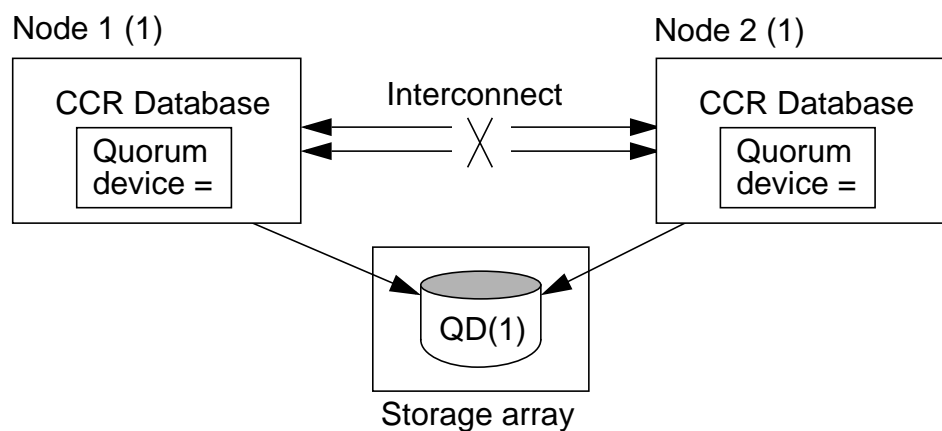


Figure 4-17 Failure Fencing

Quorum Device Rules

The general rules for quorum devices are:

- A quorum device must be available to both nodes in a two-node cluster.
- Quorum device information is maintained globally in the cluster configuration repository (CCR) database.
- A quorum device can contain user data.
- The *maximum* number of votes contributed by quorum devices should be the number of node votes minus one (N-1).

If the number of quorum devices equals or exceeds the number of nodes, the cluster cannot come up if too many quorum devices fail.

- Quorum devices are not required in clusters with greater than two nodes, but they are recommended for higher cluster availability.
- Quorum devices are manually configured after the Sun Cluster software installation is complete.
- Quorum devices are configured using DID devices and are available only to directly attached nodes.

Two-Node Cluster Quorum Devices

As shown in Figure 4-18, a two-node cluster needs a single quorum disk. The total votes are three. With the quorum disk, a single node can start clustered operation with a majority of votes (2).

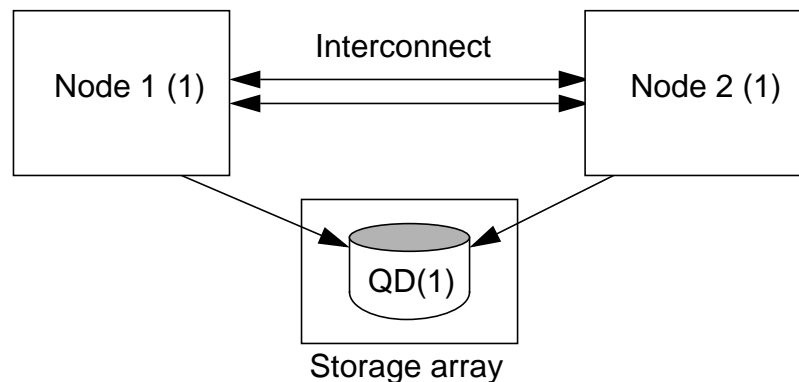


Figure 4-18 Two-Node Cluster Quorum Devices

Clustered-Pair Quorum Disks

In a clustered-pairs configuration, shown in Figure 4-19, there are always an even number of cluster nodes (2, 4, 6, 8). The nodes in each pair usually provide data service failover backup for one another.

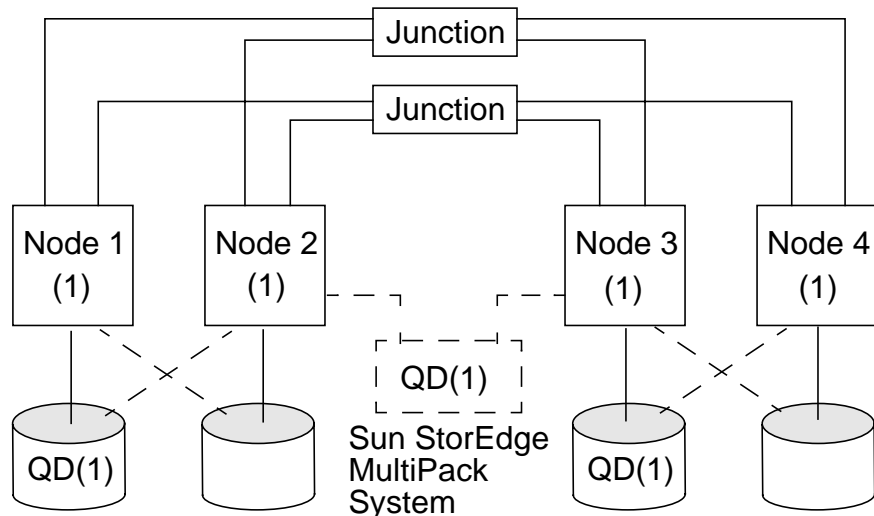


Figure 4-19 Clustered-Pair Quorum Devices

There are many possible split-brain scenarios. Not all of the possible split-brain combinations allow the continuation of clustered operation. The following is true for a clustered pair configuration:

- There are six possible votes.
- A quorum is four votes.
- If both quorum devices fail, the cluster can still come up.

The nodes wait until all are present (booted).

- If Nodes 1 and 2 fail, there are not enough votes for Nodes 3 and 4 to continue running

A token quorum device between Nodes 2 and 3 can eliminate this problem. A Sun StorEdge MultiPack System could be used for this purpose.

- A node in each pair can fail, and there are still four votes.

Pair+N Quorum Disks

Figure 4-20 illustrates a typical quorum disk configuration in a Pair+2 configuration. Three quorum disks are used. You use this configuration for scalable data services.

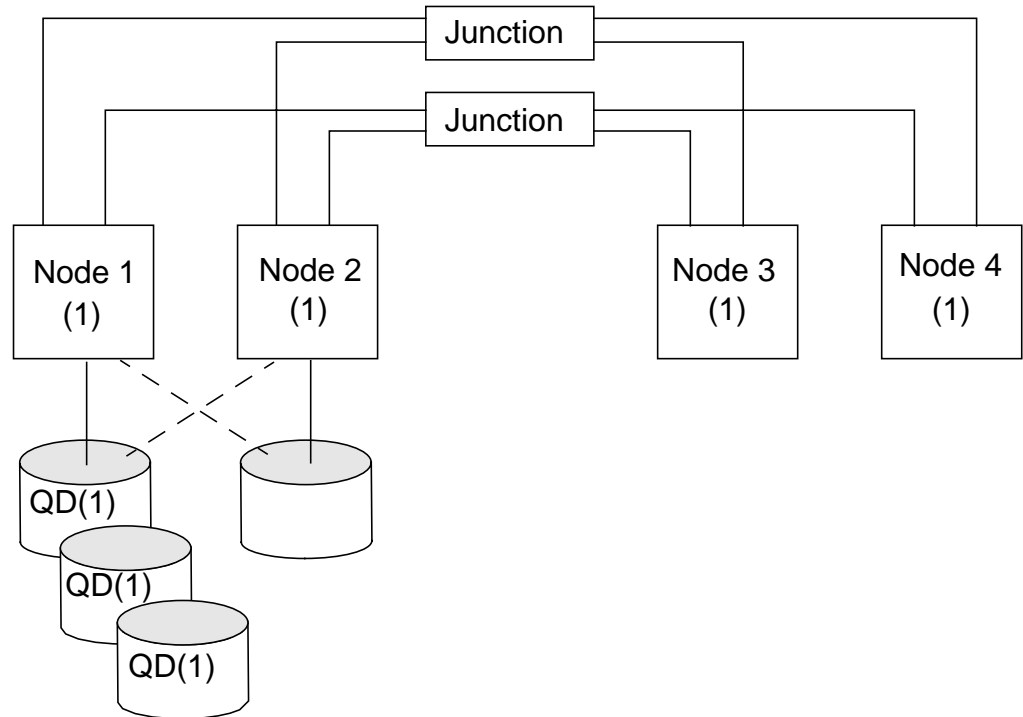


Figure 4-20 Pair+N Quorum Devices

The following is true for the Pair+N configuration shown in Figure 4-20:

- There are three quorum disks.
- There are seven possible votes.
- A quorum is four votes.
- Nodes 3 and 4 do not have access to any quorum devices.
- Nodes 1 or 2 can start clustered operation by themselves.
- Up to three nodes can fail (1, 3, and 4 or 2, 3, and 4), and clustered operation can continue.

N+1 Quorum Disks

The N+1 configuration shown in Figure 4-21 requires a different approach. Node 3 is the failover backup for both Node 1 and Node 2.

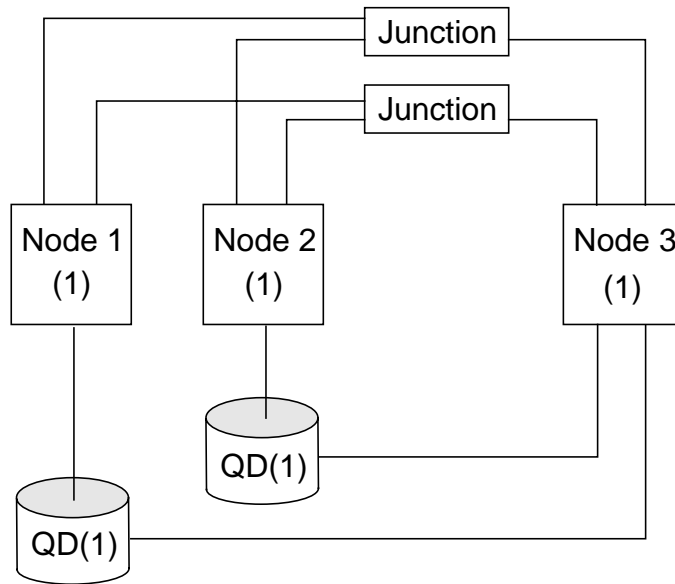


Figure 4-21 N+1 Quorum Devices

The following is true for the N+1 configuration shown in Figure 4-21:

- There are five possible votes.
- A quorum is three votes.
- If Nodes 1 and 2 fail, Node 3 can continue.

Public Network Configuration

The Public Network Measurement (PNM) software creates and manages designated groups of local network adapters commonly referred to as network adapter failover (NAFO) groups. If a cluster host network adapter fails, its associated IP address is transferred to a backup adapter in its group.

As shown in Figure 4-22, the PNM daemon (`pnmd`) continuously monitors designated network adapters on a single node. If a failure is detected, `pnmd` uses information in the CCR and the `pnmconfig` file to initiate a failover to a healthy adapter in the backup group.

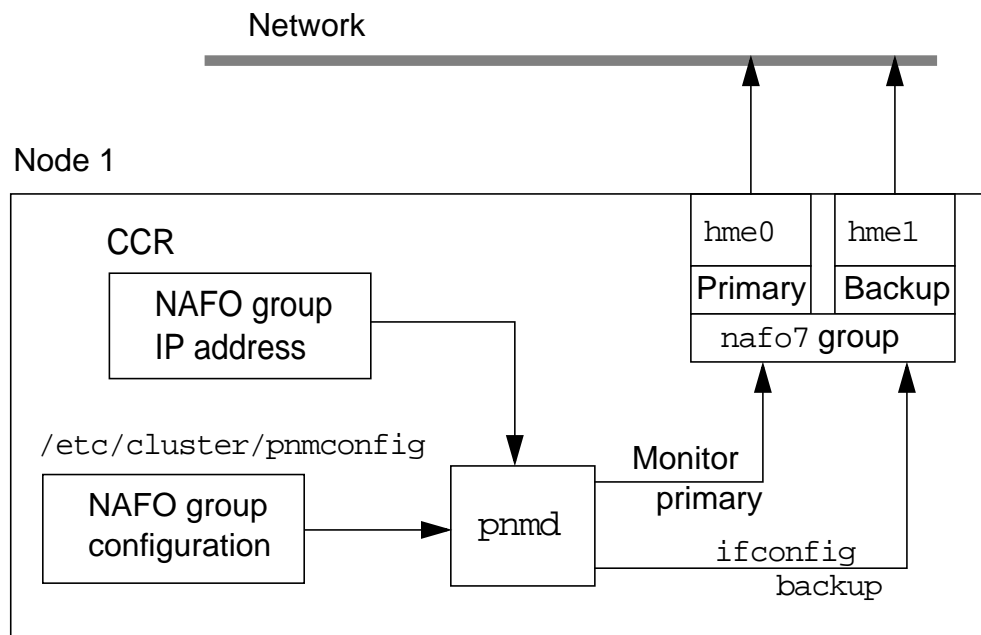


Figure 4-22 PNM Components

The adapters within a group must meet the following criteria:

- All adapters in a NAFO group must be of the same media type.
- The adapters can be 100-Mbit/second or Gigabit Ethernet.

Identifying Storage Array Firmware

Storage array installations have significant amounts of firmware. A typical Sun StorEdge A5x00 system installation has firmware in the following locations:

- In each host bus adapter (HBA)
- In each array interface board (IB)
- In many array disk drive models
- Sun StorEdge T3 arrays have their own unique firmware scheme but still rely on HBA firmware in some configurations.

Most of the firmware revisions can be verified using the `luxadm` command. The `luxadm` command is a standard Solaris Operating Environment command.



Note – The firmware upgrade process is complex and can permanently disable storage array interfaces unless performed correctly. You should contact your Sun field representative about firmware-related questions.

Identifying Attached Sun StorEdge A5x00 Storage Arrays

The `luxadm probe` option displays basic information about all attached Sun StorEdge A5x00 storage arrays.

```
# /usr/sbin/luxadm probe
Found Enclosure(s):
SENA          Name:AA   Node WWN:5080020000034ed8
  Logical Path:/dev/es/ses1
  Logical Path:/dev/es/ses3
SENA          Name:BB   Node WWN:5080020000029e70
  Logical Path:/dev/es/ses6
  Logical Path:/dev/es/ses7
```

Identifying Host Bus Adapter Firmware

The `luxadm` command can display information about HBA firmware revisions for the FC/S, FC100/S, and FC100/P Fibre-Channel cards.

To check firmware revisions on any type of Fibre-Channel HBA, type:

```
# luxadm fcode_download -p
```



Note – The older FC/S SBus cards are not supported for use with Sun™ Cluster 3.0 07/01 and Sun StorEdge A5x00 storage arrays.

An example of the command output follows.

```
# /usr/sbin/luxadm fcode_download -p
```

```
Found Path to 0 FC/S Cards  
Complete
```

```
Found Path to 0 FC100/S Cards  
Complete
```

```
Found Path to 2 FC100/P, ISP2200 Devices
```

```
Opening Device: /devices/pci@6,4000/SUNW,ifp@2:devctl  
Detected FCode Version:      FC100/P FC-AL Host  
Adapter Driver: 1.9 00/03/10
```

```
Opening Device: /devices/pci@6,4000/SUNW,ifp@3:devctl  
Detected FCode Version:      FC100/P FC-AL Host  
Adapter Driver: 1.9 00/03/10  
Complete
```



Caution – The same `luxadm` command option both checks and downloads firmware.

Identifying Sun StorEdge A5x00 Interface Board Firmware

The Sun StorEdge A5x00 storage arrays can have two interface boards. You can check their firmware revisions as follows:

```
# luxadm display enclosure_name
```



Note – Sun StorEdge A5x00 storage arrays are usually assigned unique enclosure names from their light-emitting diode (LED) display panel.

A typical luxadm display output follows.

```
# luxadm display BB
```

```

                                SENA
                                DISK STATUS
SLOT   FRONT DISKS             (Node WWN)           REAR DISKS       (Node WWN)
0       On (O.K.)              20000020370c2d2b   Not Installed
1       On (O.K.)              20000020370cbc90   Not Installed
2       Not Installed
3       On (O.K.)              20000020370d6570   Not Installed
4       Not Installed
5       On (O.K.)              20000020370d6940   Not Installed
6       Not Installed
                                SUBSYSTEM STATUS
FW Revision:1.09   Box ID:1   Node WWN:5080020000029e70   Enclosure
Name:BB
Power Supplies (0,2 in front, 1 in rear)
. . . . .
. . . . .
. . . . .
Loop configuration
    Loop A is configured as a single loop.
    Loop B is configured as a single loop.
Language           USA English
```



Note – Although there can be two interface boards in a Sun StorEdge A5x00 storage array, the firmware displays as a single value and both boards load automatically during firmware upgrade procedures.

Identifying Sun StorEdge T3 Array Firmware

The Sun StorEdge T3 arrays have internal configuration and control software (pSOS) that can verify firmware revisions. You must telnet to a particular array, log in as user root, and use the ver (version) command to verify the current firmware revision.

An example of a typical session follows.

```
$telnet t3
Trying 129.150.47.115...
Connected to purple15.
Escape character is '^]'.

pSOSystem (129.150.47.115)

Login: root
Password:
T300 Release 1.00 1999/12/15 16:55:46 (129.150.47.115)

t3:/:<1> ver

T300 Release 1.14 1999/12/15 16:55:46 (129.150.47.115)
```

Note – In the example, the firmware version is 1.14.



Exercise: Preinstallation Preparation

In this exercise, you complete the following tasks:

- Configure a cluster topology
- Identify quorum devices needed
- Verify the cluster interconnect cabling
- Verify storage array firmware revisions

Preparation

To begin this exercise, you must be connected to the cluster hosts through the `cconsole` tool, and you are logged into them as user `root`.

Your assigned cluster should be similar to the configuration shown in Figure 4-23:

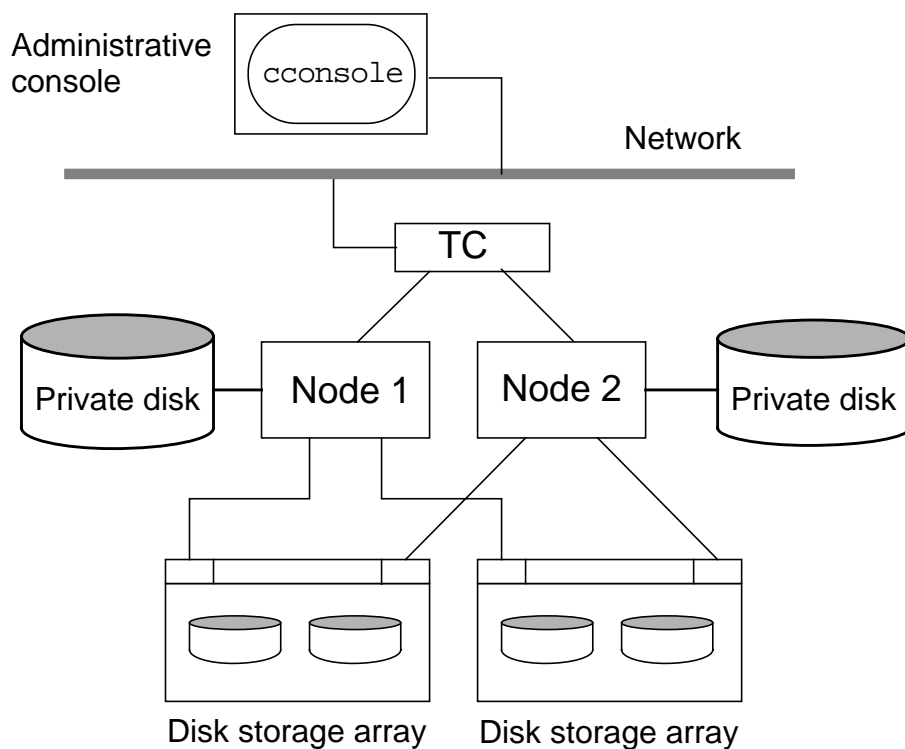


Figure 4-23 Cluster Configuration



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Verifying the Solaris Operating Environment

In this section, you verify that the boot disk is correctly partitioned on all nodes.

1. Type the `/etc/prtconf` command on each node and record the size of physical memory (`/etc/prtconf |grep Memory`).

Node 1 memory: _____

Node 2 memory: _____

2. Type the `swap -l` command on each node and verify that the swap space is at least twice the size of memory.

Note – Divide the number of blocks by 2000 to approximate the size of the swap space in megabytes.

3. Type the `df -k1` command on each node and verify that there is a 100-Mbyte `/globaldevices` file system mounted.

Task – Identifying a Cluster Topology

1. Record the desired topology configuration of your cluster.

Topology Configuration	
Number of nodes	
Number of storage arrays	
Types of storage arrays	

2. Verify that the storage arrays in your cluster are properly connected for your target topology. Recable the storage arrays if necessary.

Task – Selecting Quorum Devices

1. Record the estimated number of quorum devices you must configure after the cluster host software installation.

Estimated number of quorum devices: _____



Note – Please consult with your instructor if you are not sure about your quorum device configuration.

2. Type the **format** command and record the logical path to a suitable quorum disk drive in one of your storage arrays.

Quorum Disk(s): _____

2. Type Control-D to cleanly exit the `format` utility.

Task – Verifying the Cluster Interconnect Configuration

This task describes how to verify the cluster interconnect configuration.

Configuring a Point-to-Point Ethernet Interconnect

Skip this section if your cluster interconnect is not point-to-point.

1. Ask your instructor for assistance in determining the logical names of your cluster interconnect interfaces.
2. Complete the form in Figure 4-24 if your cluster uses an Ethernet-based point-to-point interconnect configuration.

Node 1			Node 2	
Primary interconnect interface				Primary interconnect interface
Secondary interconnect interface				Secondary interconnect interface

Figure 4-24 Ethernet Interconnect Point-to-Point Form

Configuring a Switch-based Ethernet Interconnect

Perform the following steps to configure a switch-based Ethernet interconnect:

1. Complete the form in Figure 4-25 if your cluster uses an Ethernet-based cluster interconnect with switches. Record the logical names of the cluster interconnect interfaces (`hme2`, `qfe1`, ...).

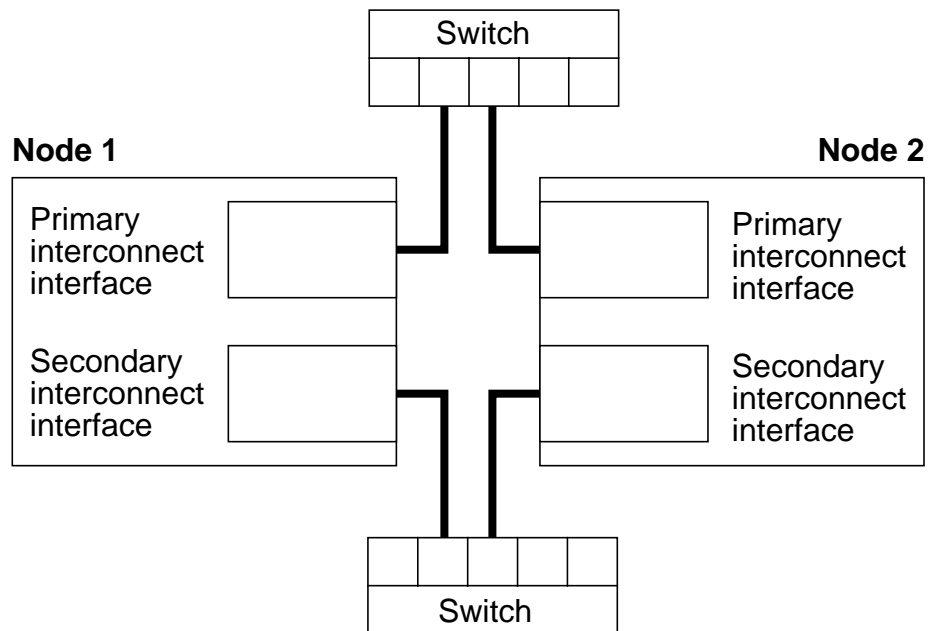


Figure 4-25 Ethernet Interconnect With Hubs Form

2. Verify that each Ethernet interconnect interface is connected to the correct hub.



Note – If you have any doubt about the interconnect cabling, consult with your instructor now. Do not continue this lab until you are confident that your cluster interconnect system is cabled correctly.

Task – Verifying Storage Array Firmware Revisions

Perform the following steps to verify Storage array firmware revisions:

1. If your assigned cluster uses Sun StorEdge A5x00 storage arrays, enter the **luxadm probe** command to identify the names of attached storage arrays. Record the enclosure names.

Enclosure name: _____

Enclosure name: _____

2. Enter the **luxadm fcode_download -p** command to display the firmware revision of any Fibre-Channel HBA cards. Record the HBA firmware revisions.

HBA firmware revision: _____

HBA firmware revision: _____

3. Enter the **luxadm display enclosure_name** command to display the firmware revision of each Sun StorEdge A5x00 storage array.

Array name: _____ Firmware: _____

Array name: _____ Firmware: _____

4. If your assigned cluster uses Sun StorEdge T3 storage arrays, enter the **telnet** command to log in to each array as user `root`. Type the **ver** command to display the firmware revision of each array. Log out of the arrays when finished.

Array name: _____ Firmware: _____

Array name: _____ Firmware: _____

Note – Ask your instructor for the names or IP addresses of the T3 storage arrays assigned to your cluster.



Task – Selecting Public Network Interfaces

Ask for help from your instructor in identifying public network interfaces on each node that can be used in PNM NAFO groups.

Although it is not supported, you can configure a single interface in a NAFO group. Standard training practice is to configure the primary system interface in the NAFO group.

Perform the following steps to select public network interfaces:

1. Record, in Table 4-2, the logical names of potential PNM Ethernet interfaces on each node.

Table 4-3 Logical Names of Potential PNM Ethernet Interfaces

System	Primary NAFO interface	Backup NAFO interface
Node 1		
Node 2		



Note – It is important that you are sure about the logical name of each NAFO interface (hme2, qfe3, and so on).

2. Verify that the target NAFO interfaces on each node are connected to a public network.

Exercise Summary

Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ List the Sun Cluster boot disk requirements
- ☐ Physically configure a cluster topology
- ☐ Configure a supported cluster interconnect system
- ☐ Identify single points of failure in a cluster configuration
- ☐ Identify the quorum devices needed for selected cluster topologies
- ☐ Verify storage firmware revisions
- ☐ Physically configure a public network group

Think Beyond

What additional preparation might be necessary before installing the Sun Cluster host software?

Installing the Cluster Host Software

Objectives

Upon completion of this module, you should be able to:

- Install the Sun Cluster host system software
- Correctly interpret configuration questions during the Sun Cluster software installation
- Perform postinstallation configuration

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- What configuration issues might control how the Sun Cluster software is installed?
- What type of postinstallation tasks might be necessary?
- What other software might you need to finish the installation?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Sun Cluster Software Summary

Sun Cluster software is installed on a Sun Cluster hardware platform. The complete Sun™ Cluster 3.0 07/01 software collection shown in Figure 5-1, consists of the following CD-ROMs:

- Sun™ Cluster 3.0 07/01 CD-ROM
- Sun™ Cluster 3.0 07/01 Data Services CD-ROM

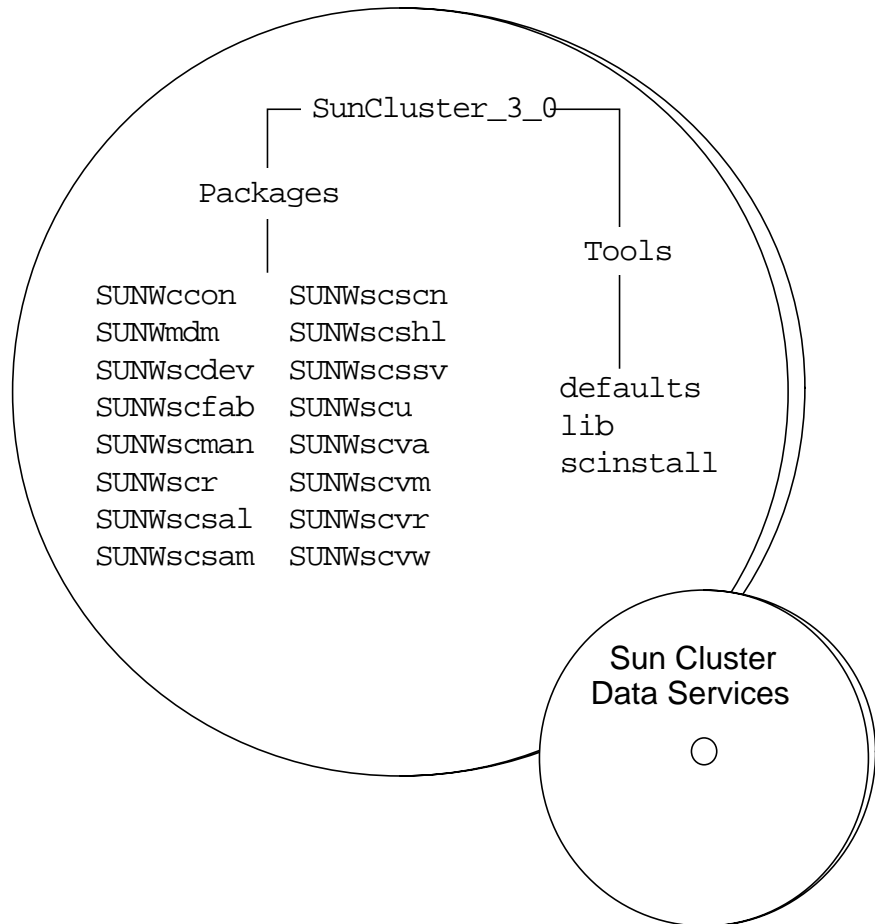


Figure 5-1 Sun Cluster CD-ROM Collection

The `scinstall` utility in the `Tools` directory is used to install the Sun Cluster software on cluster nodes.

Note – You might also need Solstice DiskSuite or Veritas Volume Manager software.



Sun Cluster Software Distribution

As shown in Figure 5-2, you install the Sun Cluster server software on each of the cluster host systems along with the appropriate data service software and virtual volume management software.

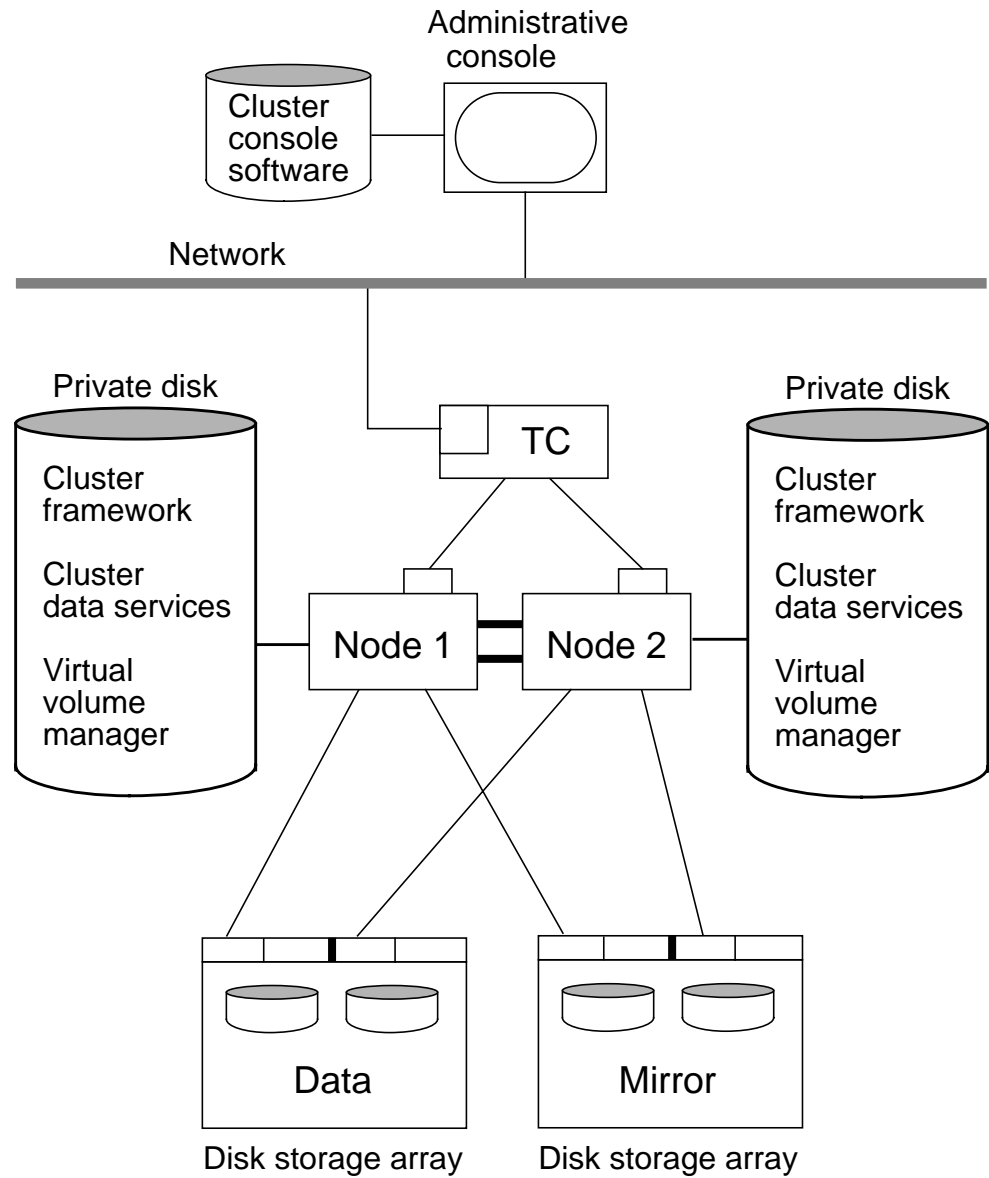


Figure 5-2 Cluster Software Distribution

Sun Cluster Framework Software

The Sun™ Cluster 3.0 07/01 CD-ROM contains the following framework software packages:

- SUNWccn – Sun Cluster Console
- SUNWmdm – Solstice DiskSuite support (Mediator software)
- SUNWscdev – Sun Cluster Developer support
- SUNWscfab – Standard Generalized Markup Language (SGML) documentation
- SUNWscman – Sun Cluster manual pages
- SUNWscr – Sun Cluster Framework software
- SUNWscsal – Sun Cluster SyMON agent library
- SUNWscsam – Sun Cluster SyMON modules
- SUNWscscn – Sun Cluster SyMON console add-on
- SUNWscshl – Sun Cluster SyMON help add-on
- SUNWscssv – Sun Cluster SyMON server add-on
- SUNWscu – Sun Cluster, (Usr)
- SUNWscva – Apache SSL components
- SUNWscvm – Sun Cluster VxVM support
- SUNWscvr – SunPlex Manager root components
- SUNWscvw – SunPlex Manager core components

Some of the packages are not part of the general cluster framework software and are installed at different times.

Sun™ Cluster 3.0 07/01 Agents

The following data service support packages are available on the Sun™ Cluster 3.0 07/01 Agents CD-ROM:

- SunCluster_Data_Service_Answer_Book
- SunCluster_HA_Apache
- SunCluster_HA_DNS
- SunCluster_HA_NFS
- SunCluster_HA_Netscape_LDAP
- SunCluster_HA_Oracle
- SunCluster_HA_SAP
- SunCluster_HA_Sybase
- SunCluster_HA_iWS
- SunCluster_Oracle_Parallel_Server



Note – The Oracle Parallel Server support package supplies the Sun Cluster 2.2 version of the cluster membership monitor (CMM) that runs in Usermode. Support for Oracle distributed lock management (UDLM/IDLM) is also supplied.

Virtual Volume Management Software

The Solstice DiskSuite software is available on the *Solaris 8 Software 2 of 2* CD in the `sol_8_sparc_2/Solaris_8/EA/products` directory.

The VERITAS Volume Manager software must be purchased separately from the Sun Cluster software.

Sun™ Cluster 3.0 07/01 Licensing

No license keys are required for the Sun Cluster software. You must, however, furnish paper license numbers to obtain service.

VERITAS Volume Manager is automatically licensed when used with some Sun storage arrays, such as the Sun StorEdge A5x00 and Sun StorEdge T3 arrays. If you use only Sun StorEdge MultiPacks, you need a special license for VERITAS Volume Manager.

You need a special license to use the Sun Management Center software.

Sun Cluster Basic Installation Process

The basic Sun™ Cluster 3.0 07/01 host system (node) installation process is completed in several major steps. The general process is:

1. Repartition boot disks to meet Sun™ Cluster 3.0 07/01 requirements.
2. Install the Solaris Operating Environment software.
3. Configure the environment for the cluster host systems.
4. Install Solaris 8 Operating Environment patches.

Search for Sun Cluster 3.0 on SunSolve™ in the Early Notifier category.

5. Install hardware-related patches.

Use interactive PatchPro with an appropriate configuration, such as: Solaris 8 Operating Environment software, Sun StorEdge A5000 storage arrays, peripheral component interface (PCI) FC100 cards, ST19171FC disk drives (<http://patchpro.ebay/>)

6. Install Sun™ Cluster 3.0 07/01 software on the first cluster node.
7. Install Sun™ Cluster 3.0 07/01 on the remaining nodes.
8. Install any Sun Cluster patches.
9. Perform postinstallation checks and configuration.

Sun Cluster Alternative Installation Processes

There are three different methods for installing the Sun Cluster software. For instructional purposes, the lab exercises in this course use the manual method. You must have a good understanding of the basic installation process before attempting the more advanced methods.

Manual Installation

Use the Sun Cluster `scinstall` utility to manually install the Sun Cluster software on each cluster host system. The Solaris Operating Environment software must be previously installed.

JumpStart™ Installation

If you have an existing JumpStart™ server, you can use the `scinstall` utility to include the Sun Cluster installation in the existing JumpStart configuration. The Sun Cluster CD-ROM image is copied to the JumpStart server. You can then automatically install the Solaris Operating Environment and the Sun Cluster software in a single operation.

Note – You must answer all of the `scinstall` questions just as you do during a manual installation. The resulting information is stored and used during the JumpStart installation.



SunPlex Manager Installation

You can use the SunPlex manager application to install the Sun Cluster software. You must first install the Solaris Operating Environment and the SunPlex Manager software on each cluster node.

You can also use SunPlex Manager to install two of the basic data services, HA for NFS and HA for Apache. If you install either of the data services, you are also required to install and use Solstice DiskSuite software. There are strict preconfiguration requirements for each installation type.

Note – The SunPlex Manager software packages are automatically installed on each node during the Sun Cluster installation process.



Configuring the Sun Cluster Node Environment

You can configure the Sun Cluster environment before you install the Sun™ Cluster 3.0 07/01 software providing you have documented your intended cluster configuration.

You should configure the user `root` environment on each cluster node and also configure local network name resolution.

Configuring the User `root` Environment

The `root` login environment should include the following search path and man page information:

```
PATH=$PATH:/usr/cluster/bin:/etc/vx/bin:/opt/VRTSvmsa/bin
```

```
MANPATH=$MANPATH:/usr/cluster/man:/usr/share/man:/opt/VRTSvxvm/man:/opt/VRTSvmsa/man
```



Note – Some of the path information depends on which virtual volume management software your cluster uses.

Configuring Network Name Resolution

The names of cluster nodes should be resolved locally so that the cluster is not disabled in the event of a naming service failure. Following is a recommended configuration for the `/etc/nsswitch.conf` files. The example shows only partial contents.

```
passwd:      files nis
group:       files nis
...
hosts:       cluster files nis
...
netmasks:   cluster files nis
```



Note – Several changes to the `/etc/nsswitch.conf` file are mandatory and are performed automatically during the Sun Cluster software installation. You should verify the changes after completing the Sun Cluster software installation configuration.

Installing Sun Cluster Node Patches

The Sun Cluster nodes might require patches in the following areas:

- Solaris Operating Environment patches
- Storage array interface firmware patches
- Storage array disk drive firmware patches
- VERITAS Volume Manager patches
- Solstice DiskSuite patches

You cannot install some patches, such as those for VERITAS Volume Manager and Solstice DiskSuite, until after the volume management software installation is completed.

Patch Installation Warnings

Before installing any patches, always do the following:

- Make sure all cluster nodes have the same patch levels.
- Do not install any firmware-related patches without qualified assistance.
- Always obtain the most current patch information.
- Read all patch README notes carefully.

Obtaining Patch Information

You should always consult with your Sun field representative about possible patch installations. You can obtain current patch information as follows:

- Consult <http://sunsolve.sun.com>.
- Use the SunSolve PatchPro tool interactively

The internal PatchPro site is: <http://patchpro.ebay/>.

- Read current Sun Cluster release notes

Installing the Sun™ Cluster 3.0 07/01 Software

Although there are three different methods to install the Sun™ Cluster 3.0 07/01 software on the cluster nodes, only the interactive method is described in this module. You must understand the interactive installation process before you attempt a more advanced method.

Installing Sun Cluster Interactively

The Sun Cluster installation program, `scinstall`, is located on the Sun™ Cluster 3.0 07/01 CD in the `SunCluster_3_0/Tools` directory. When you start the program without any options, it prompts you for cluster configuration information that is stored for use later in the process.

Before starting the installation process, you must have the following information at hand:

- The cluster name.
- The names of all nodes that will be part of this cluster.
- The node authentication mode during installation (Data Encryption Standard [DES]).
- The cluster transport network address and netmask if you do not want to use the default address and netmask.

You cannot change the private network address after the cluster has successfully formed.

- The cluster transport adapter configuration.
- The global devices file-system name.
- Whether you want an automatic reboot after installation.



Note – When you finish answering the prompts, the `scinstall` command line equivalent generated from your input displays for confirmation. This information is stored in the cluster installation log file in the `/var/cluster/logs/install` directory.

The Initial `scinstall` Menu

Although you can install the Sun Cluster software on all nodes in parallel, you can complete the installation on the first node for practice and then run `scinstall` on all other nodes in parallel. The additional nodes get some basic configuration information from the first, or sponsoring node, that you configured.

As shown in the following example, you use option 1 when establishing the first node in a new cluster. Use option 2 on all other nodes.

```
# ./scinstall
*** Main Menu ***
```

```
    Please select from one of the following (*) options:
```

- * 1) Establish a new cluster using this machine as the first node
- * 2) Add this machine as a node in an established cluster
- 3) Configure a cluster to be JumpStarted from this install server
- 4) Add support for a new data service to this cluster node
- 5) Print release information for this cluster node

- * ?) Help with menu options
- * e) Exit

```
Option:  1
```

The balance of the initial node cluster software installation is excerpted in the following sections with comments for each section.

Note – You can type Control-D at any time to return to the main `scinstall` menu and restart the installation. Your previous answers to questions become the default answers.



Supplying Cluster and Node Names

Initially, the `scinstall` program requests the name of the cluster and the name of nodes that will be added. The name of the initial node is already known. An example of the dialog follows.

```
What is the name of the cluster you want to establish?  
planets
```

```
>>> Cluster Nodes <<<
```

```
This release of Sun Cluster supports a total of up  
to 8 nodes. Please list the names of the other nodes  
planned for the initial cluster configuration. List one  
node name per line. When finished, type Control-D:
```

```
Node name: mars  
Node name (Ctrl-D to finish): ^D
```

```
This is the complete list of nodes:
```

```
venus  
mars
```

```
Is it correct (yes/no) [yes]? yes
```



Note – Use local `/etc/hosts` name resolution to prevent installation difficulties and to increase cluster availability by eliminating the cluster's dependence on a naming service.

Selecting Data Encryption Standard Authentication

As shown in the following example, you can select Data Encryption Standard (DES) authentication for use only during the remainder of the cluster installation. This can prevent unauthorized modification of the CCR contents during installation. This is not very likely but might be a requirement in high security environments.

```
Do you need to use DES authentication (yes/no) [no]? no
```



Note – If you must use DES authentication for security reasons, you must completely configure DES authentication on all nodes before starting the cluster installation. DES authentication is used only during the node installation process.

Configuring the Cluster Interconnect

The following examples summarize the process of configuring the cluster interconnect. You must approve the default network address (172.16.0.0) and netmask (255.255.0.0). If the default base address is in use elsewhere at your site, you have to supply a different address and or netmask, or both.

If your cluster is a two-node cluster, you are asked if switches are used. The connections can be point-to-point in a two-node cluster, but you can also use switches. If the cluster is greater than two nodes, switches are assumed. The switches are assigned arbitrary names.

You must also furnish the names of the cluster transport adapters.

```
Is it okay to accept the default network address (yes/no) [yes]? yes
```

```
Is it okay to accept the default netmask (yes/no) [yes]? yes
```

```
Does this two-node cluster use transport junctions (yes/no) [yes]? yes
```

```
What is the name of the first junction in the cluster [switch1]? sw1
```

```
What is the name of the second junction in the cluster [switch2]? sw2
```

```
What is the name of the first cluster transport adapter ? hme1
```

```
Name of the junction to which "hme1" is connected [sw1]? sw1
```

```
Okay to use the default for the "hme1" connection [yes]? yes
```

```
What is the name of the second cluster transport adapter [hme2]? hme2
```

```
Name of the junction to which "hme2" is connected [sw2]? sw2
```

```
Use the default port for the "hme2" connection [yes]? yes
```



Caution – If you do not specify the correct interfaces when installing the first cluster node, the installation completes without errors. When you install the Sun Cluster software on the second node, it is not able to join the cluster because of the incorrect configuration data. You have to manually correct the problem on the first node before continuing.

Configuring the Global Devices File System

Normally, a `/globaldevices` file system already exists on the boot disk of each cluster node. The `scinstall` utility asks if you want to use the default global file system. You can also use a different file system or have the `scinstall` utility create one for you.

The first example is for an existing `/globaldevices` file system.

The default is to use `/globaldevices`.

Is it okay to use this default (yes/no) [yes]? **yes**

The second example shows how you can use a file system other than the default `/globaldevices`.

Is it okay to use this default (yes/no) [yes]? **no**

Do you want to use an already existing file system (yes/no) [yes]? **yes**

What is the name of the file system **/sparefs**



Warning – If you use a file system other than the default, it must be an empty file system containing only the `lost+found` directory.

The third example shows how to create a new `/globaldevices` file system using an available disk partition.

Is it okay to use this default (yes/no) [yes]? **no**

Do you want to use an already existing file system (yes/no) [yes]? **no**

What is the name of the disk partition you want to use **/dev/dsk/c0t0d0s4**



Note – You should use the default `/globaldevices` file system. Standardization helps during problem isolation.

Selecting Automatic Reboot

To complete the basic installation of the first node, you must decide whether you want the system to reboot automatically.

```
Do you want scinstall to re-boot for you (yes/no) [yes]?
yes
```

The reboot question should be considered because you might need to install Sun Cluster patches before rebooting.



Note – If for any reason the reboot fails, you can use a special boot option (`ok boot -x`) to disable the cluster software startup until you can fix the problem.

Confirming the Final Configuration

Before starting the Sun Cluster installation, the `scinstall` utility displays the command line equivalent of the installation for your approval. This information is recorded in the installation log file that can be examined in the `/var/cluster/logs/install` directory.

A typical display is as follows:

Your responses indicate the following options to `scinstall`:

```
scinstall -i \
  -C planets \
  -N venus \
  -T node=venus,node=mars,authtype=sys \
  -A trtype=dlpi,name=hme1 -A trtype=dlpi,name=hme2 \
  -B type=switch,name=hub1 -B type=switch,name=hub2 \
  -m endpoint=:hme1,endpoint=hub1 \
  -m endpoint=:hme2,endpoint=hub2
```

```
Are these the options you want to use [yes]? yes
```

```
Do you want to continue with the install (yes/no) [yes]? yes
```

Installation Operations

During the final installation process, the `scinstall` utility performs the following operations on the first cluster node:

- Installs cluster software packages
- Disables routing on the node (`touch /etc/notrouter`)
- Creates an installation log (`/var/cluster/logs/install`)
- Reboots the node
- Creates the Disk ID (DID) devices during the reboot



Note – It is normal to see some DID error messages during the reboot. You should not see any such messages during later system reboots. A typical error message is `did_instances, no such file`.

Installing Additional Nodes

After you complete the Sun Cluster software installation on the first node, you can run `scinstall` on the remaining cluster nodes. The node IDs are assigned in the order in which the nodes are installed. During installation, the cluster is placed in *install mode*, which gives a single quorum vote to Node 1.

As the installation on each new node is complete, each node reboots and joins the cluster without a quorum vote. If you reboot the first node at this point, all the other nodes would panic because they cannot obtain a quorum. However, you can reboot the second or later nodes. They should come up and join the cluster without errors.

Cluster nodes remain in install mode until you use the `scsetup` command to reset the install mode.

You must perform postinstallation configuration to take the nodes out of install mode and also to establish quorum disks.



Note – If you must reboot the first node before performing postinstallation procedures, you can first shut down the entire cluster using the `scshutdown` command. You can also shut down single nodes using the standard `init 0` command.

Postinstallation Configuration

Postinstallation can include a number of complex tasks, such as installing a volume manager or database software or both. There are less complex tasks that you must complete first.

This section focuses on the following postinstallation tasks:

- Resetting the install mode of operation
- Configuring the Network Time Protocol

Resetting Install Mode of Operation

Before a new cluster can operate normally, you must reset the install mode attribute on all nodes. You must also define a quorum device at the same time. You can do this automatically using the `scsetup` utility or manually with the `scconf` command.

Resetting Install Mode Using the `scsetup` Utility

The `scsetup` utility is a menu-driven interface that prompts for quorum device information the first time it is run on a new cluster installation. When the quorum device is defined, the install mode attribute is reset for all nodes.



Note – You must provide the path to the quorum disks in a DID device format. You must first identify the appropriate DID devices with the `scdidadm` command.

Most of the informational text has been omitted in the following `scsetup` example for clarity.

```
# /usr/cluster/bin/scsetup
>>> Initial Cluster Setup <<<
```

```

This program has detected that the cluster
"installmode" attribute is set ...
```

```

Please do not proceed if any additional nodes have yet
to join the cluster.
```

```
Is it okay to continue (yes/no) [yes]? yes
```

Which global device do you want to use (d<N>)? **d2**

Is it okay to proceed with the update (yes/no) [yes]? **yes**

```
scconf -a -q globaldev=d2
```

Do you want to add another quorum disk (yes/no)? **no**

Is it okay to reset "installmode" (yes/no) [yes]? **yes**

```
scconf -c -q reset
```

Cluster initialization is complete.



Note – Although it appears that the `scsetup` utility uses two simple `scconf` commands to define the quorum device and reset install mode, the process is more complex. The `scsetup` utility perform numerous verification checks for you. It is recommended that you *do not* use `scconf` manually to perform these functions.

Use the `scconf -p` command to verify the install mode status.

Configuring Network Time Protocol

You must modify the Network Time Protocol (NTP) configuration file, `/etc/inet/ntp.conf`, on all cluster nodes. You must remove all private host name entries that are not being used by the cluster. Also, if you changed the private host names of the cluster nodes, update this file accordingly.

You can also make other modifications to meet your NTP requirements.

You can verify the current `ntp.conf` file configuration as follows:

```
# more /etc/inet/ntp.conf |grep clusternode
peer clusternode1-priv prefer
peer clusternode2-priv
peer clusternode3-priv
peer clusternode4-priv
peer clusternode5-priv
peer clusternode6-priv
peer clusternode7-priv
peer clusternode8-priv
```

If your cluster will ultimately have two nodes, remove the entries for nodes 3, 4, 5, 6, 7, and 8 from the `ntp.conf` file on all nodes.

Until the `ntp.conf` file configuration is corrected, you see boot error messages similar to the following:

```
May  2 17:55:28 pnode1 xntpd[338]: couldn't resolve
'clusternode3-priv', giving up on it
```

Postinstallation Verification

When you have completed the Sun Cluster software installation on all nodes, verify the following information:

- DID device configuration
- General cluster status
- Cluster configuration information

Verifying DID Devices

Each attached system sees the same DID devices but might use a different logical path to access them. You can verify the DID device configuration with the `scdidadm` command. The following `scdidadm` output demonstrates how a DID device can have a different logical path from each connected node.

```
# scdidadm -L
1      devsys1:/dev/rdisk/c0t0d0      /dev/did/rdisk/d1
2      devsys1:/dev/rdisk/c2t37d0     /dev/did/rdisk/d2
2      devsys2:/dev/rdisk/c3t37d0     /dev/did/rdisk/d2
3      devsys1:/dev/rdisk/c2t33d0     /dev/did/rdisk/d3
3      devsys2:/dev/rdisk/c3t33d0     /dev/did/rdisk/d3
4      devsys1:/dev/rdisk/c2t52d0     /dev/did/rdisk/d4
4      devsys2:/dev/rdisk/c3t52d0     /dev/did/rdisk/d4
5      devsys1:/dev/rdisk/c2t50d0     /dev/did/rdisk/d5
5      devsys2:/dev/rdisk/c3t50d0     /dev/did/rdisk/d5
6      devsys1:/dev/rdisk/c2t35d0     /dev/did/rdisk/d6
6      devsys2:/dev/rdisk/c3t35d0     /dev/did/rdisk/d6
7      devsys1:/dev/rdisk/c3t20d0     /dev/did/rdisk/d7
7      devsys2:/dev/rdisk/c2t20d0     /dev/did/rdisk/d7
8      devsys1:/dev/rdisk/c3t18d0     /dev/did/rdisk/d8
8      devsys2:/dev/rdisk/c2t18d0     /dev/did/rdisk/d8
9      devsys1:/dev/rdisk/c3t1d0      /dev/did/rdisk/d9
9      devsys2:/dev/rdisk/c2t1d0      /dev/did/rdisk/d9
10     devsys1:/dev/rdisk/c3t3d0       /dev/did/rdisk/d10
10     devsys2:/dev/rdisk/c2t3d0       /dev/did/rdisk/d10
11     devsys2:/dev/rdisk/c0t0d0       /dev/did/rdisk/d11
```

Note – Devices `d1` and `d11` are the local boot disks for each node.



Verifying General Cluster Status

The `scstat` utility displays the current status of various cluster components. You can use it to display the following information:

- The cluster name and node names
- Names and status of cluster members
- Status of resource groups and related resources
- Cluster interconnect status

The following `scstat-q` command option displays the cluster membership and quorum vote information.

```
# /usr/cluster/bin/scstat -q
Quorum
  Current Votes:                3
  Votes Configured:             3
  Votes Needed:                 2
    Node Quorum
      Node Name:                venus
        Votes Configured:       1
        Votes Contributed:      1
        Status:                 Online

      Node Name:                mars
        Votes Configured:       1
        Votes Contributed:      1
        Status:                 Online

    Device Quorum
      Quorum Device Name:       /dev/did/rdisk/d2s2
      Votes Configured:         1
      Votes Contributed:        1
      Nodes Having Access:
        venus                   Enabled
        mars                   Enabled
      Owner Node:               venus
      Status:                   Online
```

Verifying Cluster Configuration Information

Cluster configuration information is stored in the CCR on each node. You should verify that the basic CCR values are correct. The `scconf -p` command displays general cluster information along with detailed information about each node in the cluster.

The following `scconf` output is for the first node added to a new two-node cluster.

```
# scconf -p
Cluster name:                                codev
Cluster ID:                                  0x3A297CD9
Cluster install mode:                        enabled
Cluster private net:                         172.16.0.0
Cluster private netmask:                     255.255.0.0
Cluster new node authentication:              unix
Cluster new node list:                       pnode1 pnode2
Cluster nodes:                               pnode1

Cluster node name:                           pnode1
  Node ID:                                   1
  Node enabled:                              yes
  Node private hostname:                     clusternode1-priv
  Node quorum vote count:                    1
  Node reservation key:                      0x3A297CD900000001
  Node transport adapters:                   hme1 hme2

Node transport adapter:                       hme1
  Adapter enabled:                           no
  Adapter transport type:                     dlpi
  Adapter property:                           device_name=hme
  Adapter property:                           device_instance=1
  Adapter property:                           dlpi_heartbeat_timeout=10000
  Adapter property:                           1000
  Adapter property:                           nw_bandwidth=80
  Adapter property:                           bandwidth=10
  Adapter port names:                         0

  Adapter port:                               0
    Port enabled:                             no

Node transport adapter:                       hme2
  Adapter enabled:                           no
  Adapter transport type:                     dlpi
  Adapter property:                           device_name=hme
```

```

Adapter property:          device_instance=2
Adapter property:          10000
Adapter property:          1000
Adapter property:          nw_bandwidth=80
Adapter property:          bandwidth=10
Adapter port names:        0

Adapter port:              0
  Port enabled:            no

Cluster transport junctions:  switch1 switch2

Cluster transport junction:  switch1
  Junction enabled:         no
  Junction type:            switch
  Junction port names:      1

  Junction port:            1
    Port enabled:          no

Cluster transport junction:  switch2
  Junction enabled:         no
  Junction type:            switch
  Junction port names:      1

  Junction port:            1
    Port enabled:          no

Cluster transport cables

                                Endpoint      Endpoint      State
                                -----      -
Transport cable:  pnode1:hme1@0 switch1@1    Disabled
Transport cable:  pnode1:hme2@0 switch2@1    Disabled

Quorum devices:          <NULL>

```

Correcting Minor Configuration Errors

When you install the Sun Cluster software, some common mistakes are:

- Using the wrong cluster name
- Using incorrect cluster interconnect interface assignments

You can resolve these mistakes using the `scsetup` utility.

You can run the `scsetup` utility on any cluster node where it does the following tasks:

- Adds or removes quorum disks
- Adds, removes, enables, or disables cluster transport components
- Registers or unregisters VERITAS disk groups
- Adds or removes node access from a VERITAS disk group
- Changes the cluster private host names
- Prevents or permits the addition of new nodes
- Changes the name of the cluster



Note – You should use the `scsetup` utility instead of manual commands, such as `scconf`. The `scsetup` utility is less prone to errors and, in some cases, performs complex verification before proceeding.

Exercise: Installing the Sun Cluster Server Software

In this exercise, you complete the following tasks:

- Configure environment variables
- Install the Sun Cluster server software
- Perform post-installation configuration

Preparation

Obtain the following information from your instructor if you have not already done so in a previous exercise:

1. Ask your instructor about the location of the Sun™ Cluster 3.0 07/01 software. Record the location.

Software location: _____



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Verifying the Boot Disk

Perform the following steps on all nodes to verify that the boot disks have a 100-Mbyte `/globaldevices` partition on slice 4 and a small partition on slice 7 for use by Solstice DiskSuite replicas.

1. Type the `mount` command, and record the logical path to the boot disk on each node (typically `/dev/dsk/c0t0d0`).

Node 1 boot device: _____

Node 2 boot device: _____

2. Type the `prtvtoc` command to verify each boot disk meets the Sun Cluster partitioning requirements.

```
# /usr/sbin/prtvtoc /dev/dsk/c0t0d0s2
```



Note – Append slice 2 (`s2`) to the device path. The sector count for the `/globaldevices` partition should be at least 200,000.

Task – Verifying the Environment

Perform the following steps on both nodes:

1. Verify that the `.profile` file on each cluster node contains the following environment variables:

```
PATH=$PATH:/usr/cluster/bin:/etc/vx/bin:/opt/VRTSvmsa/bin
```

```
MANPATH=$MANPATH:/usr/cluster/man:/usr/share/man:/opt/VRTSvxvm/man:/opt/VRTSvmsa/man
```

```
TERM=dtterm
```

```
EDITOR=vi
```

```
export PATH MANPATH TERM EDITOR
```

Note – If necessary, create the `.profile` login file as follows:

```
cp /etc/skel/local.profile ~/.profile.
```



2. If you edit the file, verify the changes are correct by logging out and in again as user `root`.
3. On both nodes, create a `.rhosts` file in the root directory. Edit the file, and add a single line with a plus (+) sign.
4. On both cluster nodes, edit the `/etc/default/login` file and comment out the `CONSOLE=/dev/console` line.

Note – The `.rhosts` and `/etc/default/login` file modifications shown here can be a security risk in some environments. They are used here to simplify some of the lab exercises.



Task – Updating the Name Service

Perform the following steps to update the name service:

1. Edit the `/etc/hosts` file on the administrative workstation and all cluster nodes, and add the Internet Protocol (IP) addresses and host names of the administrative workstation and cluster nodes.

2. If you are using a naming service, add the IP addresses and host names to the name service.



Note – Your lab environment might already have all of the IP addresses and host names entered in the `/etc/hosts` file.

Task – Establishing a New Cluster

Perform the following steps to establish the first node in your cluster:

1. In the `cconsole` window, log in to Node 1 as user `root`.
2. Change to the location of the Sun™ Cluster 3.0 07/01 software furnished by your instructor.
3. Change to the `SunCluster_3.0/Tools` directory.
4. Start the `scinstall` script on Node 1 only.
5. As the installation proceeds, make the following choices:
 - a. Select option 1, Establish a new cluster.
 - b. Furnish your assigned cluster name.
 - c. Furnish the name of the second node that will be added later.
 - d. Verify the list of node names.
 - e. Reply **no** to using DES authentication.
 - f. Unless your instructor has stated otherwise, accept the default cluster transport base address and netmask values.
 - g. Configure the cluster transport based on your cluster configuration. Accept the default names for switches if your configuration uses them.
 - h. Accept the default global device file system.
 - i. Reply **yes** to the automatic reboot question.
 - j. Examine the `scinstall` command options for correctness. Accept them if they seem appropriate. The options should look similar to the following:

```
scinstall -i
-C devcluster
-F
```

```
-T node=pnode1,node=pnode2,authtype=sys
-A trtype=dlpi,name=hme1 -A trtype=dlpi,name=hme2
-B type=switch,name=switch1 -B type=switch,name=switch2
-m endpoint=:hme1,endpoint=switch1
-m endpoint=:hme2,endpoint=switch2
```



Note – The `scinstall -F` option is for the global devices file system. This entry should be blank if you accepted the default (`/globaldevices`). The command line output is copied into the cluster installation log file in `/var/cluster` directory.

6. Observe the following messages during the node reboot:

```
/usr/cluster/bin/scdidadm: Could not load DID instance
list.
Cannot open /etc/cluster/ccr/did_instances.
```

Booting as part of a cluster

```
NOTICE: CMM: Node pnode1 (nodeid = 1) with votecount =
1 added.
NOTICE: CMM: Node pnode1: attempting to join cluster.
NOTICE: CMM: Cluster has reached quorum.
NOTICE: CMM: Node pnode1 (nodeid = 1) is up; new
incarnation number = 975797536.
NOTICE: CMM: Cluster members: pnode1 .
NOTICE: CMM: node reconfiguration #1 completed.
NOTICE: CMM: Node pnode1: joined cluster.
```

Configuring DID devices

```
Configuring the /dev/global directory (global devices)
May  2 17:55:28 pnode1 xntpd[338]: couldn't resolve
'clusternode2-priv', giving up on it
```

```
May  2 17:57:31 pnode1 Cluster.PMF.pmf: Error opening
procfs control file </proc/492/ctl> for tag <scsymon>:
No such file or directory
The
```

Task – Adding a Second Cluster Node

Perform the following steps to complete the creation of a two-node cluster.

1. In the `cconsole` window, log in to Node 2 as user `root`.
2. Change to the location of the Sun™ Cluster 3.0 07/01 software furnished by your instructor.
3. Change to the `SunCluster_3.0/Tools` directory.
4. Start the `scinstall` script on Node 2 only.
5. As the installation proceeds, make the following choices:
 - a. Select option 2, Add this machine as a node in an established cluster.
 - b. Provide the name of a sponsoring node.
 - c. Provide the name of the cluster you want to join.

Type `scconf -p | more` on the first node (the sponsoring node) if you have forgotten the name of the cluster.

- d. Answer the node configuration question.
- e. Answer the cluster interconnect questions as required.
- f. Select the default for the global device directory.
- g. Reply **yes** to the automatic reboot question.
- h. Examine and approve the `scinstall` command line options. It should look similar to the following:

```
scinstall -i
-C scdev
-N pnode1
-A trtype=dlpi,name=hme1 -A trtype=dlpi,name=hme2
-m endpoint=:hme1,endpoint=switch1
-m endpoint=:hme2,endpoint=switch2
```

Note – You see interconnect-related errors on Node 1 until Node 2 completes the first portion of its reboot operation.



Task – Configuring a Quorum Device

Perform the following steps to finish initializing your new two-node cluster.

1. On either node, type the **scstat -q** command.

The second node is still in install mode and has no votes. Only the first node has a vote. No quorum device has been assigned.

2. On Node 1, type the **scdidadm -L** command, and record the DID device you intend to configure as a quorum disk.

Quorum disk: _____ (d4, d6, etc.)



Caution – Pay careful attention. The first few DID devices might be local disks, such as the boot disk and a CD-ROM (target 6). Examine the standard logical path to make sure the DID device you select is a disk in a storage array and is connected to both nodes.

3. On Node 1, type the **scsetup** command, and supply the name of the DID device (global device) you selected in the previous step. You should see output similar to the following.

```
scconf -a -q globaldev=d12
May  3 22:29:13 pnode1 cl_runtime: NOTICE: CMM: Cluster
members: pnode1 pnode2 .
May  3 22:29:13 pnode1 cl_runtime: NOTICE: CMM: node
reconfiguration #4 completed.
```

4. Do *not* add a second quorum disk.
5. Reply **yes** to the reset installmode question.

You should see a “Cluster initialization is complete” message.

Now that install mode has been reset and a quorum device defined, the **scsetup** utility displays its normal menu selections.

6. Type **q** to quit the **scsetup** utility.

Task – Configuring the Network Time Protocol

Perform the following steps on both nodes to complete the NTP configuration:

1. On both nodes, edit the `/etc/inet/ntp.conf` file and remove configuration entries for node instances that are not configured. In a two-node cluster, you should remove the following lines:

```
peer clusternode3-priv
peer clusternode4-priv
peer clusternode5-priv
peer clusternode6-priv
peer clusternode7-priv
peer clusternode8-priv
```

2. On both nodes, type the `scstat -q` command.

You should see three quorum votes present and a quorum device.

3. On both nodes, type the `scdidadm -L` command.

Each shared (dual-ported) DID device should show a logical path from each cluster node.

4. On either node, type the `scconf -p` command.

The cluster status, node names, transport configuration, and quorum device information should be complete.

Task – Configuring Host Name Resolution

Perform the following step on both nodes to ensure local host name resolution:

1. On both nodes edit the `/etc/nsswitch.conf` file, and make sure local files are consulted before a naming service when trying to resolve host names. The following line is a correct entry:

```
hosts:      cluster files nis [NOTFOUND=return]
```

Testing Basic Cluster Operation

Perform the following steps to verify the basic cluster software operation:



Note – You are using commands that have not yet been presented in the course. If you have any questions, consult with your instructor.

1. Log in to each of your cluster host systems as user `root`.
2. On Node 1, shut down all cluster nodes.

```
# scshutdown -y -g 15
```



Note – The `scshutdown` command completely shuts down all cluster nodes, including the Solaris Operating Environment.

3. Boot Node 1; it should come up and join the cluster.
4. Boot Node 2; it should come up and join the cluster.

Exercise Summary

Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Install the Sun Cluster host system software
- ☐ Correctly interpret configuration questions during the Sun Cluster software installation
- ☐ Perform postinstallation configuration

Think Beyond

How can you add a new node to an existing cluster?

What could happen if you did not configure any quorum devices?

Basic Cluster Administration

Objectives

Upon completion of this module, you should be able to:

- Perform basic cluster startup and shutdown operations
- Boot nodes in non-cluster mode
- Place nodes in a maintenance state
- Verify cluster status from the command line
- Recover from a cluster amnesia error

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- What must be monitored in the Sun Cluster environment?
- How current does the information need to be?
- How detailed does the information need to be?
- What types of information are available?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Cluster Status Commands

There are several cluster status commands. Some of the commands have uses other than status reporting.

Checking Status Using the `scstat` Command

Without any options, the `scstat` command displays general information for all cluster nodes. You can use options to restrict the status information to a particular type of information, to a particular node or to both.

The following command displays the cluster transport status for a single node.

```
# scstat -W -h pnode2
```

```
-- Cluster Transport Paths --
```

	Endpoint -----	Endpoint -----	Status -----
Transport path:	pnode1:hme2	pnode2:hme2	Path online
Transport path:	pnode1:hme1	pnode2:hme1	Path online

Checking Status Using the `sccheck` Command

The `sccheck` command verifies that all of the basic global device structure is correct on all nodes. Run the `sccheck` command after installing and configuring a cluster, as well as after performing any administration procedures that might result in changes to the devices, volume manager, or Sun Cluster configuration.

You can run the command without options or direct it to a single node. You can run it from any active cluster member. There is no output from the command unless errors are encountered. Typical `sccheck` command variations follow.

```
# sccheck  
# sccheck -h pnode2
```


Checking Status Using the `scinstall` Utility

During the Sun Cluster software installation, the `scinstall` utility is copied into the `/usr/cluster/bin` directory. You can run the `scinstall` utility with options that display the Sun Cluster revision, the names and revision of installed packages, or both. The displayed information is for the local node only. A typical `scinstall` status output follows.

```
# scinstall -pv
SunCluster 3.0
SUNWscr:      3.0.0,REV=2000.10.01.01.00
SUNWscdev:    3.0.0,REV=2000.10.01.01.00
SUNWscu:      3.0.0,REV=2000.10.01.01.00
SUNWscman:    3.0.0,REV=2000.10.01.01.00
SUNWscsal:    3.0.0,REV=2000.10.01.01.00
SUNWscsam:    3.0.0,REV=2000.10.01.01.00
SUNWscvm:     3.0.0,REV=2000.10.01.01.00
SUNWmdm:      4.2.1,REV=2000.08.08.10.01
#
```



Caution – Use the `scinstall` utility carefully. It is possible to create serious cluster configuration errors using the `scinstall` utility.

Cluster Control

Basic cluster control includes starting and stopping clustered operation on one or more nodes and booting nodes in non-cluster mode.

Starting and Stopping Cluster Nodes

The Sun Cluster software starts automatically during a system boot operation. Use the `init` command to shut down a single node. You use the `scshutdown` command to shut down all nodes in the cluster.

Before shutting down a node, you should switch resource groups to the next preferred node and then run `init 0` on the node.

Note – After an initial Sun Cluster installation, there are no configured resource groups with which to be concerned.



Shutting Down a Cluster

You can shut down the entire cluster with the `scshutdown` command from any active cluster node. A typical cluster shutdown example follows.

```
# scshutdown -y -g 30
Broadcast Message from root (???) on pnode1 Wed May 20
17:43:17...
  The cluster scdev will be shutdown in 30 seconds
May 20 17:43:38 pnode1 cl_runtime: WARNING: CMM: Monitoring
disabled.
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
/etc/rc0.d/K05initrgm: Calling scswitch -S (evacuate)
Print services stopped.
May 20 17:43:54 pnode1 syslogd: going down on signal 15
The system is down.
syncing file systems... done
Program terminated
ok
```

Note – Similar messages appear on all active cluster nodes.



Booting Nodes in Non-Cluster Mode

Occasionally, you might want to boot a node without it joining in a clustered operation. A common reason would be installing software patches. Some patches cannot be installed on an active cluster member. An example follows.

```
ok boot -x
Rebooting with command: boot -x
Boot device:/pci@1f,4000/scsi@3/disk@0,0 File and args: -x
SunOS Release 5.8 Version Generic_108528-07 64-bit
Copyright 1983-2001 Sun Microsystems, Inc.All rights
reserved.
configuring IPv4 interfaces: hme0.
Hostname: pnode2
Not booting as part of a cluster
The system is coming up. Please wait.
Starting IPv4 routing daemon.
starting rpc services: rpcbind done.
Setting netmask of hme0 to 255.255.255.0
Setting default IPv4 interface for multicast: add net
224.0/4: gateway pnode2
syslog service starting.
Print services started.
May 20 17:51:02 pnode2 xntpd[195]: couldn't resolve
'clusternode1-priv', giving up on it
May 20 17:51:02 pnode2 xntpd[195]: couldn't resolve
'clusternode2-priv', giving up on it
volume management starting.
The system is ready.

pnode2 console login:
```

Other active cluster nodes display transport-related errors because they try to establish a transport path to the node running in non-cluster mode. An example of typical errors on active nodes follows:

```
# May 4 21:09:42 pnode1 cl_runtime: WARNING: Path
pnode1:hme1 - pnode2:hme1 initiation encountered errors,
errno = 62. Remote node may be down or unreachable through
this path.
May 4 21:09:42 pnode1 cl_runtime: WARNING: Path pnode1:hme2
- pnode2:hme2 initiation encountered errors, errno = 62.
Remote node may be down or unreachable through this path.
```

Placing Nodes in Maintenance Mode

If you anticipate a node will be down for an extended period, you can place the node in a maintenance state from an active cluster node. The maintenance state disables the node's quorum vote. You cannot place an active cluster member in a maintenance state. A typical command follows.

```
# scconf -c -q node=pnode2,maintstate
```

The `scstat` command shows that the possible vote for `pnode2` is now set to 0.

You can reset the maintenance state for a node either by rebooting the node or with the `scconf -c -q reset` command string.

Cluster Amnesia

In a two-node cluster, special measures are required to protect the integrity of the cluster configuration repository (CCR) information.

When the first node joins a two-node cluster, the following steps are taken to protect the integrity of the CCR:

- The quorum disk is reserved with a Small Computer System Interface-2 (SCSI-2) reservation
- Persistent reservation information is written to a private area of the quorum disk. In the Sun Cluster application, this function is named persistent group reservation emulation (PGRE).

The PGRE information includes the unique incarnation number and the names of all nodes that were part of that incarnation. The PGRE reservation persists across system boot operations while the SCSI-2 disk reservation is released when the node is shut down.

The PGRE information ensures that the CCR configuration information cannot be modified without at least one node present from the last incarnation. The general rule is that the last node to leave the cluster must be the first node to join the cluster.

If you boot a node, shut it down, and then try to boot the other node, the second node cannot complete the cluster software startup sequence until the first node joins the cluster.

The second node displays the following messages:

```
NOTICE: CMM: Quorum device 1 (gdevname /dev/did/rdisk/d6s2)
can not be acquired by the current cluster members. This
quorum device is held by node 1.
NOTICE: CMM: Cluster doesn't have operational quorum yet;
waiting for quorum.
```

Normally, you resolve cluster amnesia by booting the other cluster node. When the first node joins the cluster, the second node automatically completes its cluster software startup sequence.

Manually clearing the PGRE reservations is a complex process that must be performed by trained Sun support personnel.

Monitoring With the Sun Management Center

The Sun Management Center Sun MC software is a central monitoring utility that gathers and displays status information about configured client systems. The Sun MC software has three functional sections, which are:

- Server software
- Console software
- Agent software

Figure 6-1 illustrates the Sun MC software relationship in a Sun Cluster environment.

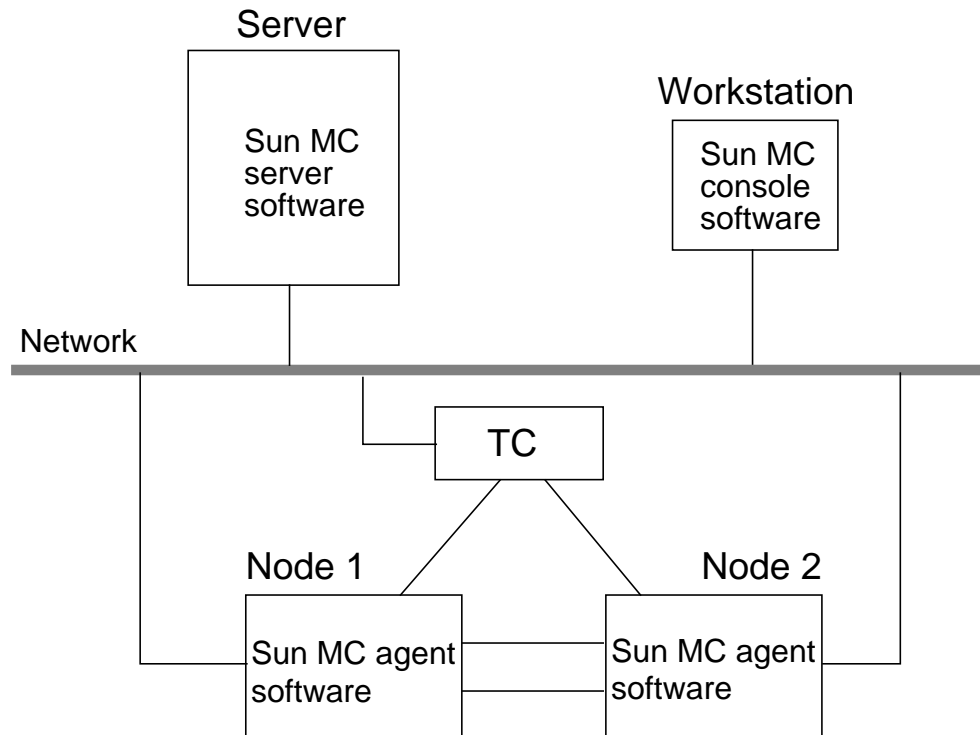


Figure 6-1 Sun Management Center Software Distribution

The cluster Sun MC agent packages, `SUNWscsal` and `SUNWscsam`, are automatically installed on the nodes during the initial Sun Cluster installation.

You can install the Sun MC server and console software on the same system as long as it has at least 128 Mbytes of memory and 50 Mbytes of available disk space in the `/opt` directory. You can choose a different directory other than `/opt`. The Sun MC software is licensed.

Sun MC Server Software

The server software usually resides on a centralized system. The server software is the majority of the Sun MC software.

Server Software Function

The function of the Sun MC server software is to accept user requests from the console software and pass the requests to the appropriate agent software. When the agent response is relayed back to the console interface, the server software interprets the response and forwards an appropriate graphical display.

Sun MC Console Software

The Sun MC console software is the user interface. You can configure the console software on several different workstations for different users.

Sun MC Agent Software

The agent software is installed on each Sun Cluster node. Agent software is unique to the installation. Sun Cluster agents gather Simple Network Management Protocol (SNMP) status information on command and transfer it to the Sun MC server.

The Sun Cluster-supplied modules for Sun MC enable you to graphically display cluster resources, resource types, and resource groups. They also enable you to monitor configuration changes and check the status of cluster components.



Note – The Sun Cluster Sun MC agent packages are part of the Sun Cluster software distribution. The Sun MC server and console software is a separately licensed product.

Exercise: Performing Basic Cluster Administration

In this exercise, you complete the following tasks:

- Perform basic cluster startup and shutdown operations
- Boot nodes in non-cluster mode
- Place nodes in a maintenance state
- Verify cluster status from the command line

Preparation

Join all nodes in the cluster and run the `cconsole` tool on the administration workstation.



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Verifying Basic Cluster Status

Perform the following steps to verify the basic status of your cluster:

1. Use the `scstat` command to verify the current cluster membership.

```
# scstat -q
```

2. Record the quorum configuration from the previous step.

Quorum votes possible: _____

Quorum votes needed: _____

Quorum votes present: _____

3. Verify that the global device structures on all nodes are correct.

```
# sccheck
```

4. Verify the revision of the currently installed Sun Cluster software on each cluster node.

```
# scinstall -pv
```

Task – Starting and Stopping Cluster Nodes

Perform the following steps to start and stop configured cluster nodes:

1. Verify that both nodes are active cluster members.
2. Shut down Node 2.

```
# init 0
```

Note – You might see remote procedure call-related (RPC-related) errors because the NFS server daemons, `nfsd` and `mountd`, are not running. This is normal if you are not sharing file systems in `/etc/dfs/dfstab`.

3. Join Node 2 into the cluster again by performing a boot operation.
4. When both nodes are members of the cluster, run `scshutdown` on one node to shut down the entire cluster.

```
# scshutdown -y -g 60 Log off now!!
```

5. Join Node 1 into the cluster by performing a boot operation.
6. When Node 1 is in clustered operation again, verify the cluster quorum configuration again.



```
# scstat -q | grep "Quorum votes"
Quorum votes possible:      3
Quorum votes needed:        2
Quorum votes present:       2
```

7. Leave Node 2 down for now.

Task – Placing a Node in Maintenance State

Perform the following steps to place a node in the maintenance state:

1. On Node 1, use the `scconf` command to place Node 2 into a maintenance state.

```
# scconf -c -q node=node2,maintstate
```

Note – Substitute the name of your node.

2. Verify the cluster quorum configuration again.

```
# scstat -q | grep "Quorum votes"
```

Note – The number of *possible* quorum votes should be reduced by two. The quorum disk drive vote is also removed.

3. Boot Node 2 again. This should reset its maintenance state. You should see the following message on both nodes:

```
NOTICE: CMM: Vote count changed from 0 to 1 for node
pnode2
```

4. Verify the cluster quorum configuration again. The number of possible quorum votes should be back to normal.



Task – Recovering from Cluster Amnesia

Perform the following steps to create a cluster amnesia problem and recover from it:

1. Verify that both nodes are active cluster members.
2. Shut down both nodes.

```
# scshutdown -y -g 10
```
3. Boot Node 1.
4. After Node 1 has completed its boot operation, log in as user `root` and shut it down again.

```
# init 0
```
5. Boot Node 2. Wait until Node 2 hangs waiting for a quorum.
6. Boot Node 1 again. Both nodes should complete the cluster software startup sequence and join in clustered operation.

Task – Booting Nodes in Non-cluster Mode

Perform the following steps to boot a cluster node so that it does not participate in clustered operation:

1. Shut down Node 2.
2. Boot Node 2 in non-cluster mode.

```
ok boot -x
```

Note – You should see a message similar to: Not booting as part of a cluster. You can also add the single-user mode option: `boot -sx`.

3. Verify the quorum status again.
4. Return Node 2 to clustered operation.

```
# init 6
```



Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Perform basic cluster startup and shutdown operations
- ☐ Boot nodes in non-cluster mode
- ☐ Place nodes in a maintenance state
- ☐ Verify cluster status from the command line
- ☐ Recover from a cluster amnesia error

Think Beyond

What strategies can you use to simplify administering a cluster with eight nodes and 200 storage arrays?

What strategies can you use to simplify administering a large installation of 20 clusters?

Volume Management Using VERITAS Volume Manager

Objectives

Upon completion of this module, you should be able to:

- Explain the disk space management technique used by VERITAS Volume Manager
- Describe the VERITAS Volume Manager initialization process
- Describe how the VERITAS Volume Manager groups disk drives
- Install and initialize VERITAS Volume Manager
- Perform VERITAS Volume Manager postinstallation configuration
- Use the basic VERITAS Volume Manager status commands
- Register VERITAS Volume Manager disk groups
- Create global file systems
- Perform basic device group management

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- Which VERITAS Volume Manager features are the most important to clustered systems?
- Are there any VERITAS Volume Manager feature restrictions when it is used in the Sun Cluster environment?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Disk Space Management

VERITAS Volume Manager manages data in a non-partitioned environment. VERITAS Volume Manager manages disk space by maintaining tables that associate a list of contiguous disk blocks with a data volume structure. A single disk drive can potentially be divided into hundreds of independent data regions.

As shown in Figure 7-1, VERITAS Volume Manager maintains detailed configuration records that equate specific blocks on one or more disk drives with virtual volume structures.

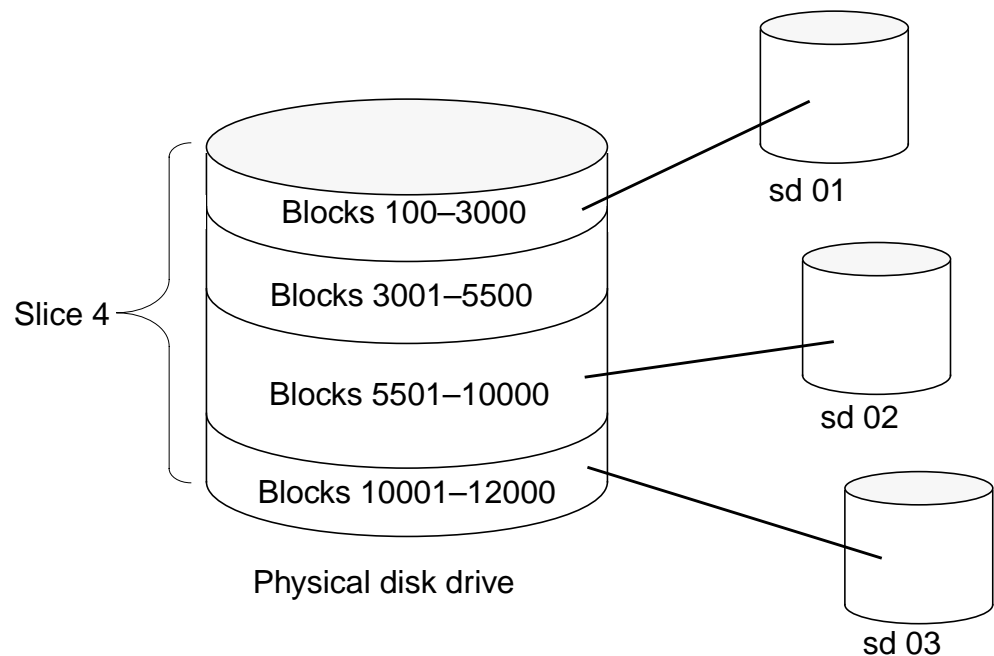


Figure 7-1 VERITAS Volume Manager Space Management

VERITAS Volume Manager divides a disk into a single slice and then allocates portions of the slice to data structures named *subdisks*. Subdisks are the basic storage space used to create VERITAS Volume Manager volumes.

VERITAS Volume Manager Disk Initialization

When a physical disk drive is initialized by VERITAS Volume Manager, it is divided into two sections called the *private* region and the *public* region.

The private and public regions are used for different purposes.

- The private region is used for configuration information.
- The public region is used for data storage.

As shown in Figure 7-2, the private region is small. It is usually configured as slice 3 on the disk and is, at most, a few cylinders in size.

The public region is the rest of the disk drive. It is usually configured as slice 4 on the disk.

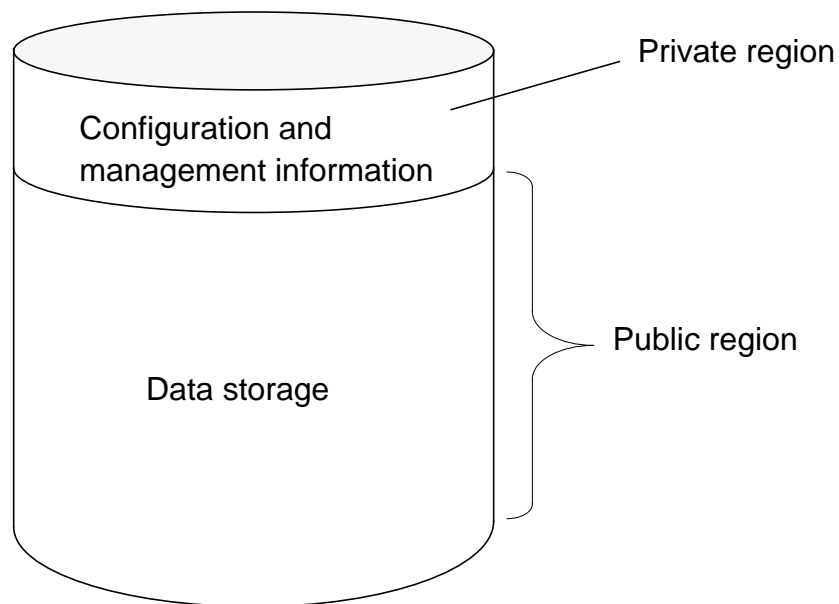


Figure 7-2 VERITAS Volume Manager Disk Initialization



Note – You must specially initialize all disk drives that VERITAS Volume Manager uses unless they have existing data you want to preserve. To preserve existing data, you *encapsulate* the disk. The VERITAS Volume Manager encapsulation process is described in the next section of this module.

Private Region Contents

The size of the private region, by default, is 1024 sectors (512 Kbytes). You can enlarge it if a large number of VERITAS Volume Manager objects are anticipated in a disk group. If you anticipate having more than 2000 VERITAS Volume Manager objects in a disk group, increase the size of the private region on the disks that are added to the disk group. The private region contents are:

- Disk header

Two copies of the file that defines and maintains the host name of the current disk group owner, a unique disk ID (DID), disk geometry information, and disk group association information.

- Table of contents

The disk header points to this linked list of blocks.

- Configuration database

This database contains persistent configuration information for all of the disks in a disk group. It is usually referred to as the `configdb` record.

- Disk group log

This log is composed of kernel-written records of certain types of actions, such as transaction commits, plex detaches resulting from I/O failures, dirty region log (DRL) failures, first write to a volume, and volume close. The VERITAS Volume Manager software uses this information after a crash or clean reboot to recover the state of the disk group just before the crash or reboot.



Note – A VERITAS Volume Manager object is a volume, a plex, a subdisk, or a disk group. Volumes are created from plexes. A plex is built from one or more subdisks. A single mirrored volume in a disk group consists of six objects: two subdisks, two plexes, the volume, and the disk group.

Private and Public Region Format

The private and public region format of an initialized VERITAS Volume Manager disk can be verified with the `prtvtoc` command. As shown in the following example, slice 2 is defined as the entire disk. Slice 3 has been assigned tag 15 and is 2016 sectors in size. Slice 4 has been assigned tag 14 and is the rest of the disk.

In this example, the private region is the first two cylinders on the disk. The disk is a 1.05-Gbyte disk, and a single cylinder has 1008 sectors or blocks, which does not meet the 1024-sector minimum size for the private region. This is calculated by using the `nhead=14` and `nsect=72` values for the disk found in the `/etc/format.dat` file.

```
# prtvtoc /dev/rdisk/c2t4d0s2
```

Partition	Tag	Flags	First Sector	Sector Count	Last Sector
2	5	01	0	2052288	2052287
3	15	01	0	2016	2015
4	14	01	2016	2050272	2052287

Initialized Disk Types

By default, VERITAS Volume Manager initializes disk drives with the type Sliced. There are other possible variations. The three types of initialized disks are:

- Simple – The private and public regions are on the same partition.
- Sliced – The private and public regions are on different partitions (default).
- `nopriv` – The disk does not have a private region.

Note – Do not use the `nopriv` format. It is normally used only for random access memory (RAM) disk storage on non-Sun systems.



VERITAS Volume Manager Disk Groups

VERITAS Volume Manager uses the term *disk group* to define a related collection of disk drives. The disk groups are given unique names and current ownership is assigned to a single cluster node.

The term “dg” is commonly used when referring to a disk group.

As shown in Figure 7-3, VERITAS Volume Manager disk groups are owned by an individual node and the `hostname` of that node is written onto private regions on the physical disks.

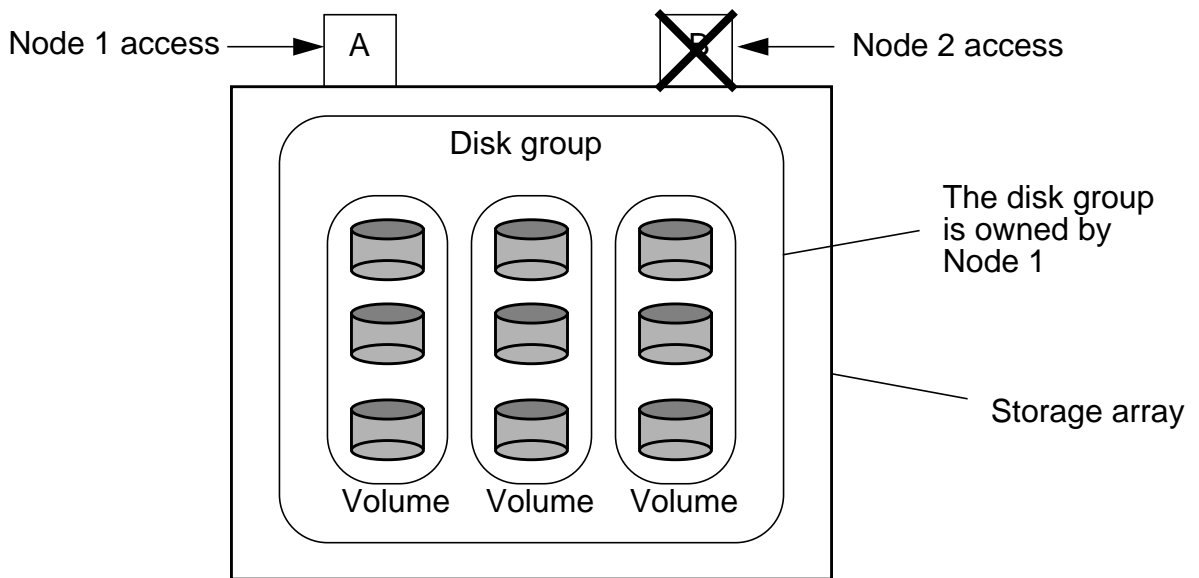


Figure 7-3 VERITAS Volume Manager Disk Group Ownership

Even though another node is physically connected to the same array, it cannot access data in the array that is part of a disk group it does not own. During a node failure, the disk group ownership can be transferred to another node that is physically connected to the array. To take ownership of a disk group, a node uses a VERITAS Volume Manager command to *import* the disk group. This is the backup node scheme used by all of the supported high-availability data services.

VERITAS Volume Manager Status Commands

Although the graphical user interface (GUI) for VERITAS Volume Manager furnishes useful visual status information, the most reliable and the quickest method of checking status is from the command line. Command line status tools are easy to use in script files, cron jobs, and remote logins.

Checking Volume Status

The VERITAS Volume Manager `vxprint` command is the easiest way to check the status of all volume structures. The following `vxprint` sample output shows the status of two plexes in a volume as bad. One of the plexes is a log.

```
# vxprint
```

```
Disk group: sdg0
```

TY	NAME	ASSOC	KSTATE	LENGTH	PLOFFS	STATE
dg	sdg0	sdg0	-	-	-	-
dm	disk0	c4t0d0s2	-	8368512	-	-
dm	disk7	c5t0d0s2	-	8368512	-	-
v	vol0	fsgen	ENABLED	524288	-	ACTIVE
pl	vol0-01	vol0	DISABLED	525141	-	IOFAIL
sd	disk0-01	vol0-01	ENABLED	525141	0	-
pl	vol0-02	vol0	ENABLED	525141	-	ACTIVE
sd	disk7-01	vol0-02	ENABLED	525141	0	-
pl	vol0-03	vol0	DISABLED	LOGONLY	-	IOFAIL
sd	disk0-02	vol0-03	ENABLED	5	LOG	-



Note – You can use the `vxprint -ht vol0` command to obtain a detailed analysis of the volume. This gives you all the information you need, including the physical path to the bad disk.

You can also use the `vxprint` command to create a backup configuration file that is suitable for re-creating the entire volume structure. This is useful as a worst-case disaster recovery tool.

Checking Disk Group Status

You can use the `vxldg` command to display a brief summary of disk groups that are currently owned (imported) by a particular node.

```
# vxldg list
NAME          STATE          ID
rootdg        enabled        992055967.1025.ns-east-104
nfsdg         enabled        992276322.1136.ns-east-103
webdg         enabled        992277026.1136.ns-east-104
```

Checking Disk Status

When disk drives fail, the VERITAS Volume Manager software can lose contact with a disk and no longer displays the physical path with the `vxprint -ht` command. At those times, you must find the media name of the failed disk from the `vxprint` command and then use the `vxldisk list` command to associate the media name with the physical device.

```
# vxldisk list
```

DEVICE	TYPE	DISK	GROUP	STATUS
c0t0d0s2	sliced	-	-	error
c0t1d0s2	sliced	disk02	rootdg	online
-	-	disk01	rootdg	failed was:c0t0d0s2

When a disk fails and becomes detached, the VERITAS Volume Manager software cannot currently find the disk but still knows the physical path. This is the origin of the `failed was` status, which means that the disk has failed and the physical path was the value displayed.

Saving Configuration Information

You can also use the `vxprint` and `vxldisk` commands to save detailed configuration information that is useful in disaster-recovery situations. Copy the output of the following commands into a file and store it on tape. You should also keep a printed copy of the files.

```
# vxprint -ht > filename
# vxldisk list > filename
```


Optimizing Recovery Times

In the Sun Cluster environment, data volumes are frequently mirrored to achieve a higher level of availability. If one of the cluster hosts system fails while accessing a mirrored volume, the recovery process might involve several steps, including:

- Synchronizing mirrors
- Checking file systems

Mirror synchronization can take a long time and must be completed before you can check file systems. If your cluster uses many large volumes, the complete volume recovery process can take hours.

You can expedite mirror synchronization by using the VERITAS Volume Manager dirty region logging (DRI) feature.

You can expedite file system recovery by using the Solaris 8 Operating Environment UNIX file system (UFS) logging feature.

Dirty Region Logging

A DRL is a VERITAS Volume Manager log file that tracks data changes made to mirrored volumes. The DRL speeds recovery time when a failed mirror needs to be synchronized with a surviving mirror.

- Only those regions that have been modified need to be synchronized between mirrors.
- Improper placement of DRLs can negatively affect performance.

A volume is divided into regions and a bitmap (where each bit corresponds to a volume region) is maintained in the DRL. When a write to a particular region occurs, the respective bit is set to on. When the system is restarted after a crash, this region bitmap limits the amount of data copying that is required to recover plex consistency for the volume. The region changes are logged to special log subdisks linked with each of the plexes associated with the volume.

The Solaris Operating Environment UFS Logging

UFS logging is a standard feature of the Solaris 8 Operating Environment. All Sun Cluster global file systems are mounted using the `logging mount` option.

If the logging option is specified for a file system, then logging is enabled for the duration of the mounted file system. Logging is the process of storing transactions (changes that make up a complete UFS operation) in a log before the transactions are applied to the file system. After a transaction is stored, the transaction can be applied to the file system. This prevents file systems from becoming inconsistent, therefore eliminating the need to run `fsck`. And, because `fsck` can be bypassed, logging reduces the time required to reboot a system if it crashes or after an unclean halt. The default behavior is no logging.

The log is allocated from free blocks on the file system, and is sized approximately 1 Mbyte per 1 Gbyte of file system, up to a maximum of 64 Mbytes. Logging can be enabled on any UFS, including `root (/)`. The log created by UFS logging is continually flushed as it fills up. The log is totally flushed when the file system is unmounted or as a result of the `lockfs -f` command.

VERITAS Volume Manager Installation Overview

The basic VERITAS Volume Manager installation process consists of the following:

- Ensuring Dynamic Multipathing is not enabled
- Installing VERITAS Volume Manager packages
- Verifying the `vxio` driver major numbers after installing VERITAS Volume Manager
- Setting unique `rootdg` minor numbers

VERITAS Volume Manager Dynamic Multipathing

The Dynamic Multipath (DMP) driver is a VERITAS Volume Manager product feature. As shown in Figure 7-4, the DMP driver can access the same storage array through more than one path. The DMP driver automatically configures multiple paths to the storage array if they exist. Depending on the storage array model, the paths are used for load-balancing in a primary or backup mode of operation.

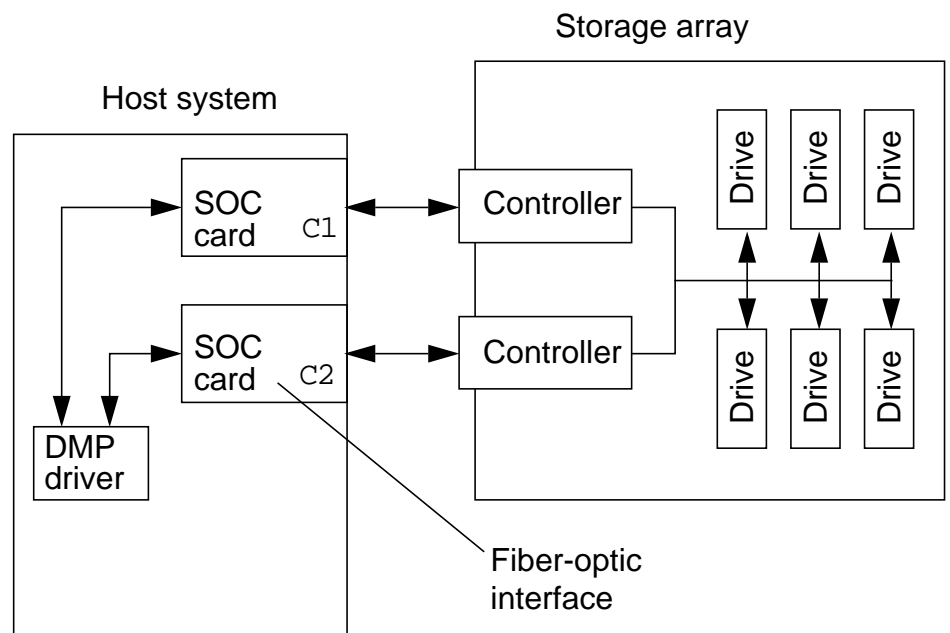


Figure 7-4 Dynamic Multipath Driver

Disabling VERITAS Volume Manager DMP

The DMP feature is automatically enabled when you install the VERITAS Volume Manager software. When you install VERITAS Volume Manager version 3.1 or older, you can take steps to prevent this from happening.

If you install a version newer than 3.1, you cannot permanently disable the DMP feature. Even if you take steps to disable DMP, it automatically reenables itself each time the system is booted. You must ensure that dual-paths to a storage device are not connected to a single host system.

To prevent VERITAS Volume Manager version 3.1 and earlier from enabling the DMP feature, perform the following procedure *before* installing the VERITAS Volume Manager software:

1. Symbolically link `/dev/vx/dmp` to `/dev/dsk` and `/dev/vx/rdmp` to `/dev/rdsk`.

```
# mkdir /dev/vx
# ln -s /dev/dsk /dev/vx/dmp
# ln -s /dev/rdsk /dev/vx/rdmp
```

Installing the VERITAS Volume Manager Software

You can manually install the VERITAS Volume Manager software on each node using the `pkgadd` utility. Not all of the packages are necessary. A summary of the packages follows.

VRTSVmdev	Header and library files
VRTSvmdoc	VERITAS Portable Document Format (PDF) or PostScript™ format documents
VRTSvmman	VERITAS manual pages
VRTSvmsa	Volume Manager Storage Administrator
VRTSvxvm	VERITAS binaries and related files

Run the `pkgadd` command on all nodes to install the VERITAS Volume Manager software.

```
# pkgadd -d . VRTSVmdev VRTSvmman VRTSvxvm
```

Note – The VERITAS document package `VRTSvmdoc` and the graphic management tool in `VRTSvmsa` are not required. The manual pages package, `VRTSvmman`, is not required, but you should always install it.



Verifying the vxio Driver Major Numbers

During VERITAS Volume Manager software installation, device drivers are assigned a major number in the `/etc/name_to_major` file. Unless these numbers are the same on High Availability (HA) for NFS primary and backup host systems, the HA for NFS users receive “Stale file handle” error messages after an HA for NFS logical host migrates to a backup system. This effectively terminates the user session and destroys the high-availability feature.

It makes no difference what the major numbers are as long as they agree on both host systems attached to a disk group or storage array and are unique. Check all nodes associated with a HA for NFS logical host as follows:

```
# grep vxio /etc/name_to_major
vxio 45
```

Make sure that the number is unique in all of the files. Change one so that they all match or, if that is not possible, assign a completely new number in all of the files.



Note – The current standard for the Sun Cluster 3.0 7/01 release is to set the vxio major number to 210 on all nodes and to resequence the rootdg file system minor numbers so they start at 50 on the first node and increment by 50 on each additional node (50, 100, 150, 200).

If your boot disk is not encapsulated, you can stop all activity on the nodes and edit the `/etc/name_to_major` files so they all agree.



Note – If your boot disk has been encapsulated, the process is more complex. Consult the *SunTM Cluster 3.0 07/01 Installation Guide* for detailed instructions.



Caution – You must stop the Sun Cluster software before making changes to the vxio driver major number.

Setting Unique rootdg Minor Device Numbers

After the `vxio` driver major numbers are set to the recommended value of 210 and the system is rebooted, the minor device numbers for the `rootdg` disk group is automatically set to a default value on the system as shown.

```
# ls -l /dev/vx/dsk/rootdg
total 0
brw----- 1 root root 210,7 Apr 4 10:48 rootdisk_4vol
brw----- 1 root root 210,8 Apr 4 10:48 rootdisk_7vol
brw----- 1 root root 210, 0 Mar 30 16:37 rootvol
brw----- 1 root root 210,9 Mar 30 16:37 swapvol
```

The `rootdg` minor number must be unique on each cluster node. The recommended numbering scheme is to set the base `rootdg` minor number to 50 on the first node and 100 on the second node. The `rootdg` minor number on additional nodes should increase in increments of 50.

Use the following command to resequence the `rootdg` minor device numbers on each cluster node:

```
# vxdg reminor rootdg 50
```

Initializing the rootdg Disk Group

After the basic VERITAS Volume Manager installation is complete, you must establish a `rootdg` disk group on each cluster node. The `rootdg` disk group is private to each node.

The VERITAS Volume Manager software cannot start until a private disk group named `rootdg` is created on a node.

There are three ways to satisfy this requirement:

- Initialize any storage array disk, and add it to the `rootdg` disk group
- Initialize any local non-root disk, and add it to the `rootdg` disk group
- Encapsulate the system boot disk

Creating a Simple rootdg Disk Group

If you do not want to encapsulate the boot disk on each of your cluster nodes, you can create a `rootdg` disk group that contains a single disk drive in a storage array. An example of such a configuration is shown in Figure 7-5.

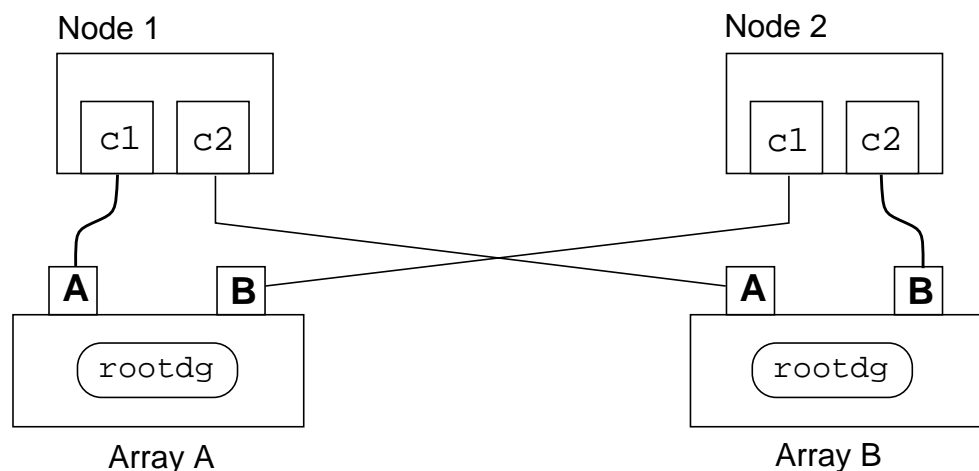


Figure 7-5 Storage Array `rootdg` Disk Group

The only difficulty with this configuration is that VERITAS Volume Manager scans all attached disks during startup to find configuration information. Each node's own `rootdg` disk displays error messages about finding another `rootdg` disk with the wrong ownership information. Ignore the error.

Encapsulating the System Boot Disk

To preserve existing data on a disk, you can choose to *encapsulate* the disk instead of initializing it. The VERITAS Volume Manager encapsulation process preserves existing file systems, such as those on your system boot disk.

When you install the VERITAS Volume Manager software on a system, you can place your system boot disk under VERITAS Volume Manager control using the `vxinstall` program or the new Sun™ Cluster 3.0 07/01 `scvxinstall` utility.

Preferred Boot Disk Configuration

Although there are many possible boot disk variations, the preferred boot disk configuration is shown in Figure 7-6.

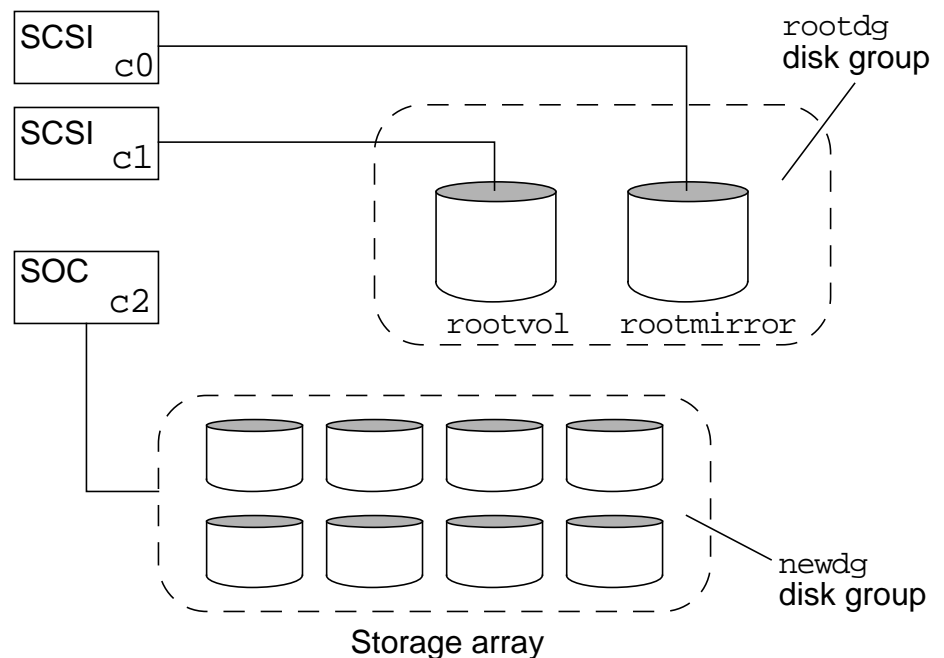


Figure 7-6 Preferred Boot Disk Configuration

The preferred configuration has the following features:

- The boot disk and mirror are on separate interfaces
- The boot disk and mirror are not in a storage array
- Only the boot disk and mirror are in the `rootdg` disk group

Prerequisites for Boot Disk Encapsulation

For the boot disk encapsulation process to succeed, the following prerequisites must be met:

- The disk must have at least two unused slices
- The boot disk should not have any slices in use other than the following:
 - `root`
 - `swap`
 - `/globaldevices`
 - `var`
 - `usr`

An additional prerequisite that is desirable but not mandatory: There should be at least 1024 sectors of unused space on the disk. Practically, this is at least two full cylinders at either the beginning or end of the disk.



Note – This is needed for the private region. If necessary, VERITAS Volume Manager takes the space from the end of the swap partition.

Primary and Mirror Configuration Differences

When you encapsulate your system boot disk, the location of all data remains unchanged even though the partition map is modified.

When you mirror the encapsulated boot disk, the location of the data on the mirror is different from the original boot disk.

During encapsulation, a copy of the system boot disk partition map is made so that the disk can later be returned to a state that allows booting directly from a slice again.

The original boot disk can be returned to its unencapsulated format with the assistance of VERITAS programs, such as the `vxunroot` utility.

The /etc/vfstab File

The original boot device mount information is commented out and retained before the new boot disk path names are configured. The following /etc/vfstab file entries are typical for an encapsulated boot disk with a single partition root file system and a swap partition.

#device	device	mount	FS	fsck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
/dev/vx/dsk/swapvol-		-	swap	-	no	
/dev/vx/dsk/rootvol	/dev/vx/rdsk/rootvol	/	ufs	1	no	-

#NOTE: volume rootvol (/) encapsulated partition c0t0d0s0
#NOTE: volume swapvol (swap) encapsulated partition c0t0d0s1

Sun Cluster Boot Disk Encapsulation Process

Manual encapsulation of the boot disk on the cluster nodes is a complex procedure. The `rootdg` disk group major device number is assigned by the `vxio` driver so that the `rootdg` disk group on two or more nodes has the same major device number. Unless corrective measures are taken, the `rootdg` disk group file systems also have the same minor numbers. This is a severe problem, and you must correct it before you start the cluster operation.

You must configure unique `rootdg` file system names and minor numbers on each node. Special steps must also be taken to ensure that the `/global` file systems are still available on each node.

Sun Cluster Manual Encapsulation Process

The general process for encapsulating the boot disks in a Sun™ Cluster 3.0 07/01 environment is as follows:

1. Install the VERITAS Volume Manager software on Node 1.
2. Set the `vxio` major number to 210 on all nodes.
3. Use `vxinstall` to encapsulate the boot disk on Node 1.
 - a. Choose a unique name for the boot disk.
 - b. Do not accept automatic reboot.
 - c. Do not add any other disks to the `rootdg` disk group.
4. Change the `/global` mount entry in `vfstab` to use the original logical device path instead of the DID device path.
5. Repeat previous steps on all cluster nodes.
6. Shut down the cluster nodes with `scshutdown`.
7. Boot all nodes in non-cluster mode.
8. Bypass any `/global` `fsck` errors on nodes by pressing Control-D.

One node successfully mounts its `/global` file system.

VERITAS Volume Manager finishes the basic boot disk encapsulation process during the reboot on each node.

9. Unmount the one successful `/global` file system.

```
# umount /global/.devices/node@nodeid
```

10. Re-minor the rootdg disk group on each cluster node.

```
# vxdg reminor rootdg 50
```

Use a different minor number on each node, increment the base number by 50 on each node.

11. Verify the root disk volume minor numbers are unique on each node.

```
# ls -l /dev/vx/dsk/rootdg
total 0
brw----- 1 root root 210,50 Apr 4 10:48 rootdisk_4vol
brw----- 1 root root 210,51 Apr 4 10:48 rootdisk_7vol
brw----- 1 root root 210, 0 Mar 30 16:37 rootvol
brw----- 1 root root 210,52 Mar 30 16:37 swapvol
```

The swap volume is automatically renumbered after a reboot.

12. If there are not separate /var or /usr file systems, then shut down the cluster, and reboot each node in cluster mode.



Note – If the encapsulated boot disks had a separate /var or /usr partition before encapsulation, you must perform additional work before rebooting. Consult Appendix B of the *Sun™ Cluster 3.0 07/01 Installation Guide* for additional information.

13. After you reboot the nodes, you can mirror the boot disk.



Note – Mirroring the boot disk also requires additional configuration that can be found in Appendix B of the *Sun™ Cluster 3.0 07/01 Installation Guide*.

Sun Cluster Automated Encapsulation Process

The Sun™ Cluster 3.0 07/011 software release has a new utility, `scvxinstall`, that provides assistance when installing and configuring VERITAS Volume Manager software on Sun Cluster hosts. The `scvxinstall` utility can automatically perform the following tasks:

- Disable Dynamic Multipathing (DMP)
- Install the VERITAS Volume Manager software packages and a license key
- Set the `vxio` driver major numbers
- Set the `rootdg` disk group minor device numbers
- Encapsulate the root disk
- Correct `/global` entries in the `/etc/vfstab` file



Note – You must still ensure that the boot disk partitioning requirements are satisfied on all nodes before starting the `scvxinstall` utility.

The `scvxinstall` utility has command options that provide the following functionality:

- Install the VERITAS software packages, do not encapsulate the boot disk
- If necessary, install the VERITAS software packages, then encapsulate the boot disk

There are also options to specify a license key and the path to the VERITAS software packages. The `-s` option displays the current installation status as shown in the following example:

```
# scvxinstall -s
DMP is disabled.
The Volume Manager package installation step is complete.
The vxio major number is set to 210.
The "PHOTON" Volume Manager feature is licensed.
The Volume Manager root disk encapsulation step is complete
The /etc/vfstab file includes the necessary updates.
The rootdg remingoring step is complete.
```

If the `scvxinstall` utility is started without any command options, it runs in an interactive mode and prompts for all necessary information.

The following is a summary of the `scvxinstall` startup phase:

```
# scvxinstall
Do you want Volume Manager to encapsulate root [no]? yes
Where is the Volume Manager cdrom? /LabFiles/VM_3.1
Disabling DMP.
Installing packages from /LabFiles/VM_3.1.
Installing VRTSvxvm.
Installing VRTSvmdev.
Installing VRTSvmman.
Setting the vxio major number to 210.
Volume Manager installation is complete.
The "PHOTON" Volume Manager feature is already licensed.
One or more Volume Manager features are already licensed.
If you do not want to supply an additional license key,
just press ENTER. Otherwise, you may provide one
additional key.
Please enter a Volume Manager license key [none]: none

The Volume Manager root disk encapsulation step will begin
in 20 seconds.
Type Ctrl-C to abort .....
Verifying encapsulation requirements.
Arranging for Volume Manager encapsulation of the root
disk.
The vxconfigd daemon has been started and is in disabled
mode...
Reinitialized the volboot file...
Created the rootdg...
Added the rootdisk to the rootdg...
The setup to encapsulate rootdisk is complete...
Updating /global/.devices entry in /etc/vfstab.

This node will be re-booted in 20 seconds.
Rebooting with command: boot -x
Not booting as part of a cluster
```

The node reboots two times. The first reboot completes the VERITAS boot disk encapsulation process, and the second reboot brings the node up in clustered operation again. After the configuration of the first node has completed, you run the `scvxinstall` utility on the next node.

Note – To mirror the boot disk on each node, there is a full procedure in the *SunTM Cluster 3.0 07/01 Installation Guide*.



Registering VERITAS Volume Manager Disk Groups

After you create a new VERITAS Volume Manager disk group, you must register the disk group using either the `scsetup` utility or the `scconf` command. The `scsetup` utility is recommended.

When a VERITAS Volume Manager disk group is registered in the Sun Cluster environment, it is referred to as a *disk device group*.

Until a VERITAS Volume Manager disk group is registered, the cluster does not detect it, and you cannot build file systems on volumes in the disk group. The `newfs` utility cannot find the volume paths. The VERITAS Volume Manager utilities, such as `vxprint` and `vxdisk`, show the disk group is present and the volumes are enabled and active.

The `scstat -D` command does not recognize disk groups until they are registered and become *disk device groups*.

When you register a disk group, you can associate it with several device group parameters. You must associate it with a list of attached nodes. The first node in the list is the primary node. When the cluster is first booted, each node imports disk groups for which it is the assigned primary.

The secondary node takes over the disk group in the event the primary node fails. The `preferenced` option enables the order in which nodes attempt to take over as primary for a disk device group.

You can also establish a failback policy. If failback is enabled, the disk group always migrates back to the primary node as soon as it becomes available again.

A typical `scconf` command to register a disk group is as follows:

```
# scconf -a -D type=vxvm,name=webdg, \  
nodelist=pnode2:pnode1, \  
preferenced=true,failback=enabled
```



Caution – Do not use VERITAS commands to deport and import disk groups in the Sun™ Cluster 3.0 07/01 environment. Use the `scsetup` utility to register and unregister VERITAS Volume Manager disk groups, to take them online and offline, and to synchronize volume changes within a disk group.

Device Group Policies

When you first register a VERITAS Volume Manager disk group with the Sun Cluster framework, you can associate a list of nodes with the disk group. You can change the node list and associated policies with the `scconf -c` command. An example follows.

```
# scconf -c -D name=nfsdg,nodelist=pnode1:pnode2
```

You can apply a preferred node policy to the node list along with a failback policy.

- Preferred node policy

If the `preferenced` option is set to `true`, the first node (primary) in the list is the preferred owner of the disk group. It imports the disk group when it boots. The second node in the list (secondary) does not import the disk group when it boots. If the primary node fails, the secondary node takes ownership of the disk group.

```
# scconf -c -D name=nfsdg,nodelist=pnode1:pnode2 \
,preferenced=true
```

If the `preferenced` option is set to `false`, any node in the node list that starts clustered operation imports the disk group and places the device group online. If both node are booted simultaneously, either one might import any or all of the available disk groups.

```
# scconf -c -D name=nfsdg,nodelist=pnode1:pnode2 \
,preferenced=false
```

- Failback policy

If the primary node fails, the secondary node automatically takes ownership of the disk group. If the failback policy is `true`, the disk group automatically migrates back (fail back) to the primary node when it boots again.

```
# scconf -c -D name=nfsdg,nodelist=pnode1:pnode2 \
,preferenced=true,failback=enabled
```

If the `preferenced` option is set to `false`, the failback feature automatically disables.

Exercise: Configuring Volume Management

In this exercise, you complete the following tasks:

- Install the VERITAS Volume Manager software
- Initialize the VERITAS Volume Manager software
- Create demonstration volumes
- Perform Sun Cluster disk group registration
- Create global file systems
- Modify device group policies

Preparation

Record the location of the VERITAS Volume Manager software you will install during this exercise.

Location: _____

During this exercise, you create a private `rootdg` disk group for each node and two data service disk groups. Each data service disk group contains a single mirrored volume, as shown in Figure 7-7.

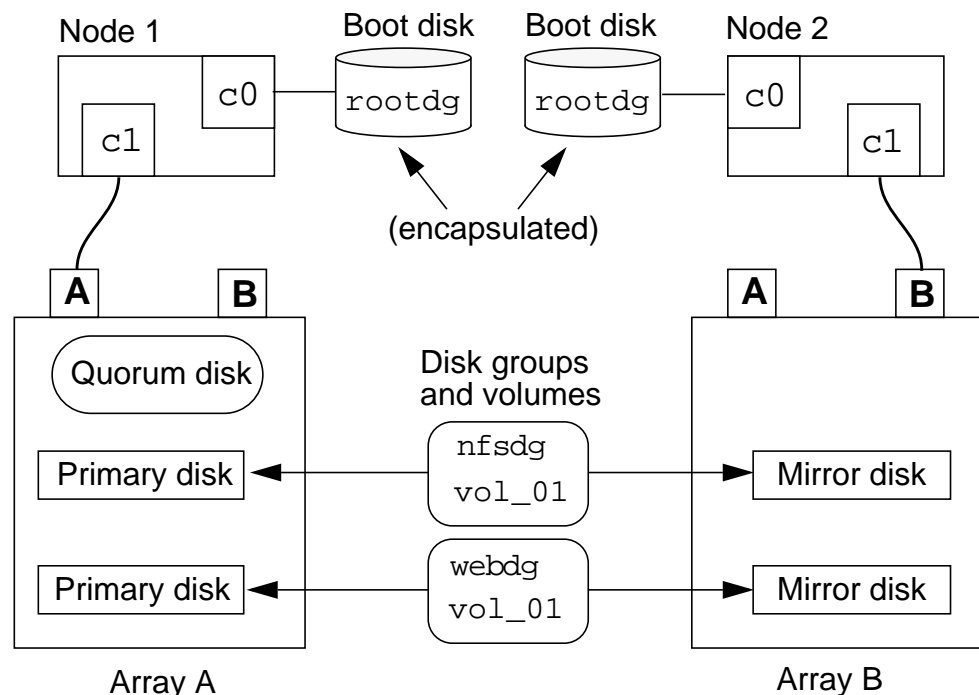


Figure 7-7 Configuring Volume Management



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Selecting Disk Drives

Before proceeding with this exercise, you must have a clear picture of the disk drives that are used throughout this course. You have already configured one disk drive as a quorum disk. In this exercise, you must identify the boot disks to be encapsulated and two disks in each storage array for use in the two demonstration disk groups, *nfsdg* and *webdg*.

Perform the following steps to select disk driver:

1. Use the `scdidadm -l` and `format` commands to identify and record the logical addresses of disks for use during this exercise. Use the address format: `c0t2d0`.

Quorum disk: _____

	<u>Node 1</u>	<u>Node 2</u>
Boot disks:	_____	_____
	<u>Array A</u>	<u>Array B</u>
<i>nfsdg</i> disks:	_____	_____
<i>webdg</i> disks:	_____	_____

Task – Using `scvxinstall` to Install and Initialize VERITAS Volume Manager Software

If you prefer to manually install and initialize VERITAS Volume Manager software, skip all of the subtasks in this section and go to the “Task – Manually Installing and Initializing VERITAS Volume Manager Software” on page 7-30.

Perform the following steps to install and initialize the VERITAS Volume Manager software on your cluster host systems:

1. Type the **scstat -q** command to verify that both nodes are joined as cluster members. Do not proceed until this requirement is met.
2. Start the `/usr/cluster/bin/scvxinstall` utility on Node 1.
3. Respond to the `scvxinstall` questions as follows:
 - a. Reply **yes** to root disk encapsulation.



Note – If you want to install VERITAS Volume Manager patches, you can reply **no** to boot disk encapsulation and just install the VERITAS packages without performing the encapsulation. You can rerun `scvxinstall` later, after the appropriate VERITAS patches have been added.

- b. Type the full path name to the VERITAS Volume Manager software packages.
 - c. Type **none** for the license key.



Note – Not all Sun storage devices are prelicensed. You might need a basic license key in your workplace.

4. After the installation is complete, log in as user `root` on Node 1.
5. Type the **vxprint** command to verify the `rootdg` disk group is configured and operational.
6. Start the `scvxinstall` utility on Node 2.
7. Respond to the `scvxinstall` questions as follows:
 - a. Reply **yes** to boot disk encapsulation.
 - b. Type the full path name to the VERITAS Volume Manager software packages.
 - c. Type **none** for the license key.
8. After the VERITAS configuration is complete on both nodes, perform the following steps on both node:
 - a. Type **scstat -q** and verify both nodes are cluster members.
 - b. Type **vxprint** on both nodes and verify that the names of the `rootdg` volumes are unique between nodes.
 - c. Type **ls -l /dev/vx/rdisk/rootdg** on both nodes to verify that the `rootdg` disk group minor device numbers are different on each node.
 - d. Type **scvxinstall -s** on both nodes and verify all of the installation and configuration steps have completed.

Task – Manually Installing and Initializing VERITAS Volume Manager Software

Do *not* perform the steps in this section if you have already initialized the VERITAS Volume Manager software using the `scvxinstall` utility.

Subtask – Disabling Dynamic Multipathing

Perform the following steps to prevent VERITAS Volume Manager from enabling DMP during installation:

1. On both nodes, symbolically link `/dev/vx/dmp` to `/dev/dsk` and `/dev/vx/rdmp` to `/dev/rdsk`.

```
# mkdir /dev/vx
# ln -s /dev/dsk /dev/vx/dmp
# ln -s /dev/rdsk /dev/vx/rdmp
```

Subtask – Installing the VERITAS Volume Manager Software

To install the VERITAS Volume Manager software, perform the following steps:

1. Move to the location of the VERITAS Volume Manager software on both nodes.
2. Verify you are in the correct location on both nodes.

```
# ls
VRTSvmdev  VRTSvmdoc  VRTSvmman  VRTSvmsa
VRTSvxvm
```

3. Run the `pkgadd` command on both nodes to begin the installation.

```
# pkgadd -d . VRTSvmdev VRTSvmman VRTSvxvm
```

Note – The VERITAS document package `VRTSvmdoc` and the graphic management tool in `VRTSvmsa` are not used in this course.



- a. Answer **yes** to the Continue installation question.
 - b. Answer **yes** to the `setuid/setgid` question.
 - c. Answer **yes** to the continue with `VRTSvxvm` question.
4. After the installation completes, verify that the VERITAS binaries and manual pages are installed as follows:

```
# ls /usr/sbin/vx*
# ls /etc/vx/bin
# ls /opt/VRTSvxvm/man
```

Subtask – Installing Patches

This is the appropriate time to install any necessary VERITAS Volume Manager patches. Check with your instructor to see if you need to install any VERITAS Volume Manager patches.

Subtask – Installing a VERITAS License

Many Sun storage arrays models are prelicensed for use with the basic VERITAS product features. If you are using non-array type disk storage, you might need to manually install a license key using the `vxlicense` command.

Subtask – Verifying the `vxio` Driver Major Numbers

Perform the following steps to ensure that the `vxio` driver major numbers are the same on all cluster nodes and that the number 210 is not already in use in the `name_to_major` file:

1. Verify and record the `vxio` driver major numbers on all nodes.

```
# grep vxio /etc/name_to_major
# grep 210 /etc/name_to_major
```

Node 1 `vxio` major number: _____

Node 2 `vxio` major number: _____

2. Edit the `/etc/name_to_major` file on each node and change the `vxio` major numbers to 210.



Caution – Before you modify the `vxio` major number on a node, you must first make sure the new number is not already in use on that node. If it is, you will have to select a unique and unused number for both nodes.

Subtask – Encapsulating the Boot Disk

Both node should be active cluster members. Perform the following steps to encapsulate the system boot disk on each node:



Note – There is an alternate section, “Task – Optional `rootdg` Disk Group Initialization” on page 7-34 that you can use to initialize the `rootdg` disk group. It creates a `rootdg` disk group for each node using storage array disks.

1. Run `vxinstall` on *both* nodes to encapsulate their boot disks.

`/usr/sbin/vxinstall`

- a. Select Custom Installation (2).
 - b. Answer yes (**y**) to Encapsulate Boot Disk.
 - c. Enter a unique disk name on each node (rootdisk1, rootdisk2)
 - d. Select Leave these disk alone (4) until you get to the final summary of your choices.
 - e. Reply yes (**y**) to the boot disk encapsulation.
 - f. Reply no (**n**) to Shutdown and reboot now.
2. On Node 1, change the `/global` mount entry in `vfstab` to use the logical device paths that were used in the original `/globaldevices` mount entry.

After modification, the change on Node 1 should look similar to the following:

```
/dev/dsk/c0t0d0s4 /dev/rdisk/c0t0d0s4 /global/.devices/node@1 ufs 2 no global
```

3. On Node 2, change the `/global` mount entry in `vfstab` to use the logical device paths that were used in the original `/globaldevices` mount entry.

The changes on Node 2 should look similar to the following:

```
/dev/dsk/c0t0d0s4 /dev/rdisk/c0t0d0s4 /global/.devices/node@2 ufs 2 no global
```

4. On one node, shut down the cluster with `scshutdowndown -y -g 15`.

The default for the grace period (`-g`) is 60 seconds.

5. Boot Node 1 in non-cluster mode (`ok boot -x`).

Note – The VERITAS Volume Manager software initiates a reboot after it finishes encapsulating the boot disk.



6. Boot Node 2 in non-cluster mode. During the automatic reboot, bypass any `/global fsck` errors by pressing Control-D.

One node successfully mounts its `/global` file system.

7. Unmount the one successful /global file system. This should be on Node 1.

```
# umount /global/.devices/node@1
```

8. Check the rootdg minor numbers on each node and compare them.

```
# ls -l /dev/vx/dsk/rootdg
total 0
brw----- 1 root root 210,5 Dec 21 16:47 rootdisk24vol
brw----- 1 root root 210,6 Dec 21 16:47 rootdisk27vol
brw----- 1 root root 210,0 Dec 21 16:47 rootvol
brw----- 1 root root 210,7 Dec 21 16:47 swapvol
```

9. Re-minor the rootdg disk group on Node 1.

```
# vxdg reminor rootdg 50
```

10. Re-minor the rootdg disk group on Node 2.

```
# vxdg reminor rootdg 100
```

11. Verify the root disk volume minor numbers are unique on each node.

```
# ls -l /dev/vx/dsk/rootdg
total 0
brw----- 1 root root 210,50 Apr 4 10:48 rootdiska3vol
brw----- 1 root root 210,51 Apr 4 10:48 rootdiska7vol
brw----- 1 root root 210, 0 Mar 30 16:37 rootvol
brw----- 1 root root 210,52 Mar 30 16:37 swapvol
```

Note – The rootvol and swapvol minor numbers are automatically renumbered after a reboot.



12. Shut down the cluster, and reboot each node in cluster mode. During the reboot, the following error message displays:

```
VxVM starting special volumes ( swapvol )...
/dev/vx/dsk/swapvol: No such device or address
```



Caution – If your boot disk had a separated /var/ or /usr partition before encapsulation, you must perform additional steps before rebooting. Consult Appendix B of the *Sun™ Cluster 3.0 07/01 Installation Guide* for additional information.

Task – Optional rootdg Disk Group Initialization

Do not perform this optional rootdg configuration unless your instructor has directed you to use this procedure instead of boot-disk encapsulation.

Perform the following steps:

1. Select a rootdg disk in each storage array. *Do not use the quorum disk drive or disks you previously selected for the demonstration disk groups.*
2. On each node, manually configure the rootdg disk using the disk you selected for that particular node.

```
# vxconfigd -m disable
# vxdctl init
# vxdg init rootdg
# vxdctl add disk logical_address type=sliced
# vxdisksetup -i logical_address
# vxdg adddisk logical_address
# vxdctl enable
# rm /etc/vx/reconfig.d/state.d/install-db
```

3. Shut down and reboot Node 1.

```
# init 6
```

4. After Node 1 has completed its reboot operation, shut down and reboot Node 2.
5. Verify that you see the following VERITAS Volume Manager messages when the nodes reboot.

```
VxVM starting in boot mode...
VxVM general startup...
```



Note – You will see warnings, such as vxvm:vxconfigd: WARNING: Disk c2t32d0s2 names group rootdg, but group ID differs, when the nodes boot. This means that there are other rootdg disk groups present that do not belong to this node.

6. Verify that the rootdg disk group is operational on both nodes.

```
# vxprint
Disk group: rootdg
TY NAME ASSOC KSTATE LENGTH PLOFFS STATE TUTILO PUTILO
dg rootdg rootdg - - - - - -
dm c2t52d0 c2t52d0s2 - 17678493 - - - -
```


Task – Configuring Demonstration Volumes

Perform the following steps to configure two demonstration disk group, each containing a single mirrored volume:

1. On Node 1, create the `nfsgdg` disk group with your previously selected logical disk addresses.

```
# vxdiskadd disk01 disk02
Which disk group [<group>,none,list,q,?]
(default: rootdg) nfsgdg
Create a new group named nfsgdg? [y,n,q,?]
(default: y) y
Use default disk names for these disks? [y,n,q,?]
(default: y) y
Add disks as spare disks for nfsgdg? [y,n,q,?] (default:
n) n
Exclude disks from hot-relocation use? [y,n,q,?]
(default: n) y
```



Caution – If you are prompted about encapsulating the disk, you should reply **no**. You might also be prompted about clearing old disk usage status from a previous training class. In your work environment, be careful when answering these questions because you can destroy critical data or cluster node access information.

2. Verify the status of the disk group and the names and ownership of the disks in the `nfsgdg` disk group.

```
# vxdg list
# vxdisk list
```

3. On Node 1, verify that the new `nfsgdg` disk group is globally linked.

```
# cd /dev/vx/dsk/nfsgdg
# pwd
/global/.devices/node@1/dev/vx/dsk/nfsgdg
```

4. On Node 1, create a 500-Mbyte mirrored volume in the `nfsgdg` disk group.

```
# vxassist -g nfsgdg make vol-01 500m layout=mirror
```

5. On Node 2, create the `webdg` disk group with your previously selected logical disk addresses.

```
# vxdiskadd disk03 disk04
Which disk group [<group>,none,list,q,?]
(default: rootdg) webdg
```

6. On Node 2, create a 500-Mbyte mirrored volume in the `webdg` disk group.

```
# vxassist -g webdg make vol-01 500m layout=mirror
```

7. Type the `vxprint` command on both nodes. Notice that each node does not see the disk group that was created on a different node.

Task – Registering Demonstration Disk Groups

Perform the following steps to register the two new disk groups with the Sun Cluster framework software:

1. On Node 1, use the `scconf` utility to manually register the `nfsdg` disk group.

```
# scconf -a -D type=vxvm,name=nfsdg, \
nodelist=node1:node2,preferenced=true,failback=enabled
```

Note – Put the local node (Node 1) first in the node list.



2. On Node 2, use the `scsetup` utility to register the `webdg` disk group.

```
# scsetup
```

3. From the main menu, select option 4, Device groups and volumes. Then from the device groups menu select option 1, Register a VxVM disk group as a device group.

4. Answer the `scsetup` questions as follows:

```
Name of the VxVM disk group you want to register? webdg
Do you want to configure a preferred ordering (yes/no)
[yes]? yes
```

```
Are both nodes attached to all disks in this group
(yes/no) [yes]? yes
```

```
Which node is the preferred primary for this device
group? node2
```

```
Enable "failback" for this disk device group (yes/no)
[no]? yes
```

Note – Make sure you specify `node2` as the preferred primary node. You might see warnings about disks configured by a previous class that still contain records about a disk group named `webdg`.



5. From either node, verify the status of the disk groups.

```
# scstat -D
```

Note – Until a disk group is registered, you cannot create file systems on the volumes. Even though `vxprint` shows the disk group and volume as being active, the `newfs` utility cannot detect it.



Task – Creating a Global `nfs` File System

Perform the following steps on Node 1 to create and mount a demonstration file system on the `nfsdg` disk group volume:

1. On Node 1, create a file system on `vol-01` in the `nfsdg` disk group.

```
# newfs /dev/vx/rdisk/nfsdg/vol-01
```

2. On *both* Node 1 and Node 2, create a global mount point for the new file system.

```
# mkdir /global/nfs
```

Note – If you do not create the mount point on *both* nodes, you will not be able to mount the `/global/nfs` file system from one of the nodes.

3. On *both nodes*, add a mount entry in the `/etc/vfstab` file for the new file system with the `global` and `logging` mount options.

```
/dev/vx/dsk/nfsdg/vol-01 /dev/vx/rdisk/nfsdg/vol-01 \  
/global/nfs ufs 2 yes global,logging
```

Note – Do not use the line continuation character (`\`) in the `vfstab` file.

4. On Node 1, mount the `/global/nfs` file system. The mount might take a while.

```
# mount /global/nfs
```

5. Verify that the file system is mounted and available on *both* nodes.

```
# mount  
# ls /global/nfs  
lost+found
```



Task – Creating a Global web File System

Perform the following steps on Node 2 to create and mount a demonstration file system on the webdg disk group volume:

1. On Node 2, create a file system on vol_01 in the webdg disk group.

```
# newfs /dev/vx/rdsk/webdg/vol-01
```

2. On *both nodes*, create a global mount point for the new file system.

```
# mkdir /global/web
```

3. On *both nodes*, add a mount entry in the /etc/vfstab file for the new file system with the global and logging mount options.

```
/dev/vx/dsk/webdg/vol-01 /dev/vx/rdsk/webdg/vol-01 \  
/global/web ufs 2 yes global,logging
```

Note – Do not use the line continuation character (\) in the vfstab file.

4. On Node 2, mount the /global/web file system.

```
# mount /global/web
```

5. Verify that the file system is mounted and available on *both nodes*.

```
# mount  
# ls /global/web  
lost+found
```



Task – Testing Global File Systems

Perform the following steps to confirm the general behavior of globally available file systems in the Sun™ Cluster 3.0 07/01 environment:

1. On Node 2, move into the `/global/nfs` file system.

```
# cd /global/nfs
```
2. On Node 1, try to unmount the `/global/nfs` file system (`umount /global/nfs`). You should get an error that the file system is busy.
3. On Node 2, move out of the `/global/nfs` file system (`cd /`) and try to unmount it again on Node 1.
4. Mount the `/global/nfs` file system again on Node 1.
5. Try unmounting and mounting `/global/nfs` from both nodes.

Task – Managing Disk Device Groups

In the Sun™ Cluster 3.0 07/01 environment, VERITAS disk groups become *disk device groups* when they are registered. In most cases, they should *not* be managed using VERITAS commands. Some administrative tasks are accomplished using a combination of Sun Cluster and VERITAS commands. Common tasks are:

- Adding volumes to a disk device group
- Removing volume from a disk device group

Adding a Volume to a Disk Device Group

Perform the following steps to add a volume to an existing device group:

1. Make sure the *device group* is online (to Sun Cluster).

```
# scstat -D
```

Note – You can bring it online to a selected node as follows:

```
# scswitch -z -D nfsdg -h node1
```

2. Determine which node is currently mastering the related VERITAS *disk group* (has it imported).

```
# vxdg list  
# vxprint
```



3. On Node 1, create a 50-Mbyte test volume in the `nfsgdg` disk group.

```
# vxassist -g nfsgdg make testvol 50m layout=mirror
```
4. Verify the status of the volume `testvol`.

```
# vxprint testvol
```
5. Perform the following steps to register the changes to the `nfsgdg` disk group configuration.
 - a. Start the `scsetup` utility.
 - b. Select menu item 4, Device groups and volumes.
 - c. Select menu item 2, Synchronize volume information.
 - d. Supply the name of the disk group and quit `scsetup` when the operation is finished.

Note – The command line equivalent is:

```
scconf -c -D name=nfsgdg, sync
```



Removing a Volume From a Disk Device Group

To remove a volume from a disk device group, perform the following steps on the node that currently has the related disk group imported:

1. Unmount any file systems that are related to the volume.
2. On Node 1, recursively remove the test volume, `testvol`, from the `nfsgdg` disk group.

```
# vxedit -g nfsgdg -rf rm testvol
```
3. Register the `nfsgdg` disk group configuration change with the Sun Cluster framework.

Note – You can use either the `scsetup` utility or `scconf` as follows:

```
scconf -c -D name=nfsgdg, sync
```



Migrating Device Groups

The `scconf -p` command is the best method of determining current device group configuration parameters. Perform the following steps to verify device group behavior:

1. Verify the current demonstration device group configuration.

```
# scconf -p |grep group
Device group name:                webdg
Device group type:                VxVM
Device group failback enabled:    no
Device group node list:          pnode2, pnode1
Device group ordered node list:   yes
Diskgroup name:                  webdg
Device group name:                nfsdg
Device group type:                VxVM
Device group failback enabled:    no
Device group node list:          pnode1, pnode2
Device group ordered node list:   yes
Diskgroup name:                  nfsdg
```

2. From either node, switch the `nfsdg` device group to Node 2.

```
# scswitch -z -D nfsdg -h node2 (use your node name)
```

3. Type the `init 0` command to shut down Node 1.
4. Boot Node 1. The `nfsdg` disk group should automatically migrate back to Node 1.



Note – Device groups are only part of the resource groups that can migrate between cluster nodes. There are other resource group components, such as IP addresses and data services that also migrate as part of a resource group. This section is intended only as a basic introduction to resource group behavior in the cluster environment. Normally you migrate resource groups and not their component parts such as device groups and IP addresses.

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Explain the disk space management technique used by VERITAS Volume Manager
- ☐ Describe the VERITAS Volume Manager initialization process
- ☐ Describe how the VERITAS Volume Manager groups disk drives
- ☐ Install and initialize VERITAS Volume Manager
- ☐ Perform VERITAS Volume Manager postinstallation configuration
- ☐ Use the basic VERITAS Volume Manager status commands
- ☐ Register VERITAS Volume Manager disk groups
- ☐ Create global file systems
- ☐ Perform basic device group management

Think Beyond

Where does VERITAS Volume Manager recovery fit into the Sun Cluster environment?

Is VERITAS Volume Manager required for high-availability functionality?

Volume Management Using Solstice DiskSuite™

Objectives

Upon completion of this module, you should be able to:

- Explain the disk space management technique used by Solstice DiskSuite
- Describe the Solstice DiskSuite initialization process
- Describe how Solstice DiskSuite groups disk drives
- Use Solstice DiskSuite status commands
- Describe the Solstice DiskSuite software installation process
- Install and initialize Solstice DiskSuite
- Perform Solstice DiskSuite postinstallation configuration
- Create global file systems
- Perform basic device group management

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- Which Solstice DiskSuite features are the most important to clustered systems?
- What relationship does Solstice DiskSuite have to normal cluster operation?
- Are there any restrictions on Solstice DiskSuite features when Solstice DiskSuite is used in the Sun Cluster environment?

Additional Resources



Additional resources – The following references can provide additional information on the topics described in this module:

- *Sun™ Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *Sun™ Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *Sun™ Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *Sun™ Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *Sun™ Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *Sun™ Cluster 3.0 07/01 Concepts*, part number 806-7074
- *Sun™ Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *Sun™ Cluster 3.0 07/01 Release Notes*, part number 806-7078

Disk Space Management

The Solstice DiskSuite software manages disk space by associating standard UNIX partitions with a data volume structure. You can divide a single disk drive into only seven independent data regions, which is the UNIX partition limit for each physical disk.

Solstice DiskSuite Disk Space Management

As shown in Figure 8-1, Solstice DiskSuite manages virtual volume structures by equating standard UNIX disk partitions with virtual volume structures.

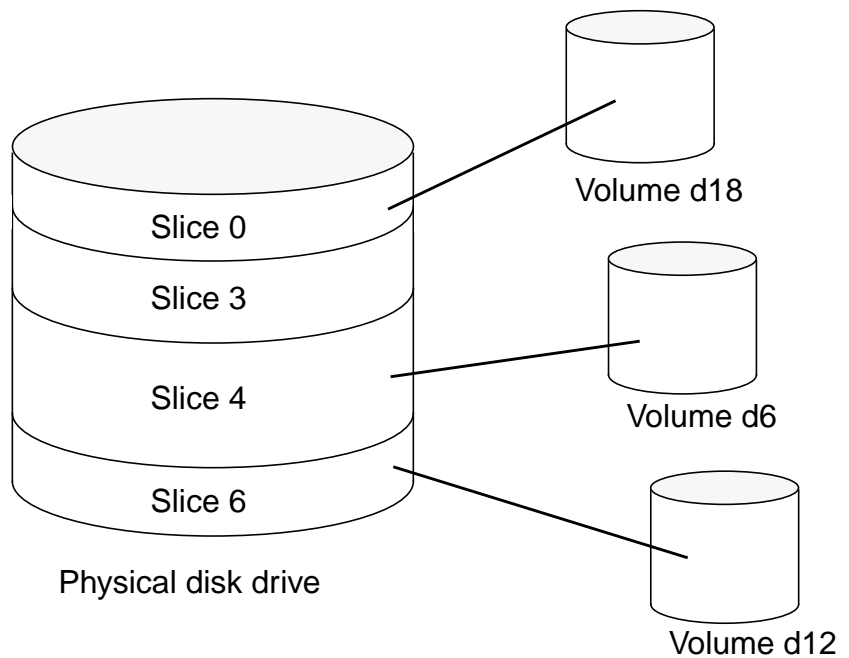


Figure 8-1 Solstice DiskSuite Space Management

Note – Reserve slice 7 on the system boot disk for Solstice DiskSuite state database storage.



Solstice DiskSuite Initialization

Disk drives that are to be used by Solstice DiskSuite do not need special initialization. Standard UNIX partitions are used without any modification.

Solstice DiskSuite needs several small databases in which to store volume configuration information along with some error and status information. These are called state databases and can be replicated on one or more disk drives as shown in Figure 8-2. Another common term for the state databases is replicas.

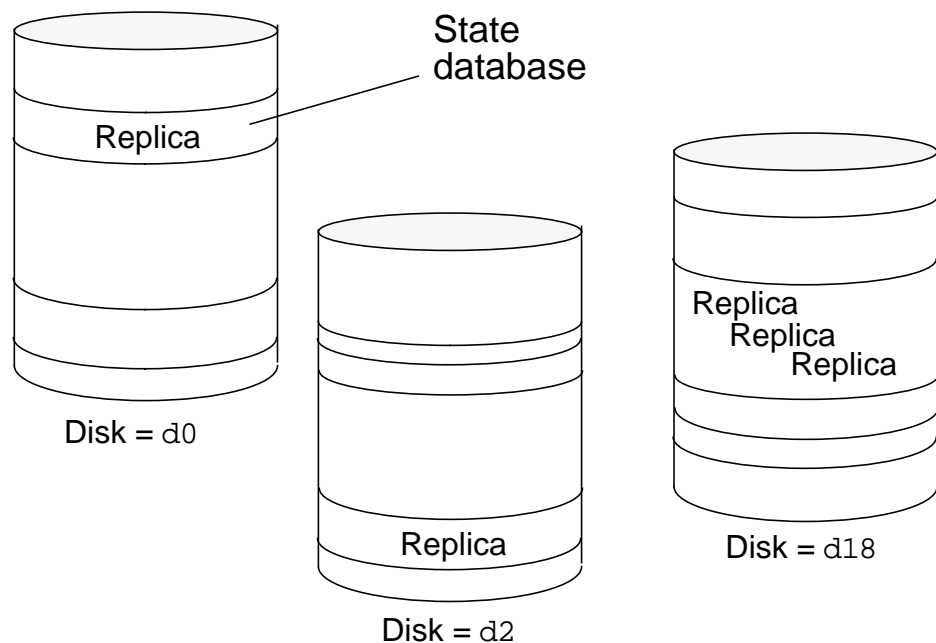


Figure 8-2 Solstice DiskSuite Replica Distribution

By default, Solstice DiskSuite requires a minimum of three copies of the state database.

The replicas are placed on standard unused partitions by a special command, `metadb`. The default size for each replica is 517 Kbytes (1034 disk blocks).

Note – You can create several replicas in a single partition.



Replica Configuration Guidelines

At least one replica is required to start the Solstice DiskSuite software. A minimum of three replicas is recommended. Solstice DiskSuite 4.2 allows a maximum of 50 replicas.

The following guidelines are recommended:

- For one drive – Put all three replicas in one slice
- For two to four drives – Put two replicas on each drive
- For five or more drives – Put one replica on each drive

Use your own judgement to gauge how many replicas are required (and how to best distribute the replicas) in your storage environment.



Note – You cannot store replicas on the `root`, `swap`, or `/usr` partitions, or on partitions containing existing file systems or data.

If multiple controllers exist on the system, replicas should be distributed as evenly as possible across all controllers. This provides redundancy in case a controller fails and also helps balance the load. If multiple disks exist on a controller, at least two of the disks on each controller should store a replica.

Do not place more than one replica on a single disk unless that is the only way to reach the minimum requirement of three replicas.

Solstice DiskSuite Disk Grouping

Disk groups are an arbitrary collection of physical disks that allow a backup host to assume a workload. The disk groups are given unique names and ownership is assigned to a single cluster host system.

Solstice DiskSuite uses the term *diskset* to define a related collection of disk drives.

A *shared diskset* is a grouping of two hosts and disk drives that are accessible by both hosts and have the same device names on both hosts. Each host can have *exclusive* access to a shared diskset; they cannot access the same diskset simultaneously.



Note – Hosts do not “share” the disk drives in a shared diskset. They can take turns having exclusive access to a shared diskset, but they cannot concurrently access the drives.

Disksets facilitate moving disks between host systems, and are an important piece in enabling high availability. Disksets also enable you to group storage by department or application (Figure 8-3).

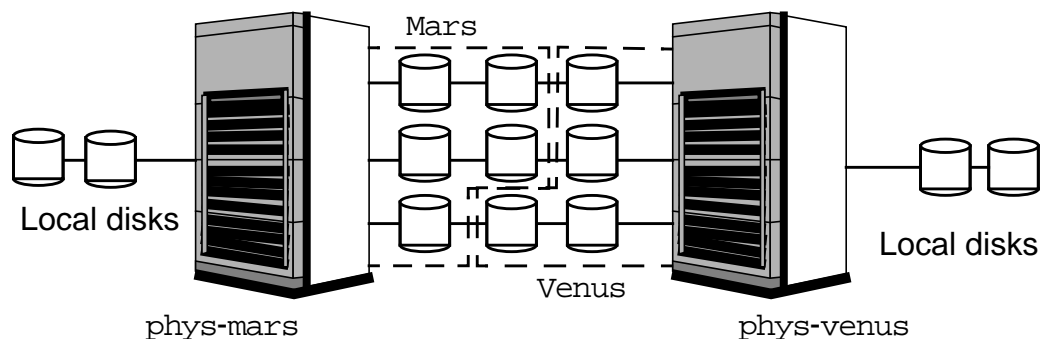


Figure 8-3 Shared Disksets

With disksets:

- Each host must have a *local diskset* that is separate from the shared diskset.
- There is one state database for each shared diskset and one state database for the local diskset.

Adding Disks to a Diskset

You use the `metaset` command to both create new empty disksets and to add disk drives into an existing diskset. In the following example, both functions are performed at the same time.

```
# metaset -s nfsds -a /dev/did/rdisk/d2 \  
/dev/did/rdisk/d5
```



Note – You should evenly distribute the disks in each diskset across at least two arrays to accommodate mirroring of data. Make sure to use the disk ID (DID) device name instead of the actual disk drive names.

When a disk is added to a diskset, it is automatically repartitioned as follows:

- A small portion of the drive (starting at cylinder 0) is placed into slice 7 to be used for metadvice state database replicas (usually at least 2 Mbytes)
- The rest of the drive is placed into slice 0

The drive is *not* repartitioned if slice 7 has the following characteristics:

- It starts at cylinder 0
- It has at least 2 Mbytes (large enough to hold a state database)
- It has the `V_UNMT` flag set (unmountable flag)
- It is not read-only

Dual-String Mediators

Solstice DiskSuite has two kinds of state databases. Initially, a local state database is replicated on each local boot disk. The local replicas are private to each host system.

When a shared diskset is created, a different set of replicas are created that are unique to the diskset. Each shared diskset has its own set of replicas.

If there is any inconsistency in the database replicas, Solstice DiskSuite uses what the majority (half + 1) of the database replicas contain. This is called a replica quorum. If a majority of the replica databases cannot be contacted, Solstice DiskSuite effectively shuts down.

When a diskset is configured in a dual-string (that is, two disk arrays) configuration, as shown in Figure 8-4, Solstice DiskSuite splits the number of replicas for each diskset evenly across both arrays.

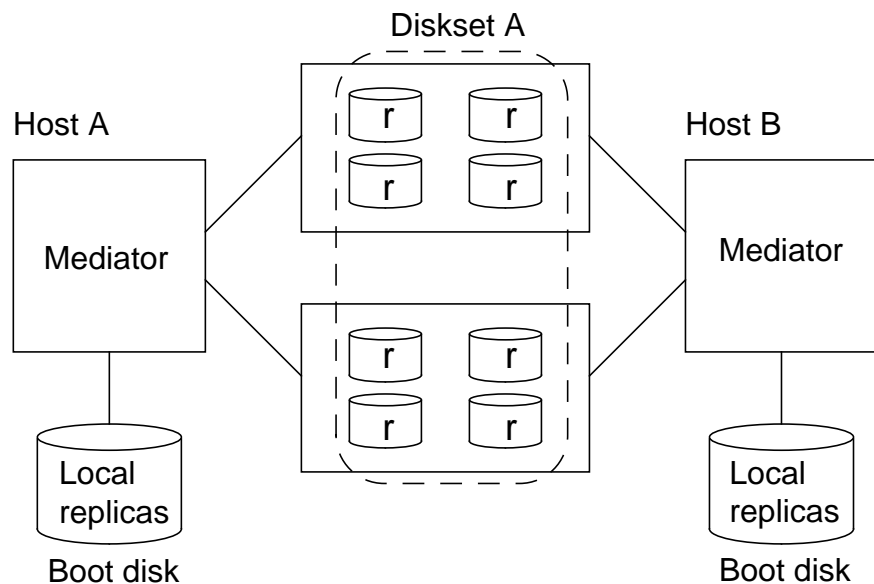


Figure 8-4 Diskset Replica Placement

If one of the disk arrays fails, exactly half of the disk replicas are available and a majority cannot be reached. If exactly half of the replicas are available, the information in the mediators (namely the state database commit count) allows Solstice DiskSuite to determine the integrity of the remaining replicas, effectively casting the tie-breaking vote, allowing Solstice DiskSuite to continue as if it had a replica quorum.

The Solaris Operating Environment UFS Logging

UFS logging is a standard feature of the Solaris 8 Operating Environment. All Sun Cluster global file systems are mounted using the `logging mount` option.

If the logging option is specified for a file system, then logging is enabled for the duration of the mounted file system. Logging is the process of storing transactions (changes that make up a complete UFS operation) in a log before the transactions are applied to the file system. After a transaction is stored, the transaction can be applied to the file system. This prevents file systems from becoming inconsistent, therefore eliminating the need to run `fsck`. And, because `fsck` can be bypassed, logging reduces the time required to reboot a system if it crashes or after an unclean halt. The default behavior is no logging.

The log is allocated from free blocks on the file system and is sized approximately 1 Mbyte per 1 Gbyte of file system, up to a maximum of 64 Mbytes. Logging can be enabled on any UFS, including `root (/)`. The log created by UFS logging is continually flushed as it fills up. The log is totally flushed when the file system is unmounted or as a result of the `lockfs -f` command.

Solstice DiskSuite Status

Although the graphical user interface (GUI) for Solstice DiskSuite furnishes useful visual status information, there are times when the images might not update completely due to window interlocks or system loads.

The most reliable and the quickest method of checking status is from the command line. Command-line status tools have the additional benefits of being easy to use in script files, cron jobs, and remote logins.

Checking Volume Status

The following `metastat` command output, is for a mirrored metadevice, `d0`, and is used with the Solstice DiskSuite volume manager.

```
# metastat d0
d0: Mirror
Submirror 0: d80
State: Okay
Submirror 1: d70
State: Resyncing  Resyncin progress: 15% done
Pass: 1
Read option: roundrobin (default)
Write option: parallel (default)
Size: 2006130 blocks
```



Note – You can also use the `metastat` command to create a backup configuration file that is suitable for recreating the entire volume structure. This is useful as a worst-case disaster-recovery tool.

Checking Mediator Status

You use the `medstat` command to verify the status of mediators in a dualstring storage configuration.

```
# medstat -s nfsds
```

Mediator	Status	Golden
pnode1	Ok	No
pnode2	Ok	No

Checking Replica Status

The status of the state databases is important, and you can verify it using the `metadb` command, as shown in the following example. You can also use the `metadb` command to initialize, add, and remove replicas.

```
# metadb
flags      first blk  block count
a u        16        1034
/dev/dsk/c0t3d0s5
a u        16        1034
/dev/dsk/c0t2d0s0
a u        16        1034
/dev/dsk/c0t2d0s1
o - replica active prior to last configuration change
u - replica is up to date
l - locator for this replica was read successfully
c - replica's location was in /etc/opt/SUNWmd/mddb.cf
p - replica's location was patched in kernel
m - replica is master, this is replica selected as
input
W - replica has device write errors
a - replica is active, commits are occurring
M - replica had problem with master blocks
D - replica had problem with data blocks
F - replica had format problems
S - replica is too small to hold current data base
R - replica had device read errors
```

The status flags for the replicas shown in the previous example indicate that all of the replicas are active and are up to date.

Recording Solstice DiskSuite Configuration Information

Archive diskset configuration information using the `metastat -p` command option. The configuration information is output in a format that you can later use to automatically rebuild your diskset volumes.

```
# metastat -s nfsds -p
nfsds/d100 -m nfsds/d0 nfsds/d10 1
nfsds/d0 1 1 /dev/did/dsk/d10s0
nfsds/d10 1 1 /dev/did/dsk/d28s0
nfsds/d101 -m nfsds/d1 nfsds/d11 1
nfsds/d1 1 1 /dev/did/dsk/d10s1
nfsds/d11 1 1 /dev/did/dsk/d28s1
nfsds/d102 -m nfsds/d2 nfsds/d12 1
nfsds/d2 1 1 /dev/did/dsk/d10s3
nfsds/d12 1 1 /dev/did/dsk/d28s3
```

Solstice DiskSuite Installation Overview

Installing and configuring Solstice DiskSuite for use in a SunTM Cluster 3.0 07/01 environment consists of the following steps:

1. Install the appropriate packages for Solstice DiskSuite.
2. Modify the `md.conf` file appropriately.
3. Reboot all nodes in the cluster.
4. Create `.rhosts` files, or add `root` to group 14.
5. Initialize the local state databases.
6. Create disksets to house data service data.
7. Add drives to each diskset.
8. Partition disks in the disksets.
9. Create the metadevices for each diskset.
10. Configure dual string mediators, if appropriate.

Solstice DiskSuite Postinstallation

After you install the Solstice DiskSuite software, you must make several postinstallation configuration modifications before Solstice DiskSuite can operate.

Modifying the `md.conf` File

Based on your planned implementation, you might need to update DiskSuite's kernel configuration file: `/kernel/drv/md.conf`. There are two variables that might need to be updated. The modifications are summarized in Table 8-1:

Table 8-1 Modification to the `md.conf` file

Variable	Default Value	Description
<code>nmd</code>	128	The maximum number of metadevices. Solstice DiskSuite uses this setting to limit the <i>names</i> of the metadevices as well. If you are going to have 100 metadevices, but you want to name them <code>d1000</code> through <code>d1100</code> , set this value to 1101, not 100. The maximum value for <code>nmd</code> is 8192.
<code>md_nsets</code>	4	The maximum number of disksets. Set this number should be set to the number of disksets you plan to create in your cluster (probably equal to the number of logical hosts you plan to have assuming one diskset per logical host). The maximum value for <code>md_nsets</code> is 32.



Note – Keep this file identical on all nodes of the cluster. Changes to this file take effect after you perform a reconfiguration reboot.

Enabling Solstice DiskSuite Node Access

Solstice DiskSuite requires root level access on each cluster node. There are two ways to satisfy this requirement:

- Create `.rhosts` files listing the names of all the cluster nodes on each node of the cluster
- Add root to the `sysadmin` group on all nodes of the cluster

Initializing Local State Database Replicas

Before you can perform any Solstice DiskSuite configuration tasks, such as creating disksets on the multihost disks or mirroring the root (`/`) file system, you must create the metadb state database replicas on the local (private) disks on each cluster node. The local disks are separate from the multihost disks. The state databases located on the local disks are necessary for basic Solstice DiskSuite operation.

A typical command to place three state database replicas on slice 7 of a system boot disk is as follows:

```
# metadb -a -c 3 -f c0t0d0s7
```

Creating Disksets for the Data Services

On one of the cluster nodes, you create the empty disksets that you need to support the data services. You must specify the names to two hosts that will access the diskset. An example follows:

```
# metaset -s nfsds -a -h pnode1 pnode1
```

Adding Disks to a Diskset

Add disks from different storage arrays to enable mirroring across arrays to maintain high availability. An example follows:

```
# metaset -s nfsds -a /dev/did/rdisk/d7 \  
/dev/did/rdisk/d15
```

Configuring Metadevices

Solstice DiskSuite metadevices can be configured in a file, `/etc/lvm/md.tab`, and automatically created by running the `metainit` command against the file.

Metadevices can also be created from the command line in a series of steps. The command sequence to create a mirrored metadevice (`d100`) in the diskset `nfds` using two DID disk `d7` and `d15` is as follows:

1. Create a submirror, `d0`, on slice 0 of DID device `d7`, in the diskset named `nfds`.

```
# metainit -s nfds nfds/d0 1 1 /dev/did/rdisk/d7s0
```
2. Create another submirror, `d10`, on slice 0 of DID device `d15`, in the diskset named `nfds`.

```
# metainit -s nfds nfds/d10 1 1 /dev/did/rdisk/d15s0
```
3. Create a metadevice, `d100`, and add the `d0` submirror to it.

```
# metainit -s nfds nfds/d100 -m nfds/d0
```
4. Attach the second submirror, `d10`, to the metadevice `d100`.

```
# metattach -s nfds nfds/d100 nfds/d10
```
5. Verify the status of the metadevice `d100`.

```
# metastat d100
d100: Mirror
Submirror 0: d0
State: Okay
Submirror 1: d10
State: Resyncing  Resync in progress: 15% done
Pass: 1
Read option: roundrobin (default)
Write option: parallel (default)
Size: 2006130 blocks
```

Note – Remember that the size of the volumes are defined by the size of the DID device partitions.



Configuring Dual-String Mediators

If you created any disksets using exactly two disk arrays (which are connected to two cluster nodes), you must configure dual-string mediators. The following rules apply when configuring dual-string mediators:

- You must configure disksets using dual strings and two hosts with exactly two mediator hosts, and these hosts must be the same two hosts used for the diskset.
- A diskset cannot have more than two mediator hosts.
- You cannot configure mediators for disksets that do not meet the two-string, two-host criteria.



Note – Mediators are not only for use in two-node clusters. Clusters having more than two nodes can also benefit from the use of mediators, depending on the topology and how the disksets are constructed.

The process to configure dual-string mediators is as follows:

1. Start the cluster software on both nodes.
2. Determine the host name of both mediator hosts (nodes).
3. Use the `hastat` command to determine the current master of the diskset you are configuring for mediators.
4. Configure the mediators using the `metaset` command on the host that is currently mastering the diskset.

```
# metaset -s nfsds -a -m pnode1
# metaset -s nfsds -a -m pnode2
```

5. Check the mediator status using the `medstat` command.

```
# medstat -s nfsds
```

Exercise: Configuring Solstice DiskSuite

In this exercise, you complete the following tasks:

- Install the Solstice DiskSuite volume manager
- Initialize the Solstice DiskSuite volume manager
- Create and manage disksets
- Create dual-string mediators if appropriate
- Create global file systems

Preparation

Ask your instructor about the location of the software that is needed during this exercise. Record the location of the Solstice DiskSuite software.

Solstice DiskSuite location: _____

Each of the cluster host boot disks must have a small unused slice that you can use for a state database during the Solstice DiskSuite installation.

During this exercise you create two data service disksets that each contain a single mirrored volume as shown in Figure 8-5.

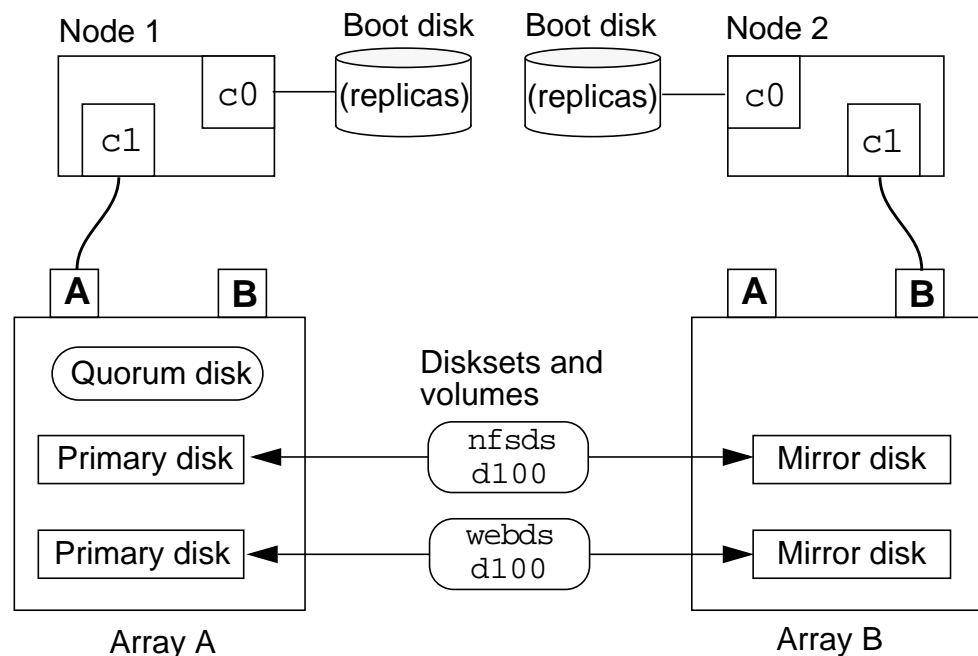


Figure 8-5 Configuring Solstice DiskSuite



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Installing the Solstice DiskSuite Software

To install the Solstice DiskSuite software, perform the following steps:

1. On both nodes, move to the location of the Solstice DiskSuite software.
2. If you are in the correct location you should see the following files:

```
# ls
SUNWmdg  SUNWmdja  SUNWmdnr  SUNWmdnu  SUNWmdr
SUNWmdu  SUNWmdx
```

3. Run the `pkgadd` command on both cluster host systems to begin the volume manager installation.

```
# pkgadd -d .
The following packages are available:
1  SUNWmdg      Solstice DiskSuite Tool
2  SUNWmdja     Japanese localization
3  SUNWmdnr     Log Daemon Configuration Files
4  SUNWmdnu     Log Daemon
5  SUNWmdr      Solstice DiskSuite Drivers
6  SUNWmdu      Solstice DiskSuite Commands
7  SUNWmdx      Solstice DiskSuite Drivers(64-bit)
```

```
Select package(s) you wish to process (or 'all' to
process all packages). (default: all) [?,?,q]:
```



Note – Install only the `SUNWmdg`, `SUNWmdr`, `SUNWmdu`, and `SUNWmdx` packages (1,5,6,7).

4. Install any current Solstice DiskSuite patches on all cluster host systems.
5. Stop the Sun Cluster software on all nodes.
6. Reboot all of the cluster hosts after you install the Solstice DiskSuite patch.

Task – Initializing the Solstice DiskSuite State Databases

Before you can use Solstice DiskSuite to create disksets and volumes, you must initialize the state database and create one or more replicas.

Configure the system boot disk on each cluster host with a small unused partition. This should be slice 7.

Perform the following steps:

1. On each node in the cluster, verify that the boot disk has a small unused slice available for use. Use the `format` command to verify the physical path to the unused slice. Record the paths of the unused slice on each cluster host. A typical path is `c0t0d0s7`.

Node 1 Replica Slice: _____

Node 2 Replica Slice: _____

Warning – You must ensure that you are using the correct slice. A mistake can corrupt the system boot disk. Check with your instructor.



2. On each node in the cluster, use the `metadb` command to create three replicas on the unused boot disk slice.

```
# metadb -a -c 3 -f replica_slice
```

Warning – Make sure you reference the correct slice address on each node. You can destroy your boot disk if you make a mistake.



3. On both nodes, verify that the replicas are configured and operational.

```
# metadb
flags      first blk    block count
a  u       16        1034      /dev/dsk/c0t0d0s7
a  u       1050      1034      /dev/dsk/c0t0d0s7
a  u       2084      1034      /dev/dsk/c0t0d0s7
```

Task – Selecting the Solstice DiskSuite Demo Volume Disk Drives

Perform the following steps to select the Solstice DiskSuite Demo Volume disk drives:

1. On Node 1, type the `scdidadm -L` command to list all of the available DID drives.
2. Record the logical path and DID path numbers of four disks that you will use to create the demonstration disk sets and volumes in Table 8-2. Remember to mirror across arrays.



Note – You need to record only the last portion of the DID path. The first part is the same for all DID devices: `/dev/did/rdisk`.

Table 8-2 Logical Path and DID Numbers

Diskset	Volumes	Primary disk	Mirror disk
<i>example</i>	<i>d400</i>	<i>c2t3d0 d4</i>	<i>c3t18d0 d15</i>
nfsds	d100		
webds	d100		



Caution – Make sure the disks you select are *not* local devices. They must be dualhosted and available to both cluster hosts.

Task – Configuring the Solstice DiskSuite Demonstration Disksets

Perform the following steps to create demonstration disksets and volumes for use in later exercises:

1. On Node 1, create the `nfsds` diskset, and configure the nodes that are physically connected to it.

```
# metaset -s nfsds -a -h node1 node2
```
2. On Node 1, create the `webds` diskset and configure the nodes that are physically connected to it.

```
# metaset -s webds -a -h node1 node2
```
3. Add the primary and mirror disks to the `nfsds` diskset.

```
# metaset -s nfsds -a /dev/did/rdisk/primary \  
/dev/did/rdisk/mirror
```
4. Add the primary and mirror disks to the `webds` diskset.

```
# metaset -s webds -a /dev/did/rdisk/primary \  
/dev/did/rdisk/mirror
```
5. On Node 1, start the `format` utility and repartition each of your diskset disks and reduce the size of slice 0 to approximately 500 Mbytes. The slices must be *identical* in size on each primary and mirror pair. This is to limit the size of the finished volumes.

Note – Make sure the first few cylinders are not already mapped to slice 7 for the local state databases.



6. Verify the status of the new disksets.

```
# metaset -s nfsds  
# metaset -s webds
```

Task – Configuring Solstice DiskSuite Demonstration Volumes

Perform the following steps on Node 1 to create a 500-Mbyte mirrored volume in each diskset:

Diskset `nflds`

1. Create a submirror on each of your disks in the `nflds` diskset.

```
# metainit -s nflds nflds/d0 1 1 /dev/did/rdisk/primarys0
# metainit -s nflds nflds/d1 1 1 /dev/did/rdisk/mirrors0
```

2. Create a metadevice, `d100`, and add the `d0` submirror to it.

```
# metainit -s nflds nflds/d100 -m nflds/d0
```

3. Attach the second submirror, `d1`, to the metadevice `d100`.

```
# metattach -s nflds nflds/d100 nflds/d1
```

Diskset `webds`

1. Create a submirror on each of your disks in the `webds` diskset.

```
# metainit -s webds webds/d0 1 1 /dev/did/rdisk/primarys0
# metainit -s webds webds/d1 1 1 /dev/did/rdisk/mirrors0
```

2. Create a metadevice, `d100`, and add the `d0` submirror to it.

```
# metainit -s webds webds/d100 -m webds/d0
```

3. Attach the second submirror, `d1`, to the metadevice `d100`.

```
# metattach -s webds webds/d100 webds/d1
```

4. Verify the status of the new volumes.

```
# metastat
```

Task – Configuring Dual-String Mediators

If your cluster is a dual-string configuration, you must configure mediation for both of the disksets you have created. Perform the following steps:

1. Make sure the cluster software is running on the cluster hosts.
2. Use the `metaset` command to determine the current master of the disksets you are configuring for mediators.

Note – Both disksets should be mastered by Node 1.



3. Configure the mediators using the `metaset` command on the host that is currently mastering the diskset.

```
# metaset -s nfsds -a -m node1
# metaset -s nfsds -a -m node2
#
# metaset -s webds -a -m node1
# metaset -s webds -a -m node2
```

4. Check the mediator status using the `medstat` command.

```
# medstat -s nfsds
# medstat -s webds
```

Task – Creating a Global `nfs` File System

Perform the following steps on Node 1 to create a global file system in the `nfsds` diskset:

1. On Node 1, create a file system on `d100` in the `nfsds` diskset.

```
# newfs /dev/md/nfsds/rdisk/d100
```

2. On *both* Node 1 and Node 2, create a global mount point for the new file system.

```
# mkdir /global/nfs
```

3. On *both nodes*, add a mount entry in the `/etc/vfstab` file for the new file system with the `global` and `logging` mount options.

```
/dev/md/nfsds/dsk/d100 /dev/md/nfsds/rdisk/d100 \  
/global/nfs ufs 2 yes global,logging
```

Note – Do not use the line continuation character (`\`) in the `vfstab` file.

4. On Node 1, mount the `/global/nfs` file system.

```
# mount /global/nfs
```

5. Verify that the file system is mounted and available on *both* nodes.

```
# mount  
# ls /global/nfs  
lost+found
```



Task – Creating a Global web File System

Perform the following steps on Node 1 to create a global file system in the webds diskset:

1. On Node 1, create a file system on d100 in the webds diskset.

```
# newfs /dev/md/webds/rdisk/d100
```

2. On *both* Node 1 and Node 2, create a global mount point for the new file system.

```
# mkdir /global/web
```

3. On *both nodes*, add a mount entry in the /etc/vfstab file for the new file system with the global and logging mount options.

```
/dev/md/webds/dsk/d100 /dev/md/webds/rdisk/d100 \  
/global/nfs ufs 2 yes global,logging
```

Note – Do not use the line continuation character (\) in the vfstab file.



4. On Node 1, mount the /global/web file system.

```
# mount /global/web
```

5. Verify that the file system is mounted and available on *both* nodes.

```
# mount  
# ls /global/web  
lost+found
```

Task – Testing Global File Systems

Perform the following steps to confirm the general behavior of globally available file systems in the Sun™ Cluster 3.0 07/01 environment:

1. On Node 2, move into the `/global/nfs` file system.

```
# cd /global/nfs
```
2. On Node 1, try to unmount the `/global/nfs` file system. You should get an error that the file system is busy.
3. On Node 2, move out of the `/global/nfs` file system (`cd /`) and try to unmount it again on Node 1.
4. Mount the `/global/nfs` file system on Node 1.
5. Try unmounting and mounting `/global/nfs` from both nodes.

Task – Managing Disk Device Groups

Perform the following steps to migrate a disk device group (diskset) between cluster nodes:

1. Make sure the *device groups* are online (to Sun Cluster).

```
# scstat -D
```

Note – You can bring a device group online to a selected node as follows:

```
# scswitch -z -D nfsds -h node1
```



2. Verify the current demonstration device group configuration.

```
pnodel# scconf -p |grep group
Device group name:                webds
Device group type:                SDS
Device group failback enabled:    no
Device group node list:          pnodel, pnodel2
Device group ordered node list:   yes
Diskset name:                    webds
Device group name:                nfsds
Device group type:                SDS
Device group failback enabled:    no
Device group node list:          pnodel, pnodel2
Device group ordered node list:   yes
Diskset name:                    nfsds
```

3. Shut down Node 1. The *nfsds* and *webds* disksets should automatically migrate to Node 2 (verify with the `scstat -D` command).

Note – The migration is initiated by Node 1 during shutdown when you see the message: `/etc/rc0.d/K05initrgm: Calling scswitch -S (evacuate).`



4. Boot Node 1. The both disksets should remain mastered by Node 2.
5. Use the `scswitch` command from either node to migrate the *nfsds* diskset to Node 1.

```
# scswitch -z -D nfsds -h node1
```

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Explain the disk space management technique used by Solstice DiskSuite
- ☐ Describe the Solstice DiskSuite initialization process
- ☐ Describe how Solstice DiskSuite groups disk drives
- ☐ Use Solstice DiskSuite status commands
- ☐ Describe the Solstice DiskSuite software installation process
- ☐ Install and initialize Solstice DiskSuite
- ☐ Perform Solstice DiskSuite postinstallation configuration
- ☐ Create global file systems
- ☐ Perform basic device group management

Think Beyond

Where does Solstice DiskSuite fit into the high-availability environment?

What planning issues are required for Solstice DiskSuite in the high-availability environment?

Is use of the Solstice DiskSuite required for high-availability functionality?

Public Network Management

Objectives

Upon completion of this module, you should be able to:

- List the main features of the Sun Cluster Public Network Management (PNM) software
- Explain the basic PNM fault monitoring mechanism
- Describe the three network adapter failover (NAFO) group states
- List NAFO group requirements
- Configure a NAFO group

Relevance

Present the following question to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answer to this question, the answer should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following question is relevant to understanding the content of this module:

- What happens if a fully functional cluster node loses its network interface to a public network?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Public Network Management

The PNM software is a Sun Cluster package that provides the mechanism for monitoring public network adapters and failing-over Internal Protocol (IP) addresses from one adapter to another when a fault is detected.

Network adapters are organized into NAFO groups. IP addresses are assigned to a NAFO group using regular name service procedures. If a NAFO group network adapter fails, its associated IP address is transferred to the next backup adapter in the group.

Only one adapter in a NAFO group can be active (plumbed) at any time.

Each NAFO backup group on a cluster host is given a unique name, such as `nafo12`, during creation. A NAFO group can consist of any number of network adapter interfaces but usually contains only a few.

As shown in Figure 9-1, the PNM daemon (`pnmd`) monitors designated network adapters on a single node. If a failure is detected, `pnmd` uses information in the cluster configuration respository (CCR) and the `pnmconfig` file to initiate a failover to a healthy adapter in the backup group.

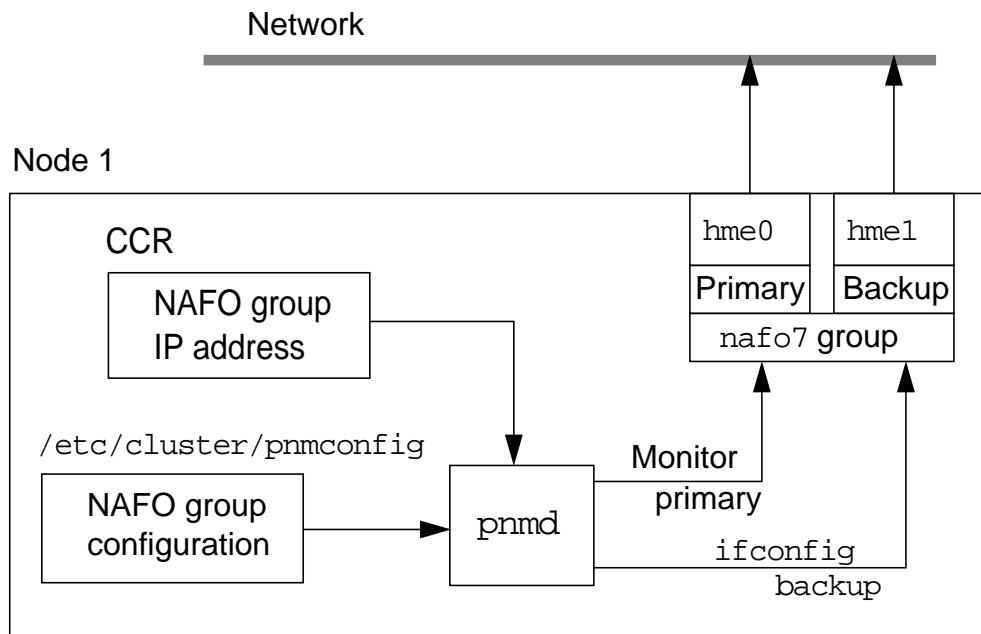


Figure 9-1 PNM Components

Supported Public Network Interface Types

PNM supports the following public network interfaces:

- SBus and peripheral component interface (PCI) bus 100-Mbit/second Ethernet
- Quad Ethernet cards (100 Mbits/second)



Note – Check current Sun Cluster release notes for Gigabit Ethernet support.

Global Interface Support

Sun™ Cluster 3.0 07/01 also provides a global interface feature. A global interface is a single network interface for incoming requests from all clients. NAFO groups that will be used for global interfaces configure with redundant adapters.



Note – The global interface feature is only used by the scalable data services.

Configuring NAFO Groups

Use the `pnmset` command to create and modify network adapter backup groups. You can create several NAFO groups at the same time or one at a time.

Only `root` can run the `pnmset` command.

Pre-Configuration Requirements

Before you use the `pnmset` command to create a new NAFO group, you must make sure `pnmset` can resolve the IP address and logical host name that will be associated with the NAFO group. You must perform the following steps:

1. Set the electrically crashable programmable read-only memory (EEPROM) parameter `local-mac-address?` to `false` on all nodes to prevent problems between PNM and the Solaris Operating Environment Network Interface Group feature.
2. Ensure that all adapters in a NAFO group are of the same media type.
3. Although it is not required, there should be an `/etc/hostname.xxx` file for the primary (first) interface in each NAFO group. The file should contain a unique host name to be associated with that NAFO group.
 - a. The host name must be resolved either in the local `/etc/hosts` file or through a naming service.
 - b. There should also be an empty `/etc/notrouter` file to prevent the cluster node from operating as a router.

If there is *not* a `/etc/hostname.xxx` file assigned to a NAFO group, `pnmset` uses the cluster node's primary IP address and host name to test and operate the NAFO group interfaces. The primary Ethernet interface for the node is taken down for a short time while `pnmset` performs tests on the proposed NAFO group interfaces.

Note – A network adapter cannot belong to more than one NAFO group.



Configuring Backup Groups

The `pnmset` program prompts you for the following information:

- The number of PNM backup groups you want to configure
- The backup group number

The group number is arbitrary. The total number of groups cannot exceed 256. If the group already exists, its configuration is overwritten with new information

- A list of network adapters to be configured in the group

The backup group should contain a minimum of two interfaces. If reconfiguring an existing group, you can add more interfaces.

Sample NAFO Group Configuration

The following is a transcript of the creation of a single NAFO backup group.

```
# pnmset
```

```
In the following, you will be prompted to do configuration
for network adapter failover
```

```
Do you want to continue ... [y/n]: y
```

```
How many NAFO groups to configure [1]: 1
```

```
Enter NAFO group number [0]: 2
```

```
Enter space-separated list of adapters in nafo2: qfe0 qfe1
```

```
Checking configuration of nafo2:
```

```
Testing active adapter qfe0...
```

```
Testing adapter qfe1...
```

```
NAFO configuration completed
```

```
# pnmset -p
```

```
current configuration is:
```

```
nafo2 qfe0 qfe1
```

Modifying Existing PNM Configurations

After PNM is initially configured, you can use `pnmset` with special options to add new NAFO groups or update existing NAFO groups.

Adding an Adapter to an Existing NAFO Group

The following is an example of using the `pnmset` command to add another adapter to an existing NAFO group:

```
# pnmset -c nafo2 -o add qfe2
```

Creating an Additional NAFO Group

After you have created an initial NAFO group on a node, you must use the following form of the `pnmset` command to create additional NAFO groups.

```
# pnmset -c nafo12 -o create qfe2 qfe3
```



Note – You should first create a `/etc/hostname.xxx` file for the new NAFO group with a new host name. You must also make changes to resolve the new IP address for the logical host.

Consult the `pnmset` manual page for additional information about adding and modifying NAFO backup groups.

PNM Status Commands

There are several status commands that are useful for checking general NAFO group status and identifying active adapters.

The `pnmstat` Command

The following shows how to use the `pnmstat` command to check the status of all local backup groups:

```
# pnmstat -l
group    adapters      status  fo_time  act_adp
nafol    qfe0:qfe1        OK      NEVER    qfe0
```

Note – The NAFO group shown has never experienced a problem, so the failover time (`fo_time`) shows NEVER.



The `pnmptor` Command

The following shows how to use the `pnmptor` command to identify which adapter is active in a given backup group:

```
# pnmptor nafol
qfe0
```

The `pnmrtop` Command

The following shows how to use the `pnmrtop` command to determine which backup group contains a given adapter:

```
# pnmrtop qe0
nafol
```

The PNM Monitoring Process

The PNM daemon is based on a remote procedure call (RPC) client-server model. It is started at boot time in an `/etc/rc3.d` script and killed in `/etc/rc0.d`.

PNM uses the CCR for storing state information for the results of the adapter monitoring test results on the various hosts. Data services can query the status of the remote adapters at any time using the data service application programming interface (API) framework.

When monitoring for network faults, the PNM software must determine where the failure is before taking action. The fault could be a general network failure and not a local adapter. PNM can use the cluster private transport interface to find out if other nodes are also experiencing network access problems. If other nodes (peers) see the problem, then the problem is probably a general network failure, and there is no need to take action.

If the detected fault is determined to be the fault of the local adapter, notify the network failover component to begin an adapter failover, which is transparent to the highly available data services. A backup network adapter is activated to replace the failed network adapter, and the associated IP address is configured on the new interface. This avoids having to move the entire server workload to another server because of the loss of a single network adapter.

If there are no more operational adapters in the NAFO group for a data service, the Sun Cluster API framework then uses NAFO status information to determine:

- Whether to migrate the data service
- Where to migrate the data service

PNM Monitoring Routines

PNM uses three general routines to manage NAFO groups. The routines perform the following functions:

- Monitor active NAFO interfaces
- Evaluate a suspected failure
- Fail over to a new adapter or host

NAFO Group Status

During testing and evaluation, a NAFO group status can transition through several states depending on the results of evaluation and failover. The possible states and associated actions are described in the following sections.

OK State

After monitoring and testing, the current adapter appears to be active. No action is taken, so monitoring continues.

DOUBT State

The current adapter appears to be inactive. Perform further testing and switch to a backup adapter, if necessary. Adapters in the NAFO group are tested sequentially to find an active adapter to replace the currently failed interface.

If a backup adapter is found, the Internet Protocol (IP) address is configured on the new adapter, and the NAFO group status returns to the OK state.

DOWN State

There are no active adapters in the NAFO group. If appropriate, take action to move the IP address and associated resources to a different cluster node.

PNM Parameters

During testing and evaluation of NAFO groups, the PNM monitoring process uses the `ping` command in increasingly invasive levels as follows:

- First `ping` ALLROUTERS multicast (224.0.0.2)
- Then `ping` ALLHOSTS multicast (224.0.0.1)
- Finally, try broadcast `ping` (255.255.255.255)

Tests are retried a number of times, and there are time-out values for responses. You can set several configurable PNM parameters in the `/etc/cluster/pnmparams` file. This file must be created manually, and entries in it override default PNM values. You can modify the parameters to help ensure that unneeded adapter or data service failovers are not performed because of temporary network performance problems. The parameters are shown in Table 9-1.

Table 9-1 Public Network Management Tunable Parameters

Parameter	Description
<code>inactive_time</code>	The number of seconds between successive probes of the packet counters of the current active adapter. The default is 5.
<code>ping_timeout</code>	The time-out value in seconds for the <code>ALL_HOST_MULTICAST</code> and subnet broadcast <code>ping</code> commands. The default is 4.
<code>repeat_test</code>	The number of times to perform the <code>ping</code> sequence before declaring that the active adapter is faulty and failover is triggered. The default is 3.
<code>slow_network</code>	The number of seconds waited after each <code>ping</code> sequence before checking packet counters for any change. The default is 2.
<code>warmup_time</code>	The number of seconds to wait after failover to a backup adapter before resuming fault monitoring. This allows extra time for any slow driver or port initialization. The default is 0. The <code>qfe</code> interface can take 15 seconds or more to initialize with certain switches. Extra warmup time might be needed during initial <code>pnmd</code> startup or during failovers to another host.

Exercise: Configuring the NAFO Groups

In this exercise, you create a NAFO group on each cluster host system

Preparation

Ask your instructor for help with defining the NAFO groups that will be used on your assigned cluster system.

You create a single NAFO group on each cluster host that contains the name of the primary network adapter for the host.

Ask your instructor for help with selecting the IP address and adapters for use during this exercise. Record them in Table 9-2.

Table 9-2 IP Address and Adapters

Node	Logical Host Name (Optional)	IP Address (Optional)	NAFO Group Number	Adapters
Node 1			1	
Node 2			2	

Although it is not mandatory, it is a good practice in a business environment to create a `/etc/hostname.xxx` file and assign a unique host name for the primary interface in each NAFO group.



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Verifying EEPROM Status

Perform the following steps to verify the EEPROM `local-mac-address?` variable is set to `false` on both nodes:

1. Type the `eeeprom` command on each node, and verify the setting of the `local-mac-address?` variable.

```
# eeeprom | grep mac
local-mac-address?=false
```

2. If necessary, change the `local-mac-address?` value, and reboot the node.

```
# eeeprom local-mac-address?=false
# init 6
```



Note – When the nodes are running fully configured data services, you should not simply shut them down. You should first identify data services running on the node and use the `scswitch` command to migrate them to a backup node.

Task – Creating a NAFO Group

Perform the following steps on each node to create a new NAFO group:

1. Verify that NAFO groups do not exist on each cluster host.

```
# pnmstat -l
```

2. On Node 1, create a single NAFO group, numbered 1.

```
# pnmset
```



Note – If at all possible, each group should consist of two interfaces, one of which can be the primary node interface (`hme0`).

3. On Node 2, create a single NAFO group, numbered 2.
4. Verify that the status of each new NAFO group is OK on all nodes.

```
# pnmstat -l
```

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ List the main features of the Sun Cluster PNM software
- ☐ Explain the basic PNM fault monitoring mechanism
- ☐ Describe the three NAFO group states
- ☐ List NAFO group requirements
- ☐ Configure a NAFO group

Think Beyond

Are there other system components that would benefit from the approach taken to network adapters by PNM?

What are the advantages and disadvantages of automatic adapter failover?

Resource Groups

Objectives

Upon completion of this module, you should be able to:

- Describe the primary purpose of resource groups
- List the components of a resource group
- Describe the resource group configuration process
- List the primary functions of the `scrgadm` command
- Explain the difference between standard and extended resource type properties
- List the `scsetup` utility resource group functions

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- What is the purpose of a resource group?
- What needs to be defined for a resource group?
- What are the restrictions on resource groups?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *Sun™ Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *Sun™ Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *Sun™ Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *Sun™ Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *Sun™ Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *Sun™ Cluster 3.0 07/01 Concepts*, part number 806-7074
- *Sun™ Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *Sun™ Cluster 3.0 07/01 Release Notes*, part number 806-7078

Resource Group Manager

The Resource Group Manager (RGM) controls data services (applications) as *resources*, which are managed by *resource type* implementations (Figure 10-1). These implementations are either supplied by Sun or created by a developer with a generic data service template, the Data Service Development Library application programming interface (API), or the Sun™ Cluster 3.0 07/01 Resource Management API. The cluster administrator creates and manages resources in containers called *resource groups*, which form the basic unit of failover and switchover. The RGM stops and starts resource groups on selected nodes in response to cluster membership changes.

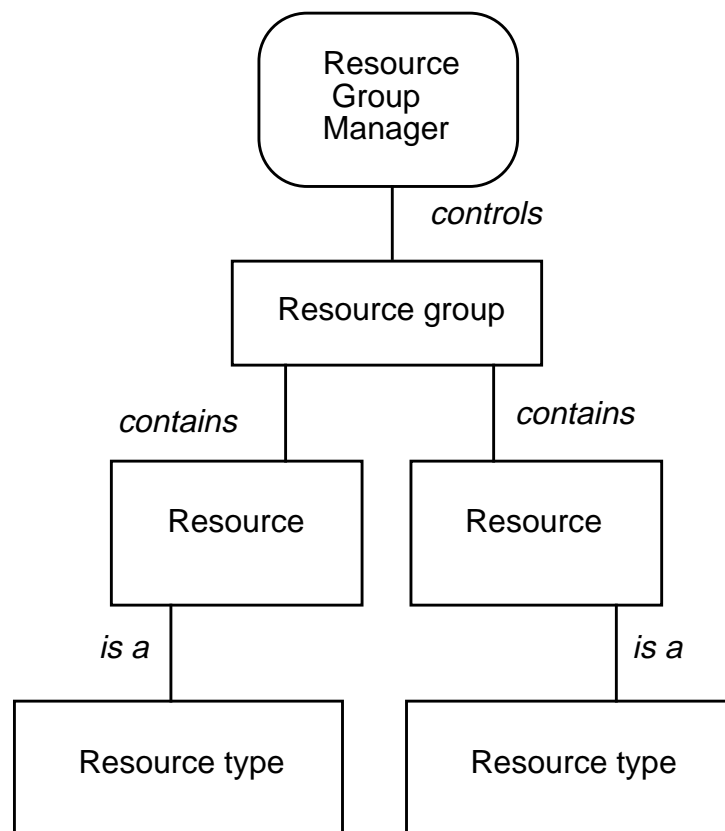


Figure 10-1 Resource Group Management

Resource Types

When you configure a data service and resource groups, you must furnish information about resource types. There are two resource types:

- Data service resource types
- Preregistered resource types (used by most data services)

Data Service Resource Types

Table 10-1 lists the resource types associated with each Sun™ Cluster 3.0 07/01 data service.

Table 10-1 Data Service Resource Types

Data Service	Resource Type
Sun Cluster HA for Oracle	SUNW.oracle_server SUNW.oracle_listener
Sun Cluster HA for iPlanet	SUNW.iws
Sun Cluster HA for Netscape Directory Server	SUNW.nsldap
Sun Cluster HA for Apache	SUNW.apache
Sun Cluster HA for Domain Name Service (DNS)	SUNW.dns
Sun Cluster HA for NFS	SUNW.nfs
Sun Cluster HA for SAP	SUNW.sap_ci SUNW.sap_as
Sun Cluster HA for Sybase	SUNW.sybase



Note – You must register a data service resource type once from any cluster node after the data service software is installed.

Preregistered Resource Types

The following resource types are common to many data services.

- SUNW.HAStorage
- SUNW.LogicalHostname (used by failover data services)
- SUNW.SharedAddress (use by scalable data services)



Note – If you accidentally remove a preregistered resource type, it can be re-registered like any resource type.

As shown in the following command output, the SUNW.HASstorage resource is not a preregistered resource type. It is not a data service either. The `scrgadm` command was executed immediately after an initial Sun Cluster software installation.

```
# scrgadm -p
```

```
Res Type name:          SUNW.LogicalHostname
  Res Type description:  Logical Hostname Resource Type
```

```
Res Type name:          SUNW.SharedAddress
  Res Type description:  HA Shared Address Resource Type
```

Failover Data Service Resources

If the node on which the data service is running (the primary node) fails, the service is migrated to another working node without user intervention. Failover services use a *failover resource group*, which is a container for application instance resources and network resources (*logical host names*). Logical host names are Internet Protocol (IP) addresses that can be configured up on one node, and, later, automatically configured down on the original node and configured up on another node (Figure 10-2).

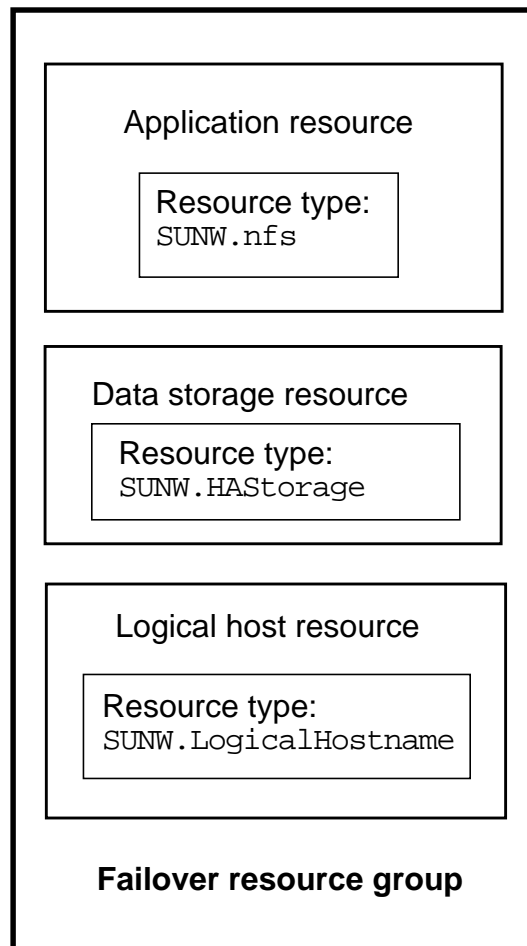


Figure 10-2 Failover Data Service

For failover data services, application instances run only on a single node. If the fault monitor detects an error, it either attempts to restart the instance on the same node, or starts the instance on another node (failover), depending on how the data service has been configured.

Scalable Data Service Resources

The scalable data service has the potential for active instances on multiple nodes. Scalable services use a *scalable resource group* to contain the application resources and a failover resource group to contain the network resources (*shared addresses*) on which the scalable service depends (Figure 10-3). The scalable resource group can be online on multiple nodes, so multiple instances of the service can be running at once. The failover resource group that hosts the shared address is online on only one node at a time. All nodes hosting a scalable service use the same shared address to host the service.

Service requests come into the cluster through a single network interface (the *global interface* or GIF) and are distributed to the nodes based on one of several predefined algorithms set by the *load-balancing policy*. For scalable services, application instances run on several nodes simultaneously. If the node that hosts the global interface fails, the global interface fails over to another node.

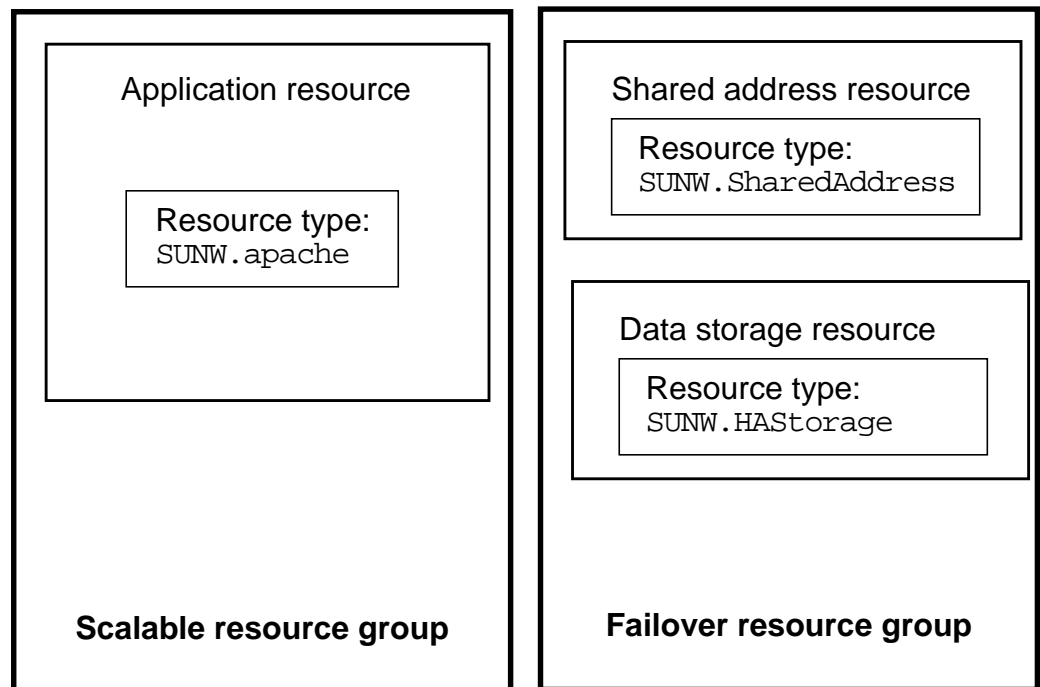


Figure 10-3 Scalable Data Service

Sun Cluster Resource Groups

The primary purpose of the Sun Cluster environment is to provide a platform on which to run data services. The data services can be of the failover or scalable types. Regardless of the data service type, they all need resources to provide effective service to clients.

Each data service configuration requires the following resource information:

- The resource group name (with a node list)
- The logical host name (client access path)
- A NAFO group name (not required)
- A data storage resource
- A data service resource
- Resource properties (mostly defaults)

Configuring a Resource Group

The process of configuring a data service mostly involves configuring a resource group and adding resources to it that are appropriate for the data service. The general process is as follows:

1. Install the data service software.
2. Register the data service (resource type).
3. Create a blank resource group and associate a list of nodes.
4. Associate resources with the new resource group:
 - a. Add a logical host name.
 - b. Add paths to data resources.
 - c. Add the data service resource.
5. Enable the resource group.

Note – Most resource-related properties default to acceptable values.



Resource Group Components

The Sun Cluster HA for NFS data service is widely used and is an example of resources that are necessary to configure a data service.

Figure 10-4 shows the general types of resources that are required for the Sun Cluster HA for NFS data service to function.

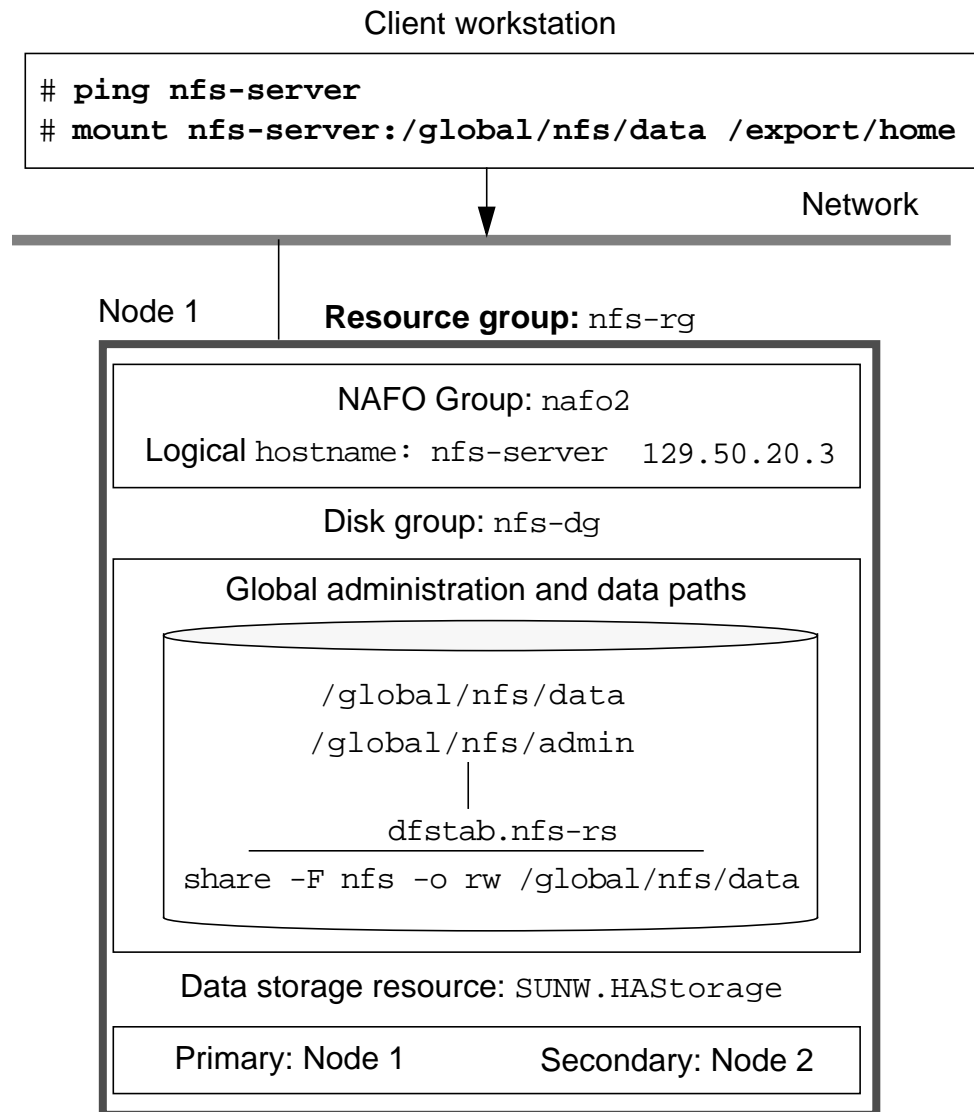


Figure 10-4 Resource Group Elements

Note – The Sun Cluster HA for NFS data service requires a small additional file system for administrative use (/global/nfs/admin in Figure 10-4).



Resource Group Administration

Use the `scrgadm` command to administer resource group. Use the `scrgadm` command to *add*, *change*, or *remove* the following:

- Resource types
- Resource groups
- Resources
- Logical host names (failover data service)
- Shared addresses (scalable data service)

The online manual pages for the `scrgadm` command are difficult to understand without practical examples. The general `scrgadm` command process to create a HA for NFS resource group follows:

1. Register resource types needed for the data service (done once after the appropriate data service software is installed).

```
# scrgadm -a -t SUNW.nfs
```

2. Create (add) a blank resource group, give it an arbitrary name, add properties that describe an administrative file system and the names of cluster hosts (nodes) that can bring the resource group online.

```
# scrgadm -a -g nfs-rg -h pnode1,pnode2 \
-y Pathprefix=/global/nfs/admin
```

3. Add the logical host name resource to the new group.

```
# scrgadm -a -L -g nfs-rg -l nfs-server
```

4. Add the storage resource (SUNW.HAStorage) to the new group.

```
# scrgadm -a -j has-res -g nfs-rg \
-t SUNW.HAStorage \
-x ServicePaths=/global/nfs/data \
-x AffinityOn=True
```

5. Add the data service resource (SUNW.nfs) to the new group.

```
# scrgadm -a -j nfs-res -g nfs-rg \
-t SUNW.nfs -y Resource_dependencies=has-res
```

6. Enable the resource group and all of its resources.

```
# scswitch -Z -g nfs-rg
```

Resource Properties

Resource types, resource groups, and resources each have multiple properties associated with them. Some of the properties can be modified; many are fixed. Figure 10-5 shows the relationship of the various resource properties.

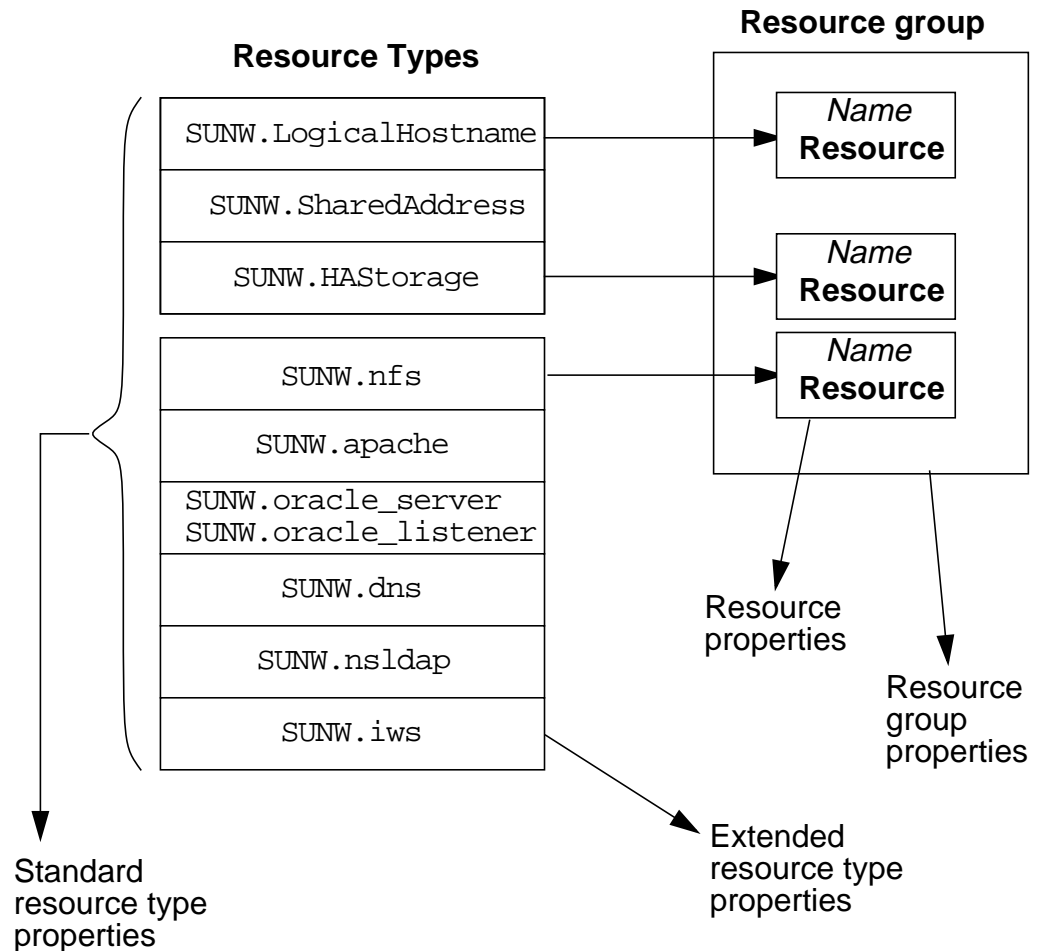


Figure 10-5 Resource Properties

When configuring resource groups, some of the properties must be modified. Most of the properties can be left at their default values.

Standard Resource Type Properties

When registering a resource type, only the `Type` property is required. The `Type` property is the name of the resource type. An example follows.

```
# scrgadm -a -t SUNW.nfs
```

Extended Resource Type Properties

Resource types also have *extended* properties that are unique to each type. Some resource types require extended resources. For instance, the `SUNW.HAStorage` resource type manages global devices associated with a resource group. You must define the path to the global data storage so that it can be managed and ensure that the storage and data service software are both resident on the same node. The `SUNW.HAStorage` resource type has two extended properties, `ServicePaths` and `AffinityOn`, that manage this. An example of their usage follows.

```
# scrgadm -a -j has-res -g nfs-rg \  
-t SUNW.HAStorage -x ServicePaths=/global/nfs/data \  
-x AffinityOn=True
```

Resource Properties

When you define a resource and add it to an existing resource group, you give the resource an arbitrary and unique name. A resource type is part of the resource definition.

You can apply standard properties to the resource. Standard resource properties apply to any resource type. The only standard resource properties that are required are `Resource_name` and `Type`. An example follows.

```
# scrgadm -a -j has-res -g nfs-rg \  
-t SUNW.HAStorage
```

Resource Group Properties

When creating (adding) a new resource group, only the resource group `RG_name` property is required. The `RG_name` property is the name of the resource group. An example follows.

```
# scrgadm -a -g my-rg
```

The `Nodelist` property defaults to all cluster nodes, but it is recommended that you supply a list nodes that can bring the resource group online. For a failover resource group, the list is in order of preference (primary/secondary) and should match the node list you supplied when you registered the disk device group that will be used by this resource group. An example follows.

```
# scrgadm -a -g web-rg -h pnode1,pnode2
```

The `Nodelist` property is not needed with a resource group intended for a scalable application. Scalable applications are usually intended to run on more than one node. The `Maximum primaries` and `Desired primaries` properties are usually explicitly set when creating a new scalable resource group. An example follows.

```
# scrgadm -a -g web-rg -y Maximum_primaries=2 \
-y Desired_primaries=2
```

A scalable resource group is usually dependent on a different resource group for proper operation. The `RG_dependencies` property is usually set when creating a scalable resource group. An example follows.

```
# scrgadm -a -g web-rg -y Maximum_primaries=2 \
-y Desired_primaries=2 -y RG_dependencies=sa-rg
```



Note – Configuring standard and extended properties for all the components of a data service is complex. The *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide* summarizes the extended properties for each resource type. There is also an appendix that lists all of the standard resource properties.

Creating Resource Groups Using the `scsetup` Utility

Resource group functions have been added to the `scsetup` utility in the Sun™ Cluster 3.0 07/01 release. After selecting Resource Groups from the `scsetup` main menu, you can create a new resource group and add resources to it. The following shows the initial dialogue when creating a new resource group.

```
*** Resource Group Menu ***
```

```
Please select from one of the following options:
```

- 1) Create a resource group
- 2) Add a network resource to a resource group
- 3) Add a data service resource to a resource group

- ?) Help
- q) Return to the previous Menu

```
Option: 1
```

```
>>> Create a Resource Group <<<
```

```
Select the type of resource group you want to add:
```

- 1) Failover Group
- 2) Scalable Group

```
Option: 1
```

```
What is the name of the group you want to add? nfs-rg  
Do you want to add an optional description (yes/no)[no]?
```

Since this cluster has two nodes, the new resource group will be configured to be hosted by both cluster nodes.

```
Which is the preferred node for hosting this group?
```

Note – The `scsetup` utility is presented in more detail in Module 11, “Data Services Configuration.”



Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Describe the primary purpose of resource groups
- ☐ List the components of a resource group
- ☐ Describe the resource group configuration process
- ☐ List the primary functions of the `scrgadm` command
- ☐ Explain the difference between standard and extended resource type properties
- ☐ List the `scsetup` utility resource group functions

Think Beyond

If the concept of a logical host did not exist, what would that imply for failover?

What complexities does having multiple backup hosts for a single logical host add to the high-availability environment?

Data Services Configuration

Objectives

Upon completion of this module, you should be able to:

- Describe the function of Sun Cluster data services
- Distinguish between highly available and scalable data services
- Describe the operation of data service fault monitors
- Configure the Sun Cluster HA for NFS failover data service
- Configure the Sun Cluster HA for Apache scalable data service
- Switch resource groups between nodes
- Monitor resource groups
- Remove resource groups

Relevance

Present the following questions to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answers to these questions, the answers should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following questions are relevant to understanding the content of this module:

- Why is NFS configured as a failover data service instead of a scalable data service?
- Why might you choose to make the Apache Web Server a failover service instead of a scalable service?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *SunTM Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *SunTM Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *SunTM Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *SunTM Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *SunTM Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *SunTM Cluster 3.0 07/01 Concepts*, part number 806-7074
- *SunTM Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *SunTM Cluster 3.0 07/01 Release Notes*, part number 806-7078

Sun Cluster Data Service Methods

Each Sun Cluster data service agent consists of specialized software that performs application-specific tasks, such as starting, stopping, and monitoring a specific application in the cluster environment. The data service agent components are referred to as *methods*.

Data Service Methods

Each Sun™ Cluster 3.0 07/01 data service agent supplies a set of data service methods. These methods run under the control of the Resource Group Manager, which uses them to start, stop, and monitor the application on the cluster nodes (Figure 11-1). These methods, along with the cluster framework software and multihost disks, enable applications to become highly available data services. As highly available data services, they can prevent significant application interruptions after any single failure within the cluster. The failure could be to a node, an interface component, or to the application itself.

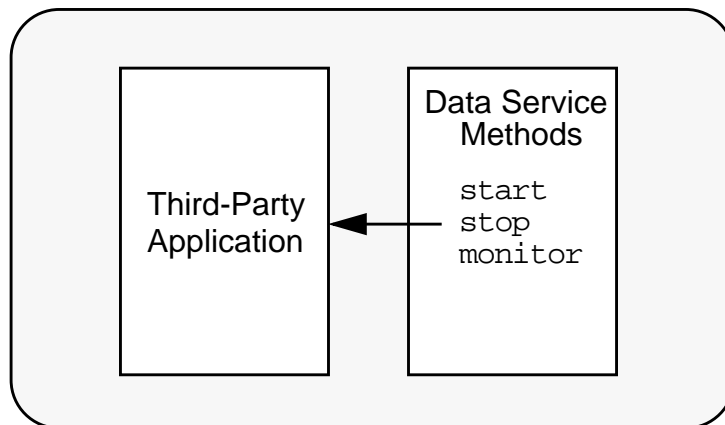


Figure 11-1 Sun Cluster Data Service Methods

Data Service Fault Monitors

Each Sun Cluster data service supplies fault monitor methods that periodically probe the data service to determine its health. A fault monitor verifies that the application daemons are running and that clients are being served. Based on the information returned by probes, predefined actions, such as restarting daemons or causing a failover, can be initiated.

The methods used to verify that the application is running or to restart the application depend upon the application, and they are defined when the data service is developed.

Sun Cluster High Availability for NFS Methods

The Sun Cluster HA for NFS data service package installs a file, `SUNW.nfs`, that equates generic functions (methods) with specific programs.

<code>PRENET_START</code>	=	<code>nfs_prenet_start;</code>
<code>START</code>	=	<code>nfs_svc_start;</code>
<code>STOP</code>	=	<code>nfs_svc_stop;</code>
<code>POSTNET_STOP</code>	=	<code>nfs_postnet_stop;</code>
<code>VALIDATE</code>	=	<code>nfs_validate;</code>
<code>UPDATE</code>	=	<code>nfs_update;</code>
<code>MONITOR_START</code>	=	<code>nfs_monitor_start;</code>
<code>MONITOR_STOP</code>	=	<code>nfs_monitor_stop;</code>
<code>MONITOR_CHECK</code>	=	<code>nfs_monitor_check;</code>

Each method program file performs tasks that are appropriate for the data service. For instance, the `START` method, when called by the resource group manager, runs the compiled program, `nfs_svc_start`. The `nfs_svc_start` program performs basic NFS startup tasks, such as verifying NFS-related daemons are running, restarting some daemons to initiate NFS client lock recovery, and exporting shared file systems.

Most of the basic functions, such as `START`, `STOP`, `MONITOR_START`, and `MONITOR_STOP`, are common to all data services. The specific files for each method, however, are different for each data service.

The fault monitoring methods usually start one or more related daemons.

Disk Device Group Considerations

One of Sun Cluster's major improvements over previous releases is the removal of the requirement that a data service must be physically attached to the physical storage storing its data. This means that in a three-or-more node cluster, the data service's resource groups might reside on a node that does not have physical access to the disk device group. File access is done over the interconnect between the data service's node and the node directly connected to the disk device group.

While this provides the benefit of faster failover, it could provide performance degradation for disk-intensive services, such as Oracle or NFS.

In addition, during the cluster boot or failover, a data service might attempt to start before global devices and cluster file systems are online. When this happens, manual intervention is required to reset the state of the resource groups.

To alleviate these problems, the Sun Cluster software includes a `SUNW.HAStorage` resource type.

The `SUNW.HAStorage` Resource Type

The resource type `SUNW.HAStorage` serves the following purposes:

- It coordinates the boot order by monitoring the global devices and cluster file systems and causing the `START` methods of the other resources in the same group with the `SUNW.HAStorage` resource to wait until the disk device resources become available.
- With the resource property `AffinityOn` set to `True`, it enforces co-location of resource groups and disk device groups on the same node, thus enhancing the performance of disk-intensive data services.

Guidelines for SUNW.HASStorage

On a two-node cluster, configuring SUNW.HASStorage is optional. To avoid additional administrative tasks, however, you could set up SUNW.HASStorage for all resource groups with data service resources that depend on global devices or cluster file systems.

You might also want to set the resource property `AffinityOn` to `True` for disk-intensive applications if performance is a concern. Be aware, however, that this increases failover times because the application does not start up on the new node until the device service has been switched over and is online on the new node.

In all cases, consult the documentation for the data service for specific recommendations.

Overview of Data Service Installation

Data service installation consists of the following steps:

- Preparing for the data service installation
- Installing and configuring the application software
- Installing the Sun Cluster data service software packages
- Registering and configuring the data service

The following sections provide an overview of the process. The actual installation and configuration of a failover and scalable data service is described later in the module.

Preparing for Data Service Installation

Proper planning will ensure that the data service installation proceeds smoothly. Before proceeding with the installation, be sure to:

- Determine the location of the application binaries
- Verify the contents of the `nsswitch.conf` file
- Plan the cluster file system configuration

Determining the Location of the Application Binaries

You can install the application software and configuration files on the local disks of each cluster node or on the cluster file system.

The advantage to placing the software and configuration files on the individual cluster nodes is that if you want to upgrade the application software later, you can do so without shutting down the cluster.

The disadvantage is that you then have several copies of the software and configuration files to maintain and administer.

If you can spare the downtime for upgrades, install the application on the cluster file system. If it is critical that the application remain up, install it on the local disks of each cluster node. In some cases, the data service documentation might contain recommendations for placement of the binaries and configuration files. Be sure to adhere to any guidelines listed in the documentation.

Verifying the Contents of the `nsswitch.conf` File

The `nsswitch.conf` file is the configuration file for name service lookups. This file determines which databases within the Solaris Operating Environment to use for name service lookups and in what order to consult the databases.

For some data services, you must change the “group” line in the file so that the entry is `cluster files`. To determine whether you need to change the “group” line, consult the documentation for the data service you are configuring.

Planning the Cluster File System Configuration

Depending on the data service, you might need to configure the cluster file system to meet Sun Cluster requirements. To determine whether any special considerations apply, consult the documentation for the data service you are configuring.

Installing and Configuring the Application Software

Installation and configuration of the application software is specific to the application itself. In some cases, you must purchase additional software.

Installing the Sun Cluster Data Service Software Packages

Sun Cluster data service software packages are included with the Sun Cluster software package on Sun Cluster Agent CD-ROM. Installation is specific to the data service.

Registering and Configuring a Data Service

The steps to register and configure a data service depend upon the particular data service and whether you will configure it as a scalable or failover service.

Sun Cluster’s release notes include an appendix with charts for planning resources and resource groups. These charts are also included in Appendix A, “Cluster Configuration Forms”. Using these charts as a guide might help the installation to proceed more smoothly.

Installing Sun Cluster HA for NFS

The Sun Cluster HA for NFS data service is an example of a failover data service. One node of the cluster serves as the NFS server. If it fails, the service fails over to one of the other nodes in the cluster.

This section details the steps for installing and configuring Sun Cluster HA for NFS on Sun Cluster servers and the steps for adding the service to a system that is already running Sun Cluster.

Preparing for Installation

It is important to perform preinstallation planning before installing any of the data services.

Determining the Location of the Application Binaries

The NFS server is part of the Solaris Operating Environment and is already installed locally on each node.

Verifying the Contents of the `nsswitch.conf` File

Ensure that the `hosts` line in `/etc/nsswitch.conf` is configured as follows:

```
hosts:          cluster files nis
```

This prevents timing-related failures because of name service lookup.

Planning the Cluster File System Configuration

Identify the global file system you created earlier to house the NFS data. This directory should already be mounted on your cluster nodes.

If you are not sure whether the directory is mounted, run the following command to verify:

```
# df -k
```

Filesystem	kbytes	used	avail	capacity	Mounted on
/dev/md/dsk/d30	1984564	821360	1103668	43%	/
/proc	0	0	0	0%	/proc
fd	0	0	0	0%	/dev/fd
mnttab	0	0	0	0%	/etc/mnttab
swap	5503960	184	5503776	1%	/var/run
swap	5503808	32	5503776	1%	/tmp
/dev/md/dsk/d70	95702	3502	82630	5%	
/global/.devices/node@2					
/dev/md/dsk/d40	95702	3502	82630	5%	
/global/.devices/node@1					
/dev/md/webds/dsk/d100					
	70011836	65819	69245899	1%	/global/web
/dev/md/nfsds/dsk/d100					
	70011836	65822	69245896	1%	/global/nfs

Installing and Configuring the Application Software

The NFS server is part of the Solaris Operating Environment and is already installed locally on each node.

Installing the Sun Cluster Data Service Software Packages

If the data service packages were installed as part of your initial Sun™ Cluster 3.0 07/01 installation, use the following command to verify the installation:

```
# pkginfo -l SUNWscnfs
  PKGINST:  SUNWscnfs
    NAME:    Sun Cluster NFS Server Component
  CATEGORY: application
    ARCH:    sparc
  VERSION:  3.0.0,REV=2000.10.01.01.00
  BASEDIR:  /opt
    VENDOR:  Sun Microsystems, Inc.
    DESC:    Sun Cluster nfs server data service
  PSTAMP:    octavia20001001013148
  INSTDATE:  Dec 05 2000 13:48
  HOTLINE:   Please contact your local service provider
  STATUS:    completely installed
  FILES:     18 installed pathnames
              3 directories
              15 executables
              2596 blocks used (approx)
```

Registering and Configuring the Data Service

The following steps complete the installation and configuration process.

1. Become superuser on a node in the cluster.
2. Verify that all nodes in the cluster are up and functional:

```
# scstat
```
3. Create a plan for the resources that will be in the failover resource group.

Planning the Resources and Resource Group

Because this is a failover data service, plan for one resource group for the application and logical host resource. Configuring HASStorage is optional but highly recommended for NFS Figure 11-2.

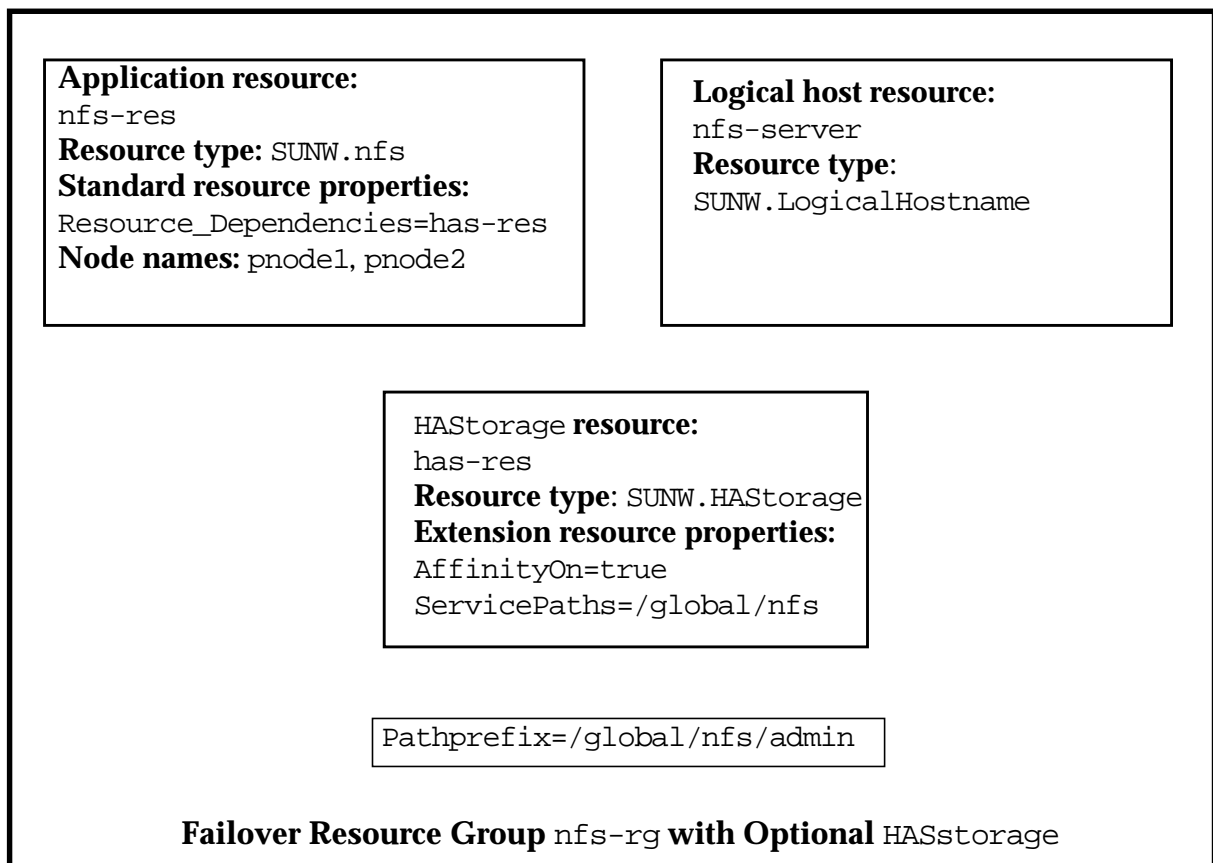


Figure 11-2 NFS Resource Group Plan

NFS Resource Pathprefix Property

The NFS data service supports a large set of standard and extension resource properties, but only one resource property must be specified at installation. This property, the Pathprefix property, specifies a directory path to hold NFS state information and dfstab files for the NFS resource.

The NFS resource contains the full list of standard and extension properties available for the NFS data service. In general, these properties contain appropriate defaults and do not need modification.

HAStorage Resource ServicePaths Extension Property

Service paths can contain global device names, paths to global devices, or the global mount point.

1. Verify that the logical host name has been added to your name service database.



Note – To avoid failures because of name service lookup, verify that the logical host name is present in the server's and client's /etc/hosts file.

2. Create the directory specified in Pathprefix.

```
# cd /global/nfs
# mkdir admin
```

3. From any node of the cluster, create the administrative directory below the Pathprefix directory for the NFS configuration files.

```
# cd admin
# mkdir SUNW.nfs
```

4. Create a dfstab.resource-name file in the newly created SUNW.nfs directory. Set up the share options for each path you have created to be shared. The format of this file is exactly the same as the format used in /etc/dfs/dfstab:

```
# cd SUNW.nfs
# vi dfstab.nfs-res
```

```
share -F nfs -o rw -d "Home Dirs" /global/nfs/data
```

The share -o rw command grants write access to all clients, including the host names used by Sun Cluster, and enables Sun Cluster HA for NFS fault monitoring to operate most efficiently.

You can specify a client list or net group for security purposes. Be sure to include all cluster nodes and logical hosts so that the cluster fault monitoring can do a thorough job.

5. Create the directory specified in the `dfstab.nfs-res` file.

```
# cd /global/nfs
# mkdir /global/nfs/data
```

6. Register the NFS and HAStorage resource types.

```
# scrgadm -a -t SUNW.nfs
# scrgadm -a -t SUNW.HAStorage
```

Note – The `scrgadm` does not produce any output if the command completes without errors.



7. Create the failover resource group.

```
# scrgadm -a -g nfs-rg -h pnode1,pnode2 \
-y Pathprefix=/global/nfs/admin
```

8. Add the logical host name resource to the resource group.

```
# scrgadm -a -L -g nfs-rg -l nfs-server
```

9. Create the SUNW.HAStorage resource.

```
# scrgadm -a -j has-res -g nfs-rg -t SUNW.HAStorage \
-x ServicePaths=/global/nfs -x AffinityOn=True
```

10. Create the NFS resource.

```
# scrgadm -a -j nfs-res -g nfs-rg -t SUNW.nfs -y
Resource_dependencies=has-res
```

11. Enable the resources and the resource monitors, manage the resource group, and switch the resource group into the online state.

```
# scswitch -Z -g nfs-rg
```

12. Verify that the data service is online.

```
# scstat -g
```

Note – Use the `scstat -g` command to monitor the status of the cluster's resources.



Testing NFS Failover

Verify that the NFS data service is working properly by switching the NFS failover resource group to the other node.

1. Use the output from the `scstat -g` command to determine which node is currently hosting the NFS server. In the following example, `pnode2` is the current host.

```
-- Resource Groups and Resources --
```

	Group Name	Resources
	-----	-----
Resources:	nfs-rg	nfs-server nfs-res has-res

```
-- Resource Groups --
```

	Group Name	Node Name	State
	-----	-----	-----
Group:	nfs-rg	pnode1	Offline
Group:	nfs-rg	pnode2	Online

```
-- Resources --
```

	Resource Name	Node Name	State	Status Message
	-----	-----	-----	-----
Resource:	nfs-server	pnode1	Offline	Offline
Resource:	nfs-server	pnode2	Online	Online -
	LogicalHostname			online.
Resource:	nfs-res	pnode1	Offline	Offline
Resource:	nfs-res	pnode2	Online	Online -
	Service is			online.
Resource:	has-res	pnode1	Offline	Offline
Resource:	has-res	pnode2	Online	Online

2. Use the `scswitch` command to switch the NFS resource group to `pnode1`.

```
# scswitch -z -h pnode1 -g nfs-rg
```



Note – As with the `scrgadm` command, the `scswitch` command only generates output if there is an error.

3. Run `scstat -g` again to verify that the switch was successful.

```
-- Resource Groups and Resources --
```

	Group Name	Resources
	-----	-----
Resources:	nfs-rg	nfs-server nfs-res has-res

```
-- Resource Groups --
```

	Group Name	Node Name	State
	-----	-----	-----
Group:	nfs-rg	pnode1	Online
Group:	nfs-rg	pnode2	Offline

```
-- Resources --
```

	Resource Name	Node Name	State	Status Message
	-----	-----	-----	-----
Resource:	nfs-server	pnode1	Offline	Online -
	LogicalHostname online			
Resource:	nfs-server	pnode2	Online	Offline -
	LogicalHostname off.			
Resource:	nfs-res	pnode1	Offline	Online -
	Successfully started NFS service			
Resource:	nfs-res	pnode2	Online	Offline -
	Completed successfully.			
Resource:	has-res	pnode1	Offline	Online
Resource:	has-res	pnode2	Online	Offline

Installing Sun Cluster Scalable Service for Apache

Sun Cluster HA for Apache is an example of a data service that can be configured as either a failover or scalable service. In this section, you learn how to configure Sun Cluster HA for Apache as a scalable service.

As a scalable service, Apache Web Server runs on all nodes of the cluster. If one of the nodes fail, the service continues to operate on the remaining nodes of the cluster.

Preparing for Installation

It is important to perform preinstallation planning before installing any of the data services.

Determining the Location of the Application Binaries

The Apache software is included in the Solaris 8 Operating Environment CD. If you installed the Entire Distribution of the Solaris 8 Operating Environment, the Apache software is already installed locally on each node.

Verify that the Apache packages are installed on your cluster by issuing the following command on all cluster nodes:

```
# pkginfo | grep Apache
```

system	SUNWapchd	Apache Web Server Documentation
system	SUNWapchr	Apache Web Server (root)
system	SUNWapchu	Apache Web Server (usr)

If the packages are not installed, you still have the option to place the binaries on a global file system. However, for performance purposes, it is recommended that you place the software on each cluster node.

Verifying the Contents of the `nsswitch.conf` File

Ensure that the `hosts` line in the `/etc/nsswitch.conf` file is configured as follows:

```
hosts:          cluster files nis
```

This prevents timing-related failures because of name service lookup.

Planning the Cluster File System Configuration

Identify the global file system you created earlier to house the Web server data. This directory should already be mounted on your cluster nodes.

If you are not sure whether the directory is mounted, run the following command to verify that the directory is mounted:

```
# df -kl
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/vx/dsk/rootvol   3256030   887735 2335735    28%      /
/proc                  0          0         0      0%      /proc
fd                     0          0         0      0%      /dev/fd
mnttab                 0          0         0      0%      /etc/mnttab
swap                  590776    176    590600     1%      /var/run
swap                  590648     48    590600     1%      /tmp
/dev/vx/dsk/rootdisk_14vol
                        96391     2429    84323     3%
/global/.devices/node@1
/dev/vx/dsk/nfsdg/vol-01
                        480751    1309    431367     1%      /global/nfs
/dev/vx/dsk/webdg/vol-01
                        480751    1344    431332     1%      /global/web
/dev/vx/dsk/rootdisk_24vol
                        96391     2429    84323     3%
/global/.devices/node@2
#
```

Installing and Configuring the Application Software

You must perform the following procedures on all nodes of the cluster. Use the cluster console to ensure that the steps are performed on each node.

If the Apache Web Server software is not installed, you can install it from the Solaris 8 Software CD-ROM 2 of 2.

```
# pkgadd -d /cdrom/cdrom0/Solaris_8/Product SUNWapchr SUNWapchu SUNWapchd
...
Installing Apache Web Server (root) as SUNWapchr
...
[ verifying class initd ]
/etc/rc0.d/K16apache linked pathname
/etc/rc1.d/K16apache linked pathname
/etc/rc2.d/K16apache linked pathname
/etc/rc3.d/S50apache linked pathname
/etc/rcS.d/K16apache linked pathname
```

1. Disable the START and STOP run control scripts that were just installed as part of the SUNWapchr package.

This step is necessary because Sun Cluster HA for Apache starts and stops the Apache application after you have configured the data service. Perform the following three steps:

- a. List the Apache run control scripts.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/K16apache
/etc/rc1.d/K16apache
/etc/rc2.d/K16apache
/etc/rc3.d/S50apache
/etc/rcS.d/K16apache
```

- b. Rename the Apache run control scripts.

```
# mv /etc/rc0.d/K16apache /etc/rc0.d/k16apache
# mv /etc/rc1.d/K16apache /etc/rc1.d/k16apache
# mv /etc/rc2.d/K16apache /etc/rc2.d/k16apache
# mv /etc/rc3.d/S50apache /etc/rc3.d/s50apache
# mv /etc/rcS.d/K16apache /etc/rcS.d/k16apache
```

- c. Verify that all the Apache-related scripts have been renamed.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/k16apache
/etc/rc1.d/k16apache
/etc/rc2.d/k16apache
/etc/rc3.d/s50apache
/etc/rcS.d/k16apache
```

2. Configure the Apache Web Server `/etc/apache/httpd.conf` configuration file.

```
# cp /etc/apache/httpd.conf-example /etc/apache/httpd.conf
# vi /etc/apache/httpd.conf
```

- a. Locate the following line in the file:

```
#ServerName new.host.name
```

- b. Remove the `#` from `ServerName` and change `new.host.name` to the name you will use for the shared address.

```
ServerName web-server
```

- c. Locate the following lines:

```
DocumentRoot "/var/apache/htdocs"
<Directory "/var/apache/htdocs">
ScriptAlias /cgi-bin/ "/var/apache/cgi-bin/"
<Directory "/var/apache/cgi-bin">
```

- d. Change the entries to point to the global directories:

```
DocumentRoot "/global/web/htdocs"
<Directory "/global/web/htdocs">
ScriptAlias /cgi-bin/ "/global/web/cgi-bin/"
<Directory "/global/web/cgi-bin">
```

- e. Save the file and exit.

3. Verify that the logical host name has been added to your name service database.



Note – To avoid any failures because of name service lookup, also verify that the shared address name is present in the server's and client's `/etc/hosts` file.

4. *On one node of the cluster only*, copy the default `htdocs` and `cgi-bin` directories to the global location as shown.

```
# cp -rp /var/apache/htdocs /global/web
# cp -rp /var/apache/cgi-bin /global/web
```

Testing the Application Software Installation

Test the server on each node before configuring the data service resources.

1. Start the server.

```
# /usr/apache/bin/apachectl start
```

2. Verify that the server is running.

```
# ps -ef | grep httpd
nobody  490   488  0 15:36:27 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
  root    488     1  0 15:36:26 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
  nobody  489   488  0 15:36:27 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
  nobody  491   488  0 15:36:27 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
  nobody  492   488  0 15:36:27 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
  nobody  493   488  0 15:36:27 ?           0:00 /usr/apache/bin/httpd -f
/etc/apache/httpd.conf
```

3. Connect to the server from the Web browser on your console server. Use `http://nodename` where *nodename* is the name of one of your cluster nodes (Figure 11-3 on page 11-23).

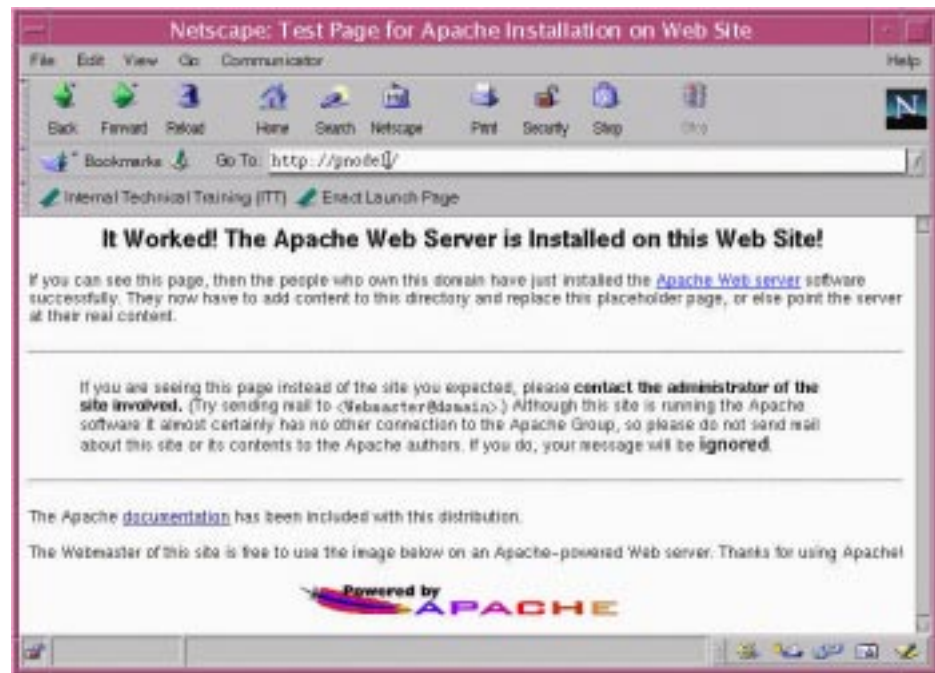


Figure 11-3 Apache Server Test Page

4. Stop the Apache Web Server.

```
# /usr/apache/bin/apachectl stop
```

5. Verify that the server has stopped.

```
# ps -ef | grep httpd
root  8394  8393  0 17:11:14 pts/6    0:00 grep httpd
```

Installing the Sun Cluster Data Service Software Packages

The data service packages were installed as part of your initial Sun™ Cluster 3.0 07/01 installation. Use the following command to verify the installation:

```
# pkginfo -l SUNWscapc
  PKGINST:  SUNWscapc
    NAME:    Sun Cluster Apache Web Server Component
  CATEGORY:  application
    ARCH:    sparc
  VERSION:   3.0.0,REV=2000.10.01.01.00
  BASEDIR:   /opt
    VENDOR:  Sun Microsystems, Inc.
    DESC:    Sun Cluster Apache web server data service
  PSTAMP:    octavia20001001013147
  INSTDATE:  Dec 05 2000 13:48
  HOTLINE:   Please contact your local service provider
  STATUS:    completely installed
  FILES:     13 installed pathnames
              3 directories
              10 executables
              1756 blocks used (approx)
```


Registering and Configuring the Data Service

The following steps complete the installation and configuration process.

1. Become superuser on a node in the cluster.
2. Verify that all nodes in the cluster are up and functional:

```
# scstat
```

3. Create a plan for the resources that will be in the failover resource group.

Planning the Resources and Resource Group

Because this is a scalable data service, plan for one resource group for the application and one resource group for the shared address (Figure 11-4). Configuring HAStorage is optional but highly recommended for scalable services. It is not included in the following examples for simplicity. However, the procedure for adding HAStorage is identical to the example for the NFS failover data service.

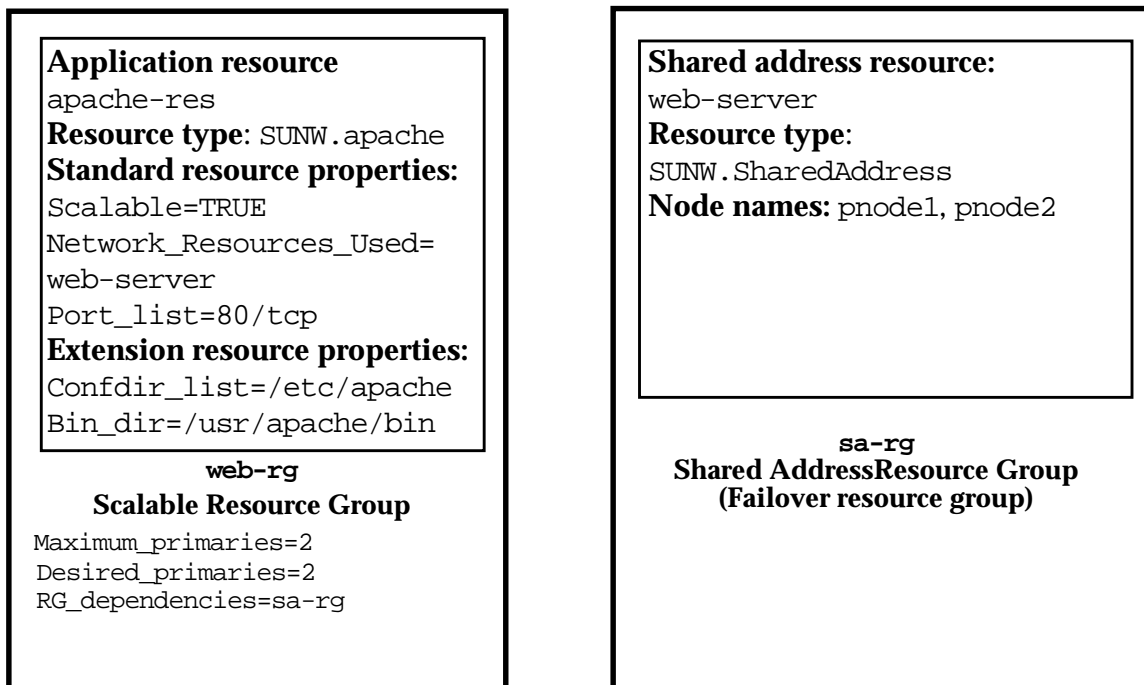


Figure 11-4 Apache Web Server Sample Resource Plan

Scalable Resource Group Properties

As with resources, resource groups have standard properties that in most cases default to reasonable values. However, scalable resource groups require that the following resource group properties be set.

- | | |
|---|--|
| <code>-y Maximum primaries=<i>m</i></code> | Specifies the maximum number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is 1. |
| <code>-y Desired primaries=<i>n</i></code> | Specifies the desired number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is 1. |
| <code>-y RG_dependencies=
<i>resource-group-name</i></code> | Identifies the resource group that contains the shared address resource on which the resource group being created depends. |

Scalable Resource Properties

The following lists the required Apache application standard and extension resource properties. The `Port_list` property is not technically required if you use the default ports, but it is required if you use more than one port or a non-standard port.

- | | |
|--|---|
| <code>-y Network_resources_used=
<i>network-resource, ...</i></code> | Specifies a comma-separated list of network resource names that identify the shared addresses used by the data service. |
| <code>-y Port_list=port-
<i>number/protocol, ...</i></code> | Specifies a comma-separated list of port numbers and protocol to be used. Defaults to <code>80/tcp</code> . |
| <code>-y Scalable=</code> | Specifies a required parameter for scalable services. It must be set to <code>True</code> . |
| <code>-x Confdir_list=config-
<i>directory,...</i></code> | Specifies a comma-separated list of the locations of the Apache configuration files. This is a required extension property. |
| <code>-x Bin_dir=<i>bin-directory</i></code> | Specifies the location where the Apache binaries are installed. This is a required extension property. |

The *Data Services Installation and Configuration Guide* contains the full list of standard and extension properties available for the Apache data service. In general, these properties contain appropriate defaults and do not need modification.

4. Register the resource type for the Apache data service.

```
# scrgadm -a -t SUNW.apache
```

5. Create a failover resource group to hold the shared network address.

```
# scrgadm -a -g sa-rg -h pnode1,pnode2
```

6. Add a network resource to the failover resource group.

```
# scrgadm -a -S -g sa-rg -l web-server
```

7. Create a scalable resource group to run on all nodes of the cluster.

```
# scrgadm -a -g web-rg -y Maximum primaries=2 \  
-y Desired primaries=2 -y RG_dependencies=sa-rg
```

8. Create an application resource in the scalable resource group.

```
# scrgadm -a -j apache-res -g web-rg \  
-t SUNW.apache -x Confdir_list=/etc/apache -x Bin_dir=/usr/apache/bin \  
-y Scalable=TRUE -y Network_Resources_Used=web-server
```

9. Bring the failover resource group online.

```
# scswitch -Z -g sa-rg
```

10. Bring the scalable resource group online.

```
# scswitch -Z -g web-rg
```

11. Verify that the data service is online.

```
# scstat -g
```

12. Connect to the server from the Web browser on your console server using `http://web-server`. The test page should be identical to the test you ran to test the application installation.

Advanced Resource Commands

The following examples demonstrate how to use the `scswitch` command to perform advanced operations on resource groups, resources, and data service fault monitors.

Advanced Resource Group Operations

Use the following commands to:

- Shut down a resource group
`# scswitch -F -g nfs-rg`
- Turn on a resource group
`# scswitch -Z -g nfs-rg`
- Restart a resource group
`# scswitch -R -h node,node -g nfs-rg`
- Evacuate all resources and resource groups from a node
`# scswitch -S -h node`

Advanced Resource Operations

Use the following commands to:

- Disable a resource and its fault monitor
`# scswitch -n -j nfs-res`
- Enable a resource and its fault monitor
`# scswitch -e -j nfs-res`

Advanced Fault Monitor Operations

Use the following commands to:

- Disable the fault monitor for a resource

```
# scswitch -n -M -j nfs-res
```

- Enable a resource fault monitor

```
# scswitch -e -M -j nfs-res
```



Note – You should not manually kill any resource group operations that are underway. Operations, such as `scswitch`, must be allowed to complete.

Exercise: Installing and Configuring Sun Cluster HA for NFS

In this exercise, you complete the following tasks:

- Prepare for Sun Cluster HA for NFS data service registration and configuration
- Use the `scrgadm` command to register and configure the Sun Cluster HA for NFS data service
- Verify that the Sun Cluster HA for NFS data service is registered and functional
- Verify that the Sun Cluster HA for NFS file system is mounted and exported
- Verify that clients can access Sun Cluster HA for NFS file systems
- Switch the Sun Cluster HA for NFS data services from one server to another
- Use the `scrgadm` and `scswitch` commands to remove the HA for NFS resource group
- Use the `scsetup` utility to register and configure the Sun Cluster HA for NFS resource group

Preparation

The following tasks are explained in this section:

- Preparing for Sun Cluster HA for NFS registration and configuration
- Registering and configuring the Sun Cluster HA for NFS data service
- Verifying access by NFS clients
- Observing Sun Cluster HA for NFS failover behavior



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Preparing for Sun Cluster HA for NFS Data Service Configuration

In earlier exercises, you created the global file system for NFS. Confirm that this file system is available and ready to configure for Sun Cluster HA for NFS. Perform the following steps:

1. Log in to Node 1 as user `root`.
2. Verify that your cluster is active.
3. Verify that the `global/nfs` file system is mounted and ready for use.

```
# scstat -p
```

4. Verify the `hosts` line in the `/etc/nsswitch.conf` file. If necessary, correct it to read:

```
hosts:      cluster files nis [NOTFOUND=return]
```

5. On both nodes, install the Sun Cluster HA for NFS data service software package by running `scinstall`. Use option 4.

```
# scinstall
```

Note – You must furnish the full path to the data service software. This is the location of the `.cdtoc` file and not the location of the packages.

6. Add an entry to the `/etc/hosts` file on each cluster node and the administrative workstation for the logical host name resource `clustername-nfs`. Substitute the IP address supplied by your instructor.

```
IP_address    clustername-nfs
```

Perform the remaining steps on just one node of the cluster.

7. Create the administrative directory that will contain the `dfstab.nfs-res` file for the NFS resource.

```
# cd /global/nfs  
# mkdir admin  
# cd admin  
# mkdir SUNW.nfs
```



8. Create the `dfstab.nfs-res` file in the `/global/nfs/admin/SUNW.nfs` directory. Add the entry to share `/global/nfs/data`.

```
# cd SUNW.nfs
# vi dfstab.nfs-res
```

```
share -F nfs -o rw -d"Home Dirs" /global/nfs/data
```

9. Create the directory specified in the `dfstab.nfs-res` file.

```
# cd /global/nfs
# mkdir /global/nfs/data
# chmod 777 /global/nfs/data
# touch /global/nfs/data/sample.file
```



Note – You are changing the mode of the home directory only for the purposes of this lab. In practice, you would be more specific about the share options in the `dfstab.nfs-res` file.

Task – Registering and Configuring the Sun Cluster HA for NFS Data Service

Perform the following steps to register the Sun Cluster HA for NFS data service:

1. From one node, register the NFS and HAStorage resource types.

```
# scrgadm -a -t SUNW.nfs
# scrgadm -a -t SUNW.HAStorage
# scrgadm -p
```



Note – The `scrgadm` command only produces output to the screen if there are errors executing the commands. If the command executes and returns, that indicates successful creation.

2. Create the failover resource group.

```
# scrgadm -a -g nfs-rg -h node1,node2 \
-y Pathprefix=/global/nfs/admin
```

3. Add the logical host name resource to the resource group.

```
# scrgadm -a -L -g nfs-rg -l clustername-nfs
```

4. Create the SUNW.HAStorage resource.

```
# scrgadm -a -j has-res -g nfs-rg -t SUNW.HAStorage \
-x ServicePaths=/global/nfs -x AffinityOn=True
```

5. Create the SUNW.nfs resource.

```
# scrgadm -a -j nfs-res -g nfs-rg -t SUNW.nfs \
-y Resource_dependencies=has-res
```

6. Enable the resources and the resource monitors, manage the resource group, and switch the resource group into the online state.

```
# scswitch -Z -g nfs-rg
```

7. Verify that the data service is online.

```
# scstat -g
```

Task – Verifying Access by NFS Clients

Perform the following steps to verify that NFS clients can access the Sun Cluster HA for NFS file system:

1. On the administration workstation, verify that you can access the cluster file system.


```
# ls -l /net/clustername-nfs/global/nfs/data  
total 0
```
2. On the administration workstation, copy the `Scripts/test.nfs` file into the `root` directory.
3. Edit the `/test.nfs` script and verify the logical host name and NFS file system names are correct.

When this script is running, it creates and writes to an NFS-mounted file system. It also displays the time to standard output (`stdout`). This script helps to time how long the NFS data service is interrupted during switchovers and takeovers.

Task – Observing Sun Cluster HA for NFS Failover Behavior

Now that the Sun Cluster HA for NFS environment is working properly, test its high-availability operation by performing the following steps:

1. On the administration workstation, start the `test.nfs` script.
2. On one node of the cluster, determine the name of the node currently hosting the Sun Cluster HA for NFS service.
3. On one node of the cluster, use the `scswitch` command to transfer control of the NFS service from one HA server to the other.

```
# scswitch -z -h dest-node -g nfs-rg
```

Substitute the name of your offline node for *dest-node*.

4. Observe the messages displayed by the `test.nfs` script.
5. How long was the Sun Cluster HA for NFS data service interrupted during the switchover from one physical host to another?

6. Use the `mount` and `share` commands on both nodes to verify which file systems they are now mounting and exporting.

7. Use the `ifconfig` command on both nodes to observe the multiple IP addresses (physical and logical) configured on the same physical network interface.

```
# ifconfig -a
```

8. On one node of the cluster, use the `scswitch` command to transfer control of the NFS service back to its preferred host.

```
# scswitch -z -h dest-node -g nfs-rg
```

Task – Removing the `nfs-rg` Resource Group

Removing an operational resource group can be confusing. The general process is to:

- Take the resource group offline
- Disable all the resources that are part of the resource group
- Remove all resources from the resource group
- Remove the resource group

Perform the following steps to remove the `nfs-rg` resource group:

1. Take the `nfs-rg` resource group offline.

```
# scswitch -F -g nfs-rg
```

2. Identify the names of resources in the `nfs-rg` resource group.

```
# scrgadm -p -g nfs-rg |grep "Res name"
Res name:      devcluster-nfs
Res name:      nfs-res
Res name:      has-res
```

3. Disable all of the resource in the `nfs-rg` resource group.

```
# scswitch -n -j nfs-res
# scswitch -n -j has-res
# scswitch -n -j devcluster-nfs
```

4. Remove the disabled resources.

```
# scrgadm -r -j nfs-res
# scrgadm -r -j has-res
# scrgadm -r -j devcluster-nfs
```

5. Remove the `nfs-rg` resource group.

```
# scrgadm -r -g nfs-rg
```

Task – Creating a Resource Group Using the `scsetup` Utility

Perform the following steps to recreate the `nfs-rg` resource group:

Recreate the `nfs-rs` resource group using the `scsetup` utility Resource groups option. Following is a summary of questions you will be asked.

1. Start the `scsetup` utility on Node 1. From the main menu, select option 2, Resource Groups, and then select option 1, Create a resource group, from the Resource Group Menu.
2. Configure the basic `nfs-rg` failover group again as follows:

Select the type of resource group you want to add:

- 1) Failover Group
- 2) Scalable Group

Option: **1**

What is the name of the group you want to add? **nfs-rg**

Do you want to add an optional description (yes/no) [no]? **no**

Which is the preferred node for hosting this group? **node1**

Do you want to specify such a directory now (yes/no) [no]? **yes**

What is the name of the directory (i.e., "Pathprefix")? **/global/nfs/admin**

Is it okay to proceed with the update (yes/no) [yes]? **yes**

3. Configure the LogicalHostname resource (`clustername-nfs`).

Do you want to add any network resources now (yes/no) [yes]? **yes**

Select the type of network resource you want to add:

- 1) LogicalHostname
- 2) SharedAddress

Option: **1**

For how many subnets do you want to add such a resource [1]? **1**

Configure the first logicalhostname resource (yes/no) [yes]? **yes**

What logical hostname do you want to add? **clustername-nfs**

Is it okay to proceed with the update (yes/no) [yes]? **yes**

4. Configure the SUNW.nfs resource.

Do you want to add any data service resources now (yes/no) [yes]? **yes**
Please wait - looking up resource types

Please select the type of resource you want to add:

- | | |
|-------------------|--------------------------|
| 1) SUNW.nfs | HA-NFS for Sun Cluster |
| 2) SUNW.HAStorage | HA Storage Resource Type |

Option: **1**

What is the name of the resource you want to add? **nfs-res**
Any extension properties you would like to set (yes/no) [yes]? **no**
Is it okay to proceed with the update (yes/no) [yes]? **yes**

5. Configure the SUNW.HAStorage resource.

Do you want to add any additional data service resources(yes/no)[no]? **yes**
Please wait - looking up resource types

Please select the type of resource you want to add:

- | | |
|-------------------|--------------------------|
| 1) SUNW.nfs | HA-NFS for Sun Cluster |
| 2) SUNW.HAStorage | HA Storage Resource Type |

Option: **2**

What is the name of the resource you want to add? **has-res**
Any extension properties you would like to set (yes/no) [yes]? **yes**
Here are the extension properties for this resource:

Property Name	Default Setting
=====	=====
ServicePaths	<NULL>
AffinityOn	True

Please enter the list of properties you want to set:
(Type Ctrl-D to finish OR "?" for help)

Property name:	ServicePaths
Property description:	The list of HA service paths to be checked
Property value:	/global/nfs
Property name:	^D

Here is the list of extension properties you entered:

ServicePaths=/global/nfs

Is it correct (yes/no) [yes]? **yes**

Is it okay to proceed with the update (yes/no) [yes]? **yes**

6. Complete the configuration and bring the resource group online.

Do you want to add any additional data service resources(yes/no) [no]?
no

Do you want to bring this resource group online now(yes/no) [yes]? **yes**

7. Quit the scsetup utility.

8. Type `scstat -g` to verify the status of the recreated `nfs-rg` resource group.

Exercise: Installing and Configuring Sun Cluster Scalable Service for Apache

In this exercise, you complete the following tasks:

- Prepare for Sun Cluster HA for Apache data service registration and configuration
- Use the `scrgadm` command to register and configure the Sun Cluster HA for Apache data service
- Verify that the Sun Cluster HA for Apache data service is registered and functional
- Verify that clients can access the Apache Web Server
- Verify the functionality of the scalable service
- Use the `scrgadm` and `scswitch` commands to remove the Apache Web Server resource groups
- Use the `scsetup` utility to register and configure the Sun Cluster HA for Apache data service

Preparation

The following tasks are explained in this section:

- Preparing for Sun Cluster HA for Apache registration and configuration
- Registering and configuring the Sun Cluster HA for Apache data service
- Verifying Apache Web Server access and scalable functionality



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Preparing for HA-Apache Data Service Configuration

Perform the following steps on each node of the cluster:

1. Install the Sun Cluster Apache data service software package by running `scinstall` on each node. Use option 4.

```
# scinstall
```

2. Disable the `START` and `STOP` run control scripts that were just installed as part of the `SUNWapchr` package.

This step is necessary because Sun Cluster HA for Apache starts and stops the Apache application after you have configured the data service. Perform the following three steps:

- a. List the Apache run control scripts.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/K16apache
/etc/rc1.d/K16apache
/etc/rc2.d/K16apache
/etc/rc3.d/S50apache
/etc/rcS.d/K16apache
```

- b. Rename the Apache run control scripts.

```
# mv /etc/rc0.d/K16apache /etc/rc0.d/k16apache
# mv /etc/rc1.d/K16apache /etc/rc1.d/k16apache
# mv /etc/rc2.d/K16apache /etc/rc2.d/k16apache
# mv /etc/rc3.d/S50apache /etc/rc3.d/s50apache
# mv /etc/rcS.d/K16apache /etc/rcS.d/k16apache
```

- c. Verify that all the Apache-related scripts have been renamed.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/k16apache
/etc/rc1.d/k16apache
/etc/rc2.d/k16apache
/etc/rc3.d/s50apache
/etc/rcS.d/k16apache
```

3. Create an entry in `/etc/hosts` for the shared address you will be configuring with the Apache Web server. In addition, create the entry on the administration workstation. Substitute the IP address supplied by your instructor.

```
IP_address      clustername-web
```

4. Copy the sample `/etc/apache/httpd.conf-example` to `/etc/apache/httpd.conf`.

```
# cp /etc/apache/httpd.conf-example /etc/apache/httpd.conf
```

5. Edit the `/etc/apache/httpd.conf` file, and change the following entries as shown.

From:

```
#ServerName new.host.name
DocumentRoot "/var/apache/htdocs"
<Directory "/var/apache/htdocs">
ScriptAlias /cgi-bin/ "/var/apache/cgi-bin/"
<Directory "/var/apache/cgi-bin">
```

To:

```
ServerName clustername-web (Uncomment the line)
DocumentRoot "/global/web/htdocs"
<Directory "/global/web/htdocs">
ScriptAlias /cgi-bin/ "/global/web/cgi-bin"
<Directory "/global/web/cgi-bin">
```

On one node of the cluster, perform the following steps:

6. Create directories for the Hypertext Markup Language (HTML) and Common GateWay Interface (CGI) files.

```
# mkdir /global/web/htdocs
# mkdir /global/web/cgi-bin
```

7. Copy the sample HTML documents to the `htdocs` directory.

```
# cp -rp /var/apache/htdocs /global/web
# cp -rp /var/apache/cgi-bin /global/web
```

8. Copy the file called “test-apache.cgi” from the classroom server to /global/web/cgi-bin. You use this file to test the scalable service. Make sure that test-apache.cgi is executable by all users.

```
# chmod 755 /global/web/cgi-bin/test-apache.cgi
```

Task – Registering and Configuring the Sun Cluster HA for Apache Data Service

Perform the following steps only on Node 1:

1. Register the resource type required for the Apache data service.

```
# scrgadm -a -t SUNW.apache
```

2. Create a failover resource group for the shared address resource. Use the appropriate node names for the -h argument.

```
# scrgadm -a -g sa-rg -h node1,node2
```

3. Add the Shared Address logical hostname resource to the resource group.

```
# scrgadm -a -S -g sa-rg -l clustername-web
```

4. Create a scalable resource group to run on all nodes of the cluster.

```
# scrgadm -a -g web-rg -y Maximum primaries=2 \
-y Desired primaries=2 -y RG_dependencies=sa-rg
```

5. Create an application resource in the scalable resource group.

```
# scrgadm -a -j apache-res -g web-rg \
-t SUNW.apache -x Confdir_list=/etc/apache -x Bin_dir=/usr/apache/bin \
-y Scalable=TRUE -y Network_Resources_Used=clustername-web
```

6. Bring the failover resource group online.

```
# scswitch -Z -g sa-rg
```

7. Bring the scalable resource group online.

```
# scswitch -Z -g web-rg
```

8. Verify that the data service is online.

```
# scstat -g
```

Task – Verifying Apache Web Server Access and Scalable Functionality

Perform the following steps to verify Apache Web Server access and scalable functionality:

1. Connect to the Web server using the browser on the administrator workstation using `http://clustername-web/cgi-bin/test-apache.cgi`.
2. Repeatedly press the Refresh or Reload button on the browser. The `test-apache.cgi` script displays the name of the cluster node that is currently servicing the request. It might take several iterations before the packet is distributed to a new node.

Task – Optional Resource Group Exercise

Ask your instructor if there is enough time to perform this exercise. If your instructor approves, perform the following steps to first remove the `sa-rg` and `web-rg` resource group and then recreate them using the `scsetup` utility.

1. Take the `sa-rg` and `web-rg` resource groups offline.

```
# scswitch -F -g web-rg
# scswitch -F -g sa-rg
```
2. Verify the resources associated with the target resource groups.

```
# scrgadm -p -g web-rg | grep "Res name"
Res name:                apache-res
# scrgadm -p -g sa-rg | grep "Res name"
Res name:                devcluster-web
```
3. Disable and remove all of the resource in the `web-rg` resource group.

```
# scswitch -n -j apache-res
# scrgadm -r -j apache-res
# scswitch -n -j devcluster-web
# scrgadm -r -j devcluster-web
```
4. Remove the `sa-rg` and `web-rg` resource groups.

```
# scrgadm -r -g web-rg
# scrgadm -r -g sa-rg
```

5. Use the `scsetup` utility to recreate the `sa-rg` failover resource group with the following features:
 - It is a failover group.
 - The preferred host is `node1`.
 - It does not use a configuration data file system.
 - It uses the shared address network resource.
 - It uses a logical host name `clustername-web`.
 - It uses no data service resources.



Caution – Before proceeding, verify that the `clustername-web` resource and the `sa-rg` resource group are both online (`scstat -g`).

6. Use the `scsetup` utility to recreate the `web-rg` scalable resource group with the following features:
 - It is dependent on the `sa-rg` resource group.
 - It does not use a configuration data file system.
 - It uses the `SUNW.apache` resource type named `apache-res`.
 - It uses the shared address resource named `clustername-web`.
 - It uses a data service port numbered 80.
 - It uses the extension `Confdir_list=/etc/apache`.
 - It uses the extension `Bin_dir=/usr/apache/bin`.
 - It uses no additional data service resources.
7. Test the Apache Web Server data service again using the `http://clustername-web/cgi-bin/test-apache.cgi` address.

Exercise Summary



Discussion – Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, highlight just the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace.

Check Your Progress

Before continuing on to the next module, check that you are able to accomplish or answer the following:

- ☐ Describe the function of Sun Cluster data services
- ☐ Distinguish between highly available and scalable data services
- ☐ Describe the operation of data service fault monitors
- ☐ Configure the Sun Cluster HA for NFS failover data service
- ☐ Configure the Apache Web Server scalable data service
- ☐ Switch resource groups between nodes
- ☐ Monitor resource groups
- ☐ Remove resource groups

Think Beyond

How difficult is it to configure additional data services while the cluster is in operation?

Using SunPlex™ Manager

Objectives

Upon completion of this module, you should be able to:

- Install and configure the SunPlex™ management software
- List the main features of SunPlex Manager
- Use SunPlex Manager to verify cluster status
- Use SunPlex Manager to remove a resource group
- Use SunPlex Manager to configure a resource group

Relevance

Present the following question to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answer to this question, the answer should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following question is relevant to understanding the content of this module:

- With the increasing size and scalability of Sun Cluster installations, how complex might administration become?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *Sun™ Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *Sun™ Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *Sun™ Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *Sun™ Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *Sun™ Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *Sun™ Cluster 3.0 07/01 Concepts*, part number 806-7074
- *Sun™ Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *Sun™ Cluster 3.0 07/01 Release Notes*, part number 806-7078

SunPlex Manager Introduction

SunPlex Manager is a graphical user interface (GUI) that enables you to graphically display cluster information, monitor configuration changes, and check the status of cluster components.

It also allows you to perform some administrative tasks, including installing and configuring some data service applications. However, the SunPlex Manager currently cannot perform all Sun Cluster administrative tasks. You must still use the command-line interface for some operations.

As shown in Figure 12-1, the SunPlex Manager software packages are installed on each node in the cluster. You access the SunPlex management functions from a Web browser on a configurable port number.

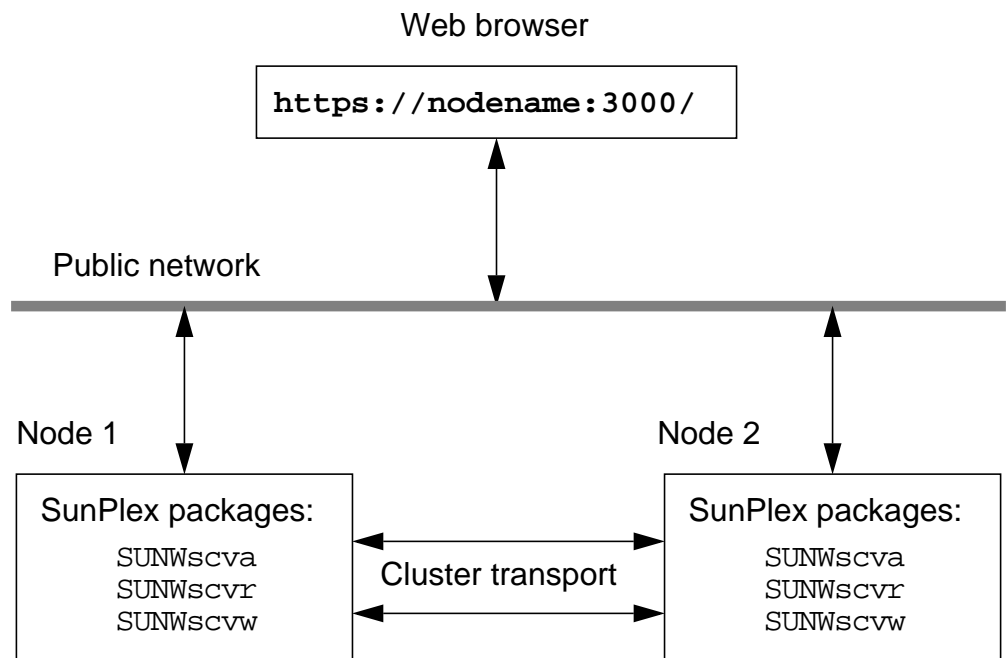


Figure 12-1 SunPlex Manager Configuration

SunPlex Manager starts in either Sun Cluster installation mode or Sun Cluster administration mode. The startup mode is dependent on the current software configuration of the nodes as follows:

- SunPlex Manager starts in installation mode if no Sun Cluster software is detected on the node.
- SunPlex Manager starts in administration mode if the Sun Cluster software is already configured on the node.

SunPlex Manager Configuration

Installing and configuring SunPlex Manager on cluster nodes requires little effort.

SunPlex Manager Installation Overview

Installing and configuring the SunPlex Manager application consists of the following steps:

1. Verify the state of the required software packages on each node.

```
# pkginfo SUNWscva SUNWscvr SUNWscvw
application SUNWscva      Apache SSL Components
application SUNWscvr      SunPlex Manager Root Components
application SUNWscvw      SunPlex Manager Core Components
```

2. Configure the SunPlex management account on each node.

- You can access the SunPlex management features on each node through the root account with no modification
- You can set up special SunPlex user accounts and passwords using the `htpasswd` program.

Note – The SunPlex accounts must have identical user IDs and passwords on all nodes. The `htpasswd` program is part of the `SUNWscvw` software package.



3. On your administrative console system, disable your Web browser's proxy feature. Proxies are not compatible with SunPlex Manager.
4. From the administrative console system, connect to the SunPlex manager software on a particular node.

`https://nodename:3000/`

Logging In to SunPlex Manager

When you connect to the SunPlex Manager application on a cluster node, you must complete the login form shown in Figure 12-2.

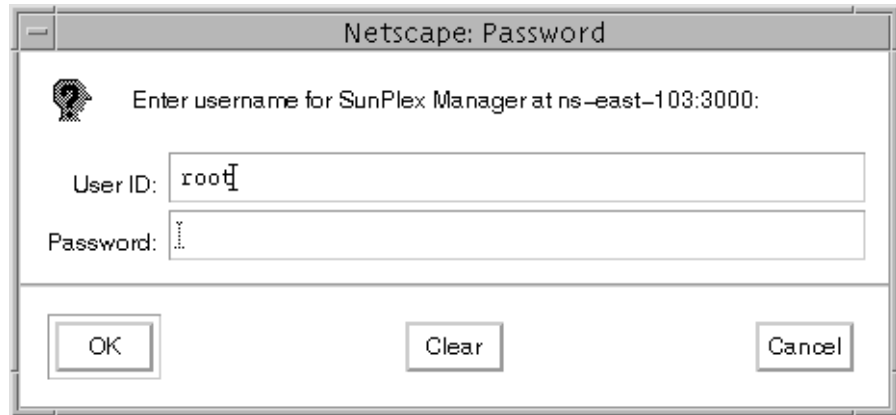
A Netscape password dialog box titled "Netscape: Password". It contains a question mark icon and the text "Enter username for SunPlex Manager at ns-east-103:3000:". Below this, there are two input fields: "User ID:" with the text "root" entered, and "Password:" which is empty. At the bottom, there are three buttons: "OK", "Clear", and "Cancel".

Figure 12-2 SunPlex Manager Login Form



Note – The `root` account must have a password set. If a password is not set, the login authentication fails.

All cluster nodes can be managed from the single login. If the node you are logged in to fails, you must to log in to the SunPlex manager application on a different node.

SunPlex Manager Initial Display

The initial SunPlex Manager display has the following features shown in Figure 12-3:

- Component navigation tree
- Component View and Action buttons
- Component status window



Figure 12-3 SunPlex Manager Initial Display

Each device tree major node can be expanded into more detailed structures.

The View button menu has Status Table and Topology entries for each major device tree selection.

The Action button menu has a wide range of administration operations appropriate for the current device tree selection.

The component status area displays detailed status or topology for the current device tree selection.

SunPlex Manager Device Tree

As shown in Figure 12-4, the SunPlex Manager device tree can be expanded to display all of the major component of a cluster. Each entry can be selected for detailed status displays in the status window.

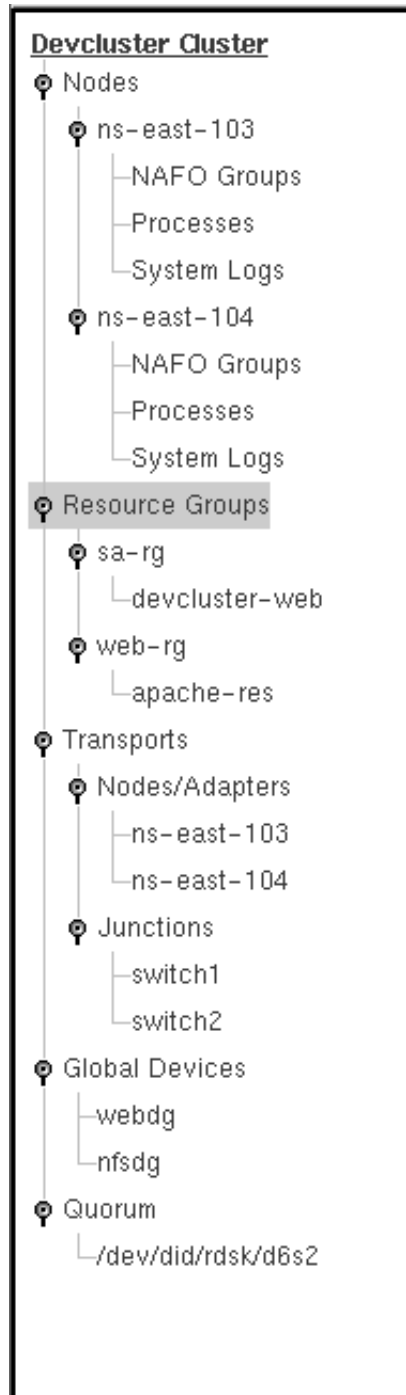


Figure 12-4 SunPlex Manager Device Tree

SunPlex Manager Resource Groups Actions

Each major device tree node has an associated list of actions. The image in Figure 12-5 shows the actions for the Resource Groups device tree node.

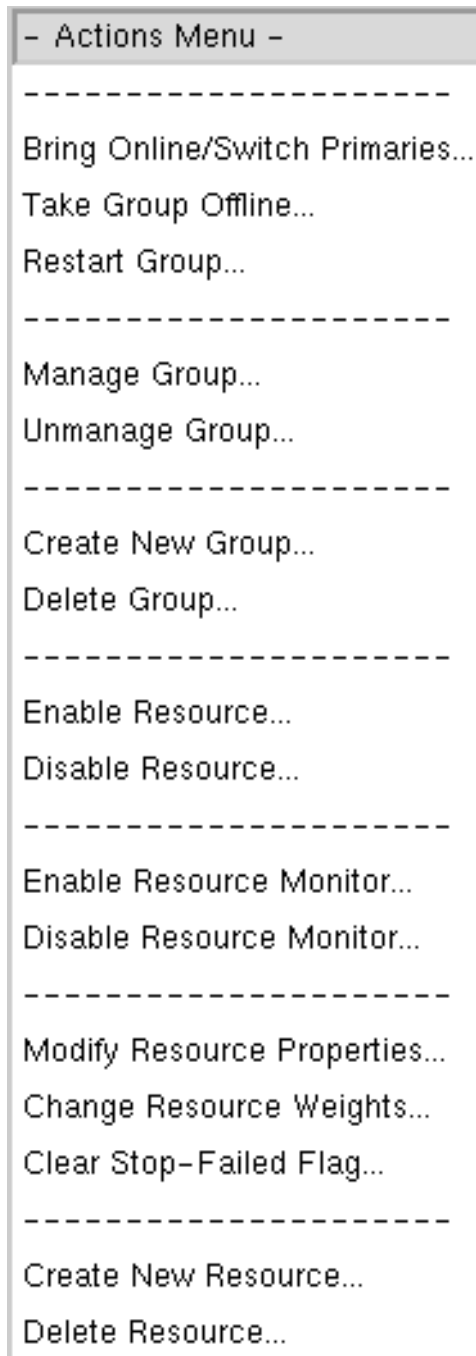


Figure 12-5 Resource Groups Actions List

SunPlex Manager Administration Summary

Each node and subnode in the SunPlex Manager application device tree displays different information and has some useful actions. The complete list of status and actions is too great to present here. Following are some highlights of the SunPlex Manager device tree actions.

Nodes Summary

The actions associated with device tree Nodes entries are:

- Access to each node using a telnet window
- Network automatic failover (NAFO) Groups status and administration
- System Processes status
- System Logs status

Resource Groups Summary

The actions associated with device tree Resource Groups entries are:

- Resource group status
- Resource status
- Resource group administration

Transports Summary

The actions associated with device tree Transports entries are:

- Cluster interconnect status
- Cluster interconnect configuration
- Cluster interconnect control

Global Devices

The actions associated with device tree Global Devices entries are:

- Device group administration
- Device group control

Quorum

The actions associated with device tree Quorum entries are:

- Quorum device status,
- Quorum device configuration
- Quorum device control



Caution – Do not perform any SunPlex actions unless you are sure of the results. You can accidentally make your production system unavailable.

Exercise: Configuring SunPlex Manager

In this exercise, you complete the following tasks:

- Verify that the SunPlex Manager software is installed
- Practice using the SunPlex Manager status features
- Use SunPlex Manager to remove a data service configuration
- Use SunPlex Manager to configure a scalable data service.

Preparation

Your administration console system must have a Web browser available such as Netscape.



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

Task – Installing SunPlex Manager

Although the SunPlex Manager software should be present on the cluster nodes, perform the following steps to verify that all of the SunPlex manager packages are installed on each node.

Perform the following steps to install SunPlex Manager:

1. On each node, verify the SunPlex Manager packages are installed.

```
# pkginfo SUNWscva SUNWscvr SUNWscvw
application SUNWscva      Apache SSL Components
application SUNWscvr      SunPlex Manager Root Components
application SUNWscvw      SunPlex Manager Core Components
```

2. Verify that the `root` account on each node has a password set.
3. From the administrative console system, start a Web browser and connect to the SunPlex Manager application on Node 1. Log in as user `root`.

```
# netscape https://node1:3000
```

Task – Navigating the Nodes Device Tree

Perform the following steps to familiarize yourself with the SunPlex Manager Nodes device tree features:

1. Expand all of the SunPlex Manager device tree nodes.
2. Select one of your nodes in the device tree Nodes section.
3. Log in to the selected node using the telnet button at the top of the status window.
4. Log out (exit) to quit the telnet window.



Note – Do not use the `telnet` window to shut down the cluster because you are disconnected before the shutdown has completed. You might miss important status or error information. It is best to use the `cconsole` windows to shut down a cluster.

5. Select each entry in the Nodes section, and examine the associated status information.

Task – Navigating the Resource Groups Device Tree

Perform the following steps to familiarize yourself with the SunPlex Manager Resource Groups device tree features:

1. Select the Resource Groups device tree node.
2. Select one of the resource groups in the device tree, such as the `sa-rg` resource group.
3. In the status area, select one of the resources.
4. Examine the resource general properties information.
5. Select each entry in the Resource Groups section, and examine the associated status information.



Note – Do not perform any resource group actions at this time.

6. Select the Resource Groups device tree node again, and then select Resource Group Topology from the View button menu.

Task – Navigating the Transports Device Tree

Perform the following steps to familiarize yourself with the SunPlex Manager Transports device tree features:

1. Select the Transports device tree node.
2. Select one the Transport Adapters in the status window.
3. Notice the adapter properties, such as the `Dlpi heartbeat timeout`. You cannot modify these properties using the SunPlex Manager application.



Caution – Do not perform any transport actions. Induced errors can be difficult to correct. It is good practice to record the `scinstall` command equivalent during an initial Sun Cluster installation.

4. Select Transport Topology from the View button menu.

Task – Navigating the Global Devices Device Tree

Perform the following steps to familiarize yourself with the SunPlex Manager Transports device tree features:

1. Select the Global Devices device tree node.
2. Select Device Group Topology from the View button menu.
3. Select one of the device groups in the Global Devices tree, such as the `webdg` device group.



Caution – Normally, you should not perform direct operations on device groups. Usually, you perform most operations as the resource group level. During normal cluster operation, device groups are managed by the Sun Cluster resource management software.

Task – Navigating the Quorum Device Tree

Perform the following steps to familiarize yourself with the SunPlex Manager Quorum device tree features:

1. Select the Quorum device tree node.
2. Select a quorum device in the Quorum device tree.

Task – Removing a Resource Group

Perform the following steps to remove and recreate the Apache scalable data service:



Note – You might have to reload the device tree frame and reselect resources during this procedure.

1. Select the Resource Groups device tree node.
2. Take the Apache data service resource groups offline:
 - a. Select the `web-rg` resource group in the device tree.
 - b. Perform the Take Group Offline action.
 - c. Select the `sa-rg` resource group in the device tree.
 - d. Perform the Take Group Offline action.
3. Disable the `apache-res` and `clustername-web` resources:
 - a. Select the `apache-res` resource in the device tree.
 - b. Perform the Disable Resource action.
 - c. Perform the Delete Resource action.
 - d. Select the `clustername-web` resource in the device tree.
 - e. Perform the Disable Resource action.
 - f. Perform the Delete Resource action.
4. Remove the `sa-rg` and `web-rg` resource groups:
 - a. Select the `web-rg` resource group in the device tree.
 - b. Perform the Delete Group action.
 - c. Select the `sa-rg` resource group in the device tree.
 - d. Perform the Delete Group action.

Task – Creating a Resource Group

Perform the following steps to recreate the Apache scalable data service:

1. Select the Resource Groups device tree node.
2. Create a failover resource group named `sa-rg` by performing the following steps:
 - a. Select the Create New Group action.
 - b. Enter the resource group name (`sa-rg`).
 - c. Select both nodes in the Primaries List.
 - d. Set Desired Primaries to 1.
 - e. Set Maximum Primaries to 1.
 - f. Select Create Resource Group.
3. Add the SharedAddress resource to the `sa-rg` failover group:
 - a. Select `sa-rg` in the device tree.
 - b. Perform the Create New Resource action.
 - c. Select the HA Shared Address Resource Type from the Resource Type menu.
 - d. Select Continue
 - e. Select `sa-rg` from the Resource Group to Hold Resource menu.
 - f. Enter `clustername-web` in the HostnameList box.
 - g. Select Create Resource
 - h. Select the `clustername-web` resource in the device tree and perform the Enable Resource action.
 - i. Select `sa-rg` in the device tree, and perform the Bring Online action.
4. Create an Apache scalable resource group named `web-rg` by performing the following steps:
 - a. Perform the Create New Group action.
 - b. Enter the resource group name (`web-rg`).
 - c. Select both nodes in the Primaries List.
 - d. Set Desired Primaries to 2.

- e. Set Maximum Primaries to 2.
 - f. Select the Configure Additional Properties button.
 - g. Select the sa-rg group in the Dependencies section.
 - h. Select Create Resource Group.
5. Add the `SUNW.apache` resource to the `web-rg` group:
 - a. Select Resource Groups in the device tree.
 - b. Select `web-rg` in the device tree.
 - c. Perform the Create New Resource action.
 - d. Select the Apache Web Server on Sun Cluster resource type.
 - e. Select the `clustername-web` from the Associated Network Resource menu.
 - f. Select the Configure Additional Properties button.
 - g. Select the Continue button.
 - h. Select `web-rg` under Resource Group To Hold Resource.
 - i. Type `apache-res` in the Name for New Resource box.
 - j. Set `Confdir_list` to `/etc/apache`.
 - k. Set `Bin_dir` to `/usr/apache/bin`.
 - l. Select the Create Resource button.
 - m. Select the `apache-res` resource in the device tree, and perform the Enable Resource action.
 - n. Select `web-rg` in the device tree, and perform the Bring Online action.
6. Verify that the resources enabled. If necessary, select each resource in the device tree, and perform the Enable Resource action.
7. Verify that the resource groups are online. If necessary, select each resource group in the device tree, and perform the Bring Online action.
8. Log out of SunPlex Manager.
9. Test the Apache Web Server data service again using the `http://clustername-web/cgi-bin/test-apache.cgi` address.

Exercise Summary

Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace

Check Your Progress

Check that you are able to accomplish or answer the following:

- ☐ Install and configure the SunPlex management software
- ☐ List the main features of SunPlex Manager
- ☐ Use SunPlex Manager to verify cluster status
- ☐ Use SunPlex Manager to remove a resource group
- ☐ Use SunPlex Manager to configure a resource group

Think Beyond

Why might you need to use alternative management techniques in place of SunPlex Manager?

Sun Cluster Administration Workshop

Objectives

Upon completion of this module, you should be able to:

- Install and configure the Sun Cluster host software
- Install and configure VERITAS Volume Manager
- Create network automatic failover (NAFO) groups
- Install, configure, and test the Sun Cluster HA for NFS failover data service
- Install, configure, and test the Sun Cluster HA for Apache scalable data service

Relevance

Present the following question to stimulate the students and get them thinking about the issues and topics presented in this module. While they are not expected to know the answer to this question, the answer should be of interest to them and inspire them to learn the material presented in this module.



Discussion – The following question is relevant to understanding the content of this module:

- What is the best way to become comfortable with new knowledge?

Additional Resources



Additional resources – The following references provide additional information on the topics described in this module:

- *Sun™ Cluster 3.0 07/01 Installation Guide*, part number 806-7069
- *Sun™ Cluster 3.0 07/01 Hardware Guide*, part number 806-7070
- *Sun™ Cluster 3.0 07/01 Data Services Installation and Configuration Guide*, part number 806-7071
- *Sun™ Cluster 3.0 07/01 Data Service Developers Guide*, part number 806-7072
- *Sun™ Cluster 3.0 07/01 System Administration Guide*, part number 806-7073
- *Sun™ Cluster 3.0 07/01 Concepts*, part number 806-7074
- *Sun™ Cluster 3.0 07/01 Error Message Guide*, part number 806-7076
- *Sun™ Cluster 3.0 07/01 Release Notes*, part number 806-7078

Sun Cluster Administration Workshop Introduction

This workshop provides practice using skills learned from this course and its prerequisites. It is designed to take an entire day to complete. To ensure adequate time, it is best to begin the software installation steps (Steps 1 and 2) at the end of the day preceding the workshop.



Note – It is less important to complete all the steps in the workshop than it is to understand the steps you do complete. Accordingly, work at a pace that promotes comprehension for your team, and complete as many steps as time permits.

This workshop also requires students to work in teams. Each team of two persons works on a single cluster system.

As you work through the steps, take time to understand the solution to each step before proceeding to the next. The goal of this workshop is to promote understanding of administration concepts through practice.

The Configuration Steps section and the Configuration Solutions section are complementary resources. The Configuration Steps describe what to accomplish on which systems, but these steps do not tell you how. The Configuration Solutions section offers general advice on how to accomplish the required tasks.

Consider dividing responsibility for the information these sections contain among members of your team.



Note – During this exercise, when you see italicized names, such as *IPaddress*, *enclosure_name*, *node1*, or *clustername* imbedded in a command string, substitute the names appropriate for your cluster.

System Preparation

Your lab must have a JumpStart server available to reinstall the Solaris 8 Operating Environment on your cluster nodes.



Note – Leave your administration console systems as they are.

Configuration Steps

Perform the following steps to complete the Sun Cluster administration workshop:

1. Halt both of your cluster nodes, and perform a JumpStart boot on them.
2. After the cluster nodes complete their JumpStart, verify that their boot disks have the configuration shown in Table 13-1:

Table 13-1 Boot Disk Configuration

Slice	Use	Size
0	/	Remainder
1	swap	Twice memory
2	backup	Entire disk
4	/global	100 Mbytes
7	unassigned	10 Mbytes

3. Set up the root environment on both cluster hosts, including:
 - The `/.profile` files
 - The `/.rhosts` files
 - The `/etc/default/login` files
 - The `/etc/hosts` files
4. Install the Sun Cluster software on Node 1 of your cluster.
5. Install the Sun Cluster software on Node 2 of your cluster.
6. Select a suitable disk ID (DID) quorum disk on Node 1.
7. On Node 1, use the `scsetup` utility to configure a quorum device and reset the `installmode` flag.
8. Configure the `/etc/inet/ntp.conf` file on both nodes.
9. Make sure the `/etc/nsswitch.conf` file has the correct `hosts` entry to search files before `nis`.
10. Verify the general status and operation of your cluster.

11. Use the `scvxinstall` utility on one node at a time to perform the following functions:
 - a. Disable Dynamic Multipathing (DMP).
 - b. Install the VERITAS Volume Manager software packages and a license key.
 - c. Set the `vxio` driver major numbers.
 - d. Set the `rootdg` disk group minor device numbers.
 - e. Encapsulate the root disk.
 - f. Correct `/global` entries in the `/etc/vfstab` file.
12. Select two disks in each storage array for use in mirrored volumes. Record the logical paths to the disks (`c3t4d0`).

	Array A	Array B
<code>nfsg</code> disks:	<code>disk01:</code> _____	<code>disk02:</code> _____
<code>webdg</code> disks:	<code>disk03:</code> _____	<code>disk04:</code> _____

13. Configure demonstration disk groups and volumes for use later.
 - a. On Node 1, create the `nfsg` disk group with two disks.
 - b. On Node 1, create a 500-Mbyte mirrored volume in the `nfsg` disk group that is named `vol-01`.
 - c. On Node 2, create the `webdg` disk group with two disks.
 - d. On Node 2, create a 500-Mbyte mirrored volume in the `webdg` disk group that is named `vol-01`.
14. Register the demonstration disk groups.
 - a. On Node 1, register the `nfsg` and `webdg` disk groups. Reverse the node list for the `webdg` disk group.
15. Create and mount the `/global/nfs` file system on Node 1.
 - a. On Node 1, initialize the `/global/nfs` file system.
 - b. On both nodes add `/etc/vfstab` mount entries for the `/global/nfs` file system.
 - c. On Node 1, mount the `/global/nfs` file system.

16. Create and mount the `/global/web` file system on Node 2.
 - a. On Node 2, initialize the `/global/web` file system.
 - b. On both nodes, add `/etc/vfstab` mount entries for the `/global/web` file system.
 - c. On Node 2, mount the `/global/web` file system.
17. Practice migrating the disk device groups between nodes.
18. Verify the electrically erasable programmable read-only memory (EEPROM) `local-mac-address?` variable is set to `false` on both nodes.
19. Create a NAFO group on each node.
20. Install the Sun Cluster HA for NFS data service software on *both* nodes.
21. Verify the `hosts` line in the `/etc/nsswitch.conf` file.
22. Resolve the Sun Cluster HA for NFS logical host name in the `/etc/hosts` files on both cluster nodes.
23. Create the Sun Cluster HA for NFS administrative directory structure and `dfstab.nfs-res` file on Node 1.
24. Create the `/global/nfs/data` directory.
25. From Node 1, run the `scsetup` utility, and use the Resource Group option to configure the Sun Cluster HA for NFS data service.
 - a. Create a failover group.
 - b. Name the failover group `nfs-rg`.
 - c. Make `node1` the preferred node.
 - d. Make the `Pathprefix` directory `/global/nfs/admin`.
 - e. Add the `LogicalHostname (clustername-nfs)` network resource.
 - f. Add the `SUNW.nfs (nfs-res)` and `SUNW.HAStorage (has-res)` resources.
 - g. Set the `ServicePaths (/global/nfs)` extension property.
 - h. Bring the resource group online.
26. Verify that clients can access the `/global/nfs/data` file system.

27. Test the `/global/nfs/data` file system while migrating the NFS data service between nodes.
28. Install the Sun Cluster HA for Apache data service software on *both* nodes.
29. Disable the standard Apache run control scripts.
30. Resolve the Sun Cluster HA for Apache logical host name in the `/etc/hosts` files on both cluster nodes.
31. Configure the Apache `httpd.conf` file and the `htdocs` and `cgi-bin` directories:
 - a. Create the `httpd.conf` file.
 - b. Edit the `httpd.conf` file and change path names.
 - c. Create global `htdocs` and `cgi-bin` directories.
 - d. Copy sample files into the `htdocs` and `cgi-bin` directories.
 - e. Copy the `test-cluster.cgi` file into the `cgi-bin` directory.
32. From Node 1, run the `scsetup` utility and configure the `sa-rg` failover resource group for the Sun Cluster HA for Apache data service with the following characteristics:
 - It is a failover group.
 - The preferred host is `node1`.
 - It does not use a configuration data file system.
 - It uses the shared address network resource.
 - It uses a logical host name `clustername-web`.
 - It uses no data service resources.
33. From Node 1, run the `scsetup` utility and configure the `web-rg` scalable resource group for the Sun Cluster HA for Apache data service with the following characteristics:
 - It is dependent on the `sa-rg` resource group.
 - It does not use a configuration data file system.
 - It uses the `SUNW.apache` resource type named `apache-res`.
 - It uses the shared address resource named `clustername-web`.
 - It uses a data service port numbered 80.
 - It uses the extension `Confdir_list=/etc/apache`.

- It uses the extension `Bin_dir=/usr/apache/bin`.
- It uses no additional data service resources.



Note – If there are any validation errors, you must verify that the `sa-rg` resource group *and* its resources are enabled and online. There are separate actions for each component.

34. Test the Apache installation by using a Web browser on the administration workstation and connecting to:
`http://clustername-web/cgi-bin/test-cluster.cgi`.

Configuration Solutions

This section provides configuration advice for completing the workshop. The information in this section is intended to support the tasks described in the Configuration Steps section of the workshop.



Note – The step numbers that follow match those in the Configuration Steps section.

1. Installing the Solaris Operating Environment both nodes might take as long as 60 minutes, depending on network traffic in the lab.

It is a good idea to start the JumpStart operation late Thursday afternoon so that the systems are ready Friday morning.

Make sure the JumpStart operation has started to load software packages before leaving.

Initiate the JumpStart operation on each node using:

```
ok boot net - install
```

2. Use the `prtconf` command to assess memory size, and use the `format` utility to examine the boot disk partitions.

Slice 2 (backup) must always remain the entire disk. Never alter slice 2.

The `/globaldevices` partition must be at least 100 Mbytes in size.

The small slice 7 partition for Solstice DiskSuite state databases (replicas) is not needed unless you are planning to install Solstice DiskSuite instead of VERITAS Volume Manager.

For this exercise, the swap partition size is not particularly important. In a production environment, it could be critical.

3. Setting up the environment early saves a lot of frustration looking for files and manual pages.

The `/.profile` files should contain the following variables:

```
PATH=$PATH:/usr/cluster/bin:/etc/vx/bin
```

```
MANPATH=$MANPATH:/usr/cluster/man:/usr/share/man:/opt/VRTSvxvm/man:/opt/VRTSvmsa/man
```

```
TERM=vt220
```

```
export PATH MANPATH TERM
```

On both nodes, create a `.rhosts` file in the root directory. Edit the file, and add a single line with a plus (+) sign.

On both cluster nodes, edit the `/etc/default/login` file and comment out the `CONSOLE=/dev/console` line.

Edit the `/etc/hosts` file on the administrative workstation and all cluster nodes and add the Internet Protocol (IP) addresses and host names of the administrative workstation and cluster nodes.



Note – The `.rhosts` and `/etc/default/login` file modifications used here can be a security risk in some environments. They are used here to simplify some of the lab exercises.

4. Run `scinstall` from the `SunCluster_3.0/Tools` directory.
 - Use option 1, Establish a new cluster.
 - Do not use Data Encryption Standard (DES) authentication.
 - Accept default transport addresses.
 - Use the default `/globaldevices` file system (it must already exist).
 - Accept the automatic reboot as there are no Sun Cluster patches.
5. Use the same process as the first node except for:
 - Use option 2, Add this machine as a node in an established cluster.
6. Use the `scdidadm -L` command on Node 1 to assist in selecting a quorum disk.

Do not use a local disk such as the system boot disk.

You can use the quorum disk for other purposes such as data storage.

The number of quorum disks usually equals the number of cluster nodes minus 1 (N-1).

7. The `scsetup` utility behaves differently the first time it is run on a newly installed cluster.

The quorum device path must be in a Disk ID device format (`d4`).

8. You must remove all private node name entries that do not apply to your cluster. For a two-node cluster, remove the following:

```
peer clusternode3-priv
peer clusternode4-priv
peer clusternode5-priv
peer clusternode6-priv
peer clusternode7-priv
peer clusternode8-priv
```

9. Make sure the `scinstall` program modified the `hosts` entry in the `/etc/nsswitch.conf` file to read as follows:

```
hosts:          cluster files nis [NOTFOUND=return]
```

10. Use the following commands to verify general cluster status and operation:

- `scstat -q`
- `scdidadm -L`
- `scconf -p`
- `sccheck`
- `scshutdown -y -g 15`

11. The `scvxinstall` utility requires little information. If you are installing the VERITAS Volume Manager software on a system that uses non-array disk storage, such as a Sun StorEdge MultiPack enclosure, you must provide a license key to enable the basic VERITAS Volume Manager functionality.

12. Make sure the disks for each disk group are in different storage arrays. Mirroring across storage arrays is a general cluster requirement.

13. Use the following command summary as a guideline for creating the demonstration disk groups and volumes.

```
# vxdiskadd disk01 disk02 (nfsdg)
# vxassist -g nfsdg make vol-01 500m layout=mirror
# vxdiskadd disk03 disk04 (webdg)
# vxassist -g webdg make vol-01 500m layout=mirror
```


14. Use the following command summary as a guideline to register the new disk groups so they become disk device groups.

```
# scconf -a -D type=vxvm,name=nfsdg,nodelist=node1:node2
# scconf -a -D type=vxvm,name=webdg,nodelist=node2:node1
```

15. Use the following command summary as a guideline to create and mount the /global/nfs file system on Node 1.

```
# newfs /dev/vx/rdisk/nfsdg/vol-01
# mkdir /global/nfs (on both nodes)
# vi /etc/vfstab (on both nodes)
/dev/vx/dsk/nfsdg/vol-01 /dev/vx/rdisk/nfsdg/vol-01 \
/global/nfs ufs 2 yes global,logging
# mount /global/nfs
```

16. Use the following command summary as a guideline to create and mount the /global/web file system on Node 2.

```
# newfs /dev/vx/rdisk/webdg/vol-01
# mkdir /global/web (on both nodes)
# vi /etc/vfstab (on both nodes)
/dev/vx/dsk/webdg/vol-01 /dev/vx/rdisk/webdg/vol-01 \
/global/web ufs 2 yes global,logging
# mount /global/web
```

17. Use the scswitch command as shown to migrate the disk device groups between nodes. You can run the commands from either node.

```
# scswitch -z -D nfsdg -h node2
# scswitch -z -D webdg -h node1
# scswitch -z -D nfsdg -h node1
# scswitch -z -D webdg -h node2
```

18. Use the following command summary to verify the EEPROM local-mac-address? variable is set to false on both nodes and, if necessary, change the value and reboot the nodes.

```
# eeprom | grep mac
# eeprom local-mac-address?=false
# init 6
```

19. Use the following example to create a NAFO group on each node.

```
# pnmset
In the following, you will be prompted to do configuration
for network adapter failover
Do you want to continue ... [y/n]: y
How many NAFO groups to configure [1]: 1
Enter NAFO group number [0]: 2
Enter space-separated list of adapters in nafo2: qfe0 qfe1
Checking configuration of nafo2:
Testing active adapter qfe0...
Testing adapter qfe1...
NAFO configuration completed
```

20. Install the Sun Cluster HA for NFS data service software on *both* nodes by running the `scinstall` utility. Use option 4.

The `scinstall` utility is on both nodes in the `/usr/cluster/bin` directory.

21. Verify the `hosts` line in the `/etc/nsswitch.conf` file. If necessary, correct it to read:

```
hosts: cluster files nis
```

22. Add an entry to the `/etc/hosts` file on each cluster node and the administrative workstation for the logical host name resource `clustername-nfs`. Substitute the IP address supplied by your instructor.

```
clustername-nfs IP_address
```

23. Use the following command sequence to create the administrative directory and the `dfstab.nfs-res` file for the NFS resource.

```
# cd /global/nfs
# mkdir admin
# cd admin
# mkdir SUNW.nfs
# cd SUNW.nfs
# vi dfstab.nfs-res
share -F nfs -o rw -d "Home Dirs" /global/nfs/data
```

24. Use the following commands to create the `/global/nfs/data` directory.

```
# cd /global/nfs
# mkdir /global/nfs/data
# chmod 777 /global/nfs/data
```

25. The following is the manual command sequence to configure the Sun Cluster HA for NFS data service. You can use this example to verify the `scsetup` utility command line output.

```
# scrgadm -a -t SUNW.nfs
# scrgadm -a -t SUNW.HAStorage
# scrgadm -a -g nfs-rg -h node1,node2 \
-y Pathprefix=/global/nfs/admin
# scrgadm -a -L -g nfs-rg -l clustername-nfs
# scrgadm -a -j has-res -g nfs-rg \
-t SUNW.HAStorage \
-x ServicePaths=/global/nfs \
-x AffinityOn=True
# scrgadm -a -j nfs-res -g nfs-rg -t SUNW.nfs \
-y Resource_dependencies=has-res
# scswitch -Z -g nfs-rg
# scstat -g
```

26. Use the following command from the administration workstation to verify client access to the `/global/nfs/data` file system.

```
# ls -l /net/clustername-nfs/global/nfs/data
```

27. Run the `Scripts/test.nfs` file on the administration workstation from the `root` directory while switching the NFS data service between nodes.

```
# scswitch -z -h node2 -g nfs-rg
# scswitch -z -h node1 -g nfs-rg
```

28. Install the Sun Cluster HA for Apache data service software on *both* nodes by running the `scinstall` utility. Use option 4.

The `scinstall` utility is on both node in the `/usr/cluster/bin` directory.

29. Use the following command sequence to disable the standard Apache run control scripts. Apache startup must be controlled by the data service software and not the system startup scripts.

- a. List the Apache run control scripts.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/K16apache
/etc/rc1.d/K16apache
/etc/rc2.d/K16apache
/etc/rc3.d/S50apache
/etc/rcS.d/K16apache
```

- b. Rename the Apache run control scripts.

```
# mv /etc/rc0.d/K16apache /etc/rc0.d/k16apache
# mv /etc/rc1.d/K16apache /etc/rc1.d/k16apache
# mv /etc/rc2.d/K16apache /etc/rc2.d/k16apache
# mv /etc/rc3.d/S50apache /etc/rc3.d/s50apache
# mv /etc/rcS.d/K16apache /etc/rcS.d/k16apache
```

- c. Verify that all the Apache-related scripts have been renamed.

```
# ls -l /etc/rc?.d/*apache
/etc/rc0.d/k16apache
/etc/rc1.d/k16apache
/etc/rc2.d/k16apache
/etc/rc3.d/s50apache
/etc/rcS.d/k16apache
```

30. Create an entry in `/etc/hosts` for the shared address you will be configuring with the Apache Web server. In addition, create the entry on the administration workstation. Substitute the IP address supplied by your instructor.

```
clustername-web IP_address
```

31. Use the following command sequence to configure the Apache application files.

```
# cp /etc/apache/httpd.conf-example \
/etc/apache/httpd.conf

# vi /etc/apache/httpd.conf
```

Change from:

```
#ServerName new.host.name
DocumentRoot "/var/apache/htdocs"
<Directory "/var/apache/htdocs">
ScriptAlias /cgi-bin/ "/var/apache/cgi-bin/"
<Directory "/var/apache/cgi-bin">
```

To:

```
ServerName clustername-web (Uncomment the line)
DocumentRoot "/global/web/htdocs"
<Directory "/global/web/htdocs">
ScriptAlias /cgi-bin/ "/global/web/cgi-bin"
<Directory "/global/web/cgi-bin">
```

```
# mkdir /global/web/htdocs
# mkdir /global/web/cgi-bin

# cp ./test-cluster.cgi /global/web/cgi-bin
# chmod 755 /global/web/cgi-bin/test-cluster.cgi
```

32. The following is the command-line sequence to configure the sa-rg resource group for the HA for Apache data service.

```
# scrgadm -a -t SUNW.apache
# scrgadm -a -g sa-rg -h node1,node2
# scrgadm -a -S -g sa-rg -l clustername-web
```

33. The following is the command-line sequence to configure the web-rg resource group for the HA for Apache data service.

```
# scrgadm -a -g web-rg -y Maximum primaries=2 \
-y Desired primaries=2 -y RG_dependencies=sa-rg

# scrgadm -a -j apache-res -g web-rg \
-t SUNW.apache -x Confdir_list=/etc/apache \
-x Bin_dir=/usr/apache/bin \
-y Scalable=TRUE \
-y Network_Resources_Used=clustername-web

# scswitch -Z -g sa-rg
# scswitch -Z -g web-rg
# scstat -g
```



Note – If there are any validation errors, you must verify that the `sa-rg` resource group is online *and* its resources are enabled. There are separate actions for each component.

34. Perform the following steps to test the Apache installation:
 - a. Connect to the Web server using the browser on the administrator workstation using `http://clustername-web/cgi-bin/test-cluster.cgi`.
 - b. Repeatedly hit the refresh button on the browser. The `test-cluster.cgi` script displays the actual node name that is servicing the request. It might take several iterations before the packet is distributed to a new node.

Exercise Summary

Take a few minutes to discuss what experiences, issues, or discoveries you had during the lab exercises.

Manage the discussion based on the time allowed for this module, which was provided in the “About This Course” module. If you do not have time to spend on discussion, then just highlight the key concepts students should have learned from the lab exercise.

- **Experiences**

Ask students what their overall experiences with this exercise have been. Go over any trouble spots or especially confusing areas at this time.

- **Interpretations**

Ask students to interpret what they observed during any aspect of this exercise.

- **Conclusions**

Have students articulate any conclusions they reached as a result of this exercise experience.

- **Applications**

Explore with students how they might apply what they learned in this exercise to situations at their workplace

Check Your Progress

Check that you are able to accomplish or answer the following:

- ☐ Install and configure the Sun Cluster host software
- ☐ Install and configure VERITAS Volume Manager
- ☐ Create NAFO groups
- ☐ Install, configure, and test the Sun Cluster HA for NFS failover data service
- ☐ Install, configure, and test the Sun Cluster HA for Apache scalable data service

Think Beyond

How useful might a *cookbook* document be to you in your workplace?

Cluster Configuration Forms

This appendix contains forms that you can use to record cluster configuration information. The following types of worksheets are in this appendix:

- Cluster and Node Names Worksheet
- Cluster Interconnect Worksheet
- Public Networks Worksheet
- Local Devices Worksheet
- Local File System Layout Worksheet
- Disk Device Group Configurations Worksheet
- Volume Manager Configurations Worksheet
- Metadevices Worksheet
- Failover Resource Types Worksheet
- Failover Resource Groups Worksheet
- Network Resource Worksheet
- HA Storage Application Resources Worksheet
- NFS Application Resources Worksheet
- Scalable Resource Types Worksheet
- Scalable Resource Groups Worksheet
- Shared Address Resource Worksheet

Cluster and Node Names Worksheet

Cluster name _____

Private network IP address _____ (default: 172.16.0.0)

Private network mask _____ (default: 255.255.0.0)

Nodes

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Node name _____

Private host name _____

Cluster Interconnect Worksheet

Adapters

Node name _____

Adapter Name	Transport Type

Node name _____

Adapter Name	Transport Type

Node name _____

Adapter Name	Transport Type

Node name _____

Adapter Name	Transport Type

Cabling

Draw lines between cable endpoints

Junctions

Junction name _____

Junction type _____

Port Number	Description (optional)

Junction name _____

Junction type _____

Port Number	Description (optional)

Public Networks Worksheet

Node name _____	Node name _____
Primary host name _____	Primary host name _____
Network name _____	Network name _____
Adapter names _____	Adapter names _____
NAFO group number: nafo_____	NAFO group number: nafo_____
Secondary host name _____	Secondary host name _____
Network name _____	Network name _____
Adapter names _____	Adapter names _____
NAFO group number : nafo_____	NAFO group number: nafo_____
Secondary host name _____	Secondary host name _____
Network name _____	Network name _____
Adapter names _____	Adapter names _____
NAFO group number : nafo_____	NAFO group number: nafo_____
Secondary host name _____	Secondary host name _____
Network name _____	Network name _____
Adapter names _____	Adapter names _____
NAFO group number : nafo_____	NAFO group number: nafo_____

Local Devices Worksheet

Node name _____

Local disks

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Other local devices

Device type _____ Name _____

Device type _____ Name _____

Device type _____ Name _____

Device type _____ Name _____

Node name _____

Local disks

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Disk name _____ Size _____

Other local devices

Device type _____ Name _____

Device type _____ Name _____

Device type _____ Name _____

Device type _____ Name _____

Local File System Layout Worksheet

Node name _____

Mirrored root

Volume Name	Component	Component	File System	Size
			/	
			/usr	
			/var	
			/opt	
			swap	
			/globaldevices	

Non-mirrored root

Device Name	File System	Size
	/	
	/usr	
	/var	
	/opt	
	swap	
	/globaldevices	

Disk Device Group Configurations Worksheet

Volume manager: _____			
Disk group/diskset name _____			
Node names (1)	(2)	(3)	(4)
(5)	(6)	(7)	(8)
Ordered priority? <input type="checkbox"/> Yes <input type="checkbox"/> No			
Maximum number of secondaries _____			
Failback? <input type="checkbox"/> Yes <input type="checkbox"/> No			
 Disk group/diskset name _____			
Node names (1)	(2)	(3)	(4)
(5)	(6)	(7)	(8)
Ordered priority? <input type="checkbox"/> Yes <input type="checkbox"/> No			
Maximum number of secondaries _____			
Failback? <input type="checkbox"/> Yes <input type="checkbox"/> No			
 Disk group/diskset name _____			
Node names (1)	(2)	(3)	(4)
(5)	(6)	(7)	(8)
Ordered priority? <input type="checkbox"/> Yes <input type="checkbox"/> No			
Maximum number of secondaries _____			
Failback? <input type="checkbox"/> Yes <input type="checkbox"/> No			

Sun™ Cluster 3.0 Administration
Copyright 2001 Sun Microsystems, Inc. All Rights Reserved. Enterprise Services, Revision B

Cluster Configuration Forms
Copyright 2001 Sun Microsystems, Inc. All Rights Reserved. Enterprise Services, Revision B

Failover Resource Types Worksheet

Indicate the nodes on which the resource type will run (other than logical host or shared address).				
Resource type name _____				
Node names	_____	_____	_____	_____
	_____	_____	_____	_____
 Resource type name _____				
Node names	_____	_____	_____	_____
	_____	_____	_____	_____

Failover Resource Groups Worksheet

Resource group name _____

(Must be unique within the cluster.)

Function of this resource group **Contains the NFS resources** _____

Failback? ☐ Yes ☒ No

(Will this resource group switch back to the primary node, after the primary node has failed and been restored?)

Node names (1) _____ (2) _____ (3) _____ (4) _____
(ordered list)
(5) _____ (6) _____ (7) _____ (8) _____

(Indicate the cluster nodes that might host this resource group. The first node in this list should be the primary, with others being the secondaries. The order of the secondaries will indicate preference for becoming primaries.)

Disk device groups upon which this resource group depends _____

(If the resources in this resource group need to create files for administrative purposes, include the subdirectory they should use.)

/global/nfs/admin

HA Storage Application Resources Worksheet

Resource name _____

Resource group name _____

Resource type:

☐ Logical host name ☐ Shared address☒ Data service/other

Host names used _____

Network name _____

Adapter or NAFO group:

Node name	Adapter/NAFO group name

Resource type name **SUNW.HAStorage**

Dependencies _____

Extension properties:

Name	Value

NFS Application Resources Worksheet

Resource name _____

Resource group name _____

Resource type:

☐ Logical host name☐ Shared address☒ Data service/other

Host names used _____

Network name _____

Adapter or NAFO group:

Node name	Adapter/NAFO group name

Resource type name **SUNW.NFS**

Dependencies _____

Extension properties:

Name	Value

Scalable Resource Types Worksheet

<p>Indicate the nodes on which the resource type will run (other than logical host or shared address)</p>				
Resource type name	SUNW.apache			
Node names	_____	_____	_____	_____
	_____	_____	_____	_____
Resource type name	_____			
Node names	_____	_____	_____	_____
	_____	_____	_____	_____

Scalable Resource Groups Worksheet

<p>Resource group name <u>web-rg</u></p> <p><i>(Must be unique within the cluster.)</i></p> <p>Function of this resource group Contains the Web server resources</p> <p>Maximum number of primaries _____</p> <p>Desired number of primaries _____</p> <p>Failback? <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No</p> <p><i>(Will this resource group switch back to the primary node, after the primary node has failed?)</i></p> <p>Node names (1) _____ (2) _____ (3) _____ (4) _____ <i>(ordered list)</i> (5) _____ (6) _____ (7) _____ (8) _____</p> <p><i>(Indicate the cluster nodes that might host this resource group. The first node in this list should be the primary, with others being the secondaries. The order of the secondaries will indicate preference for becoming primaries.)</i></p> <p>Dependencies <u>sa-rg</u></p> <p><i>(Does this resource depend upon another resource group.)</i></p>
<p>Resource group name <u>sa-rg</u></p> <p><i>(Must be unique within the cluster.)</i></p> <p>Function of this resource group Contains the shared address resources</p> <p>Maximum number of primaries <u>1</u></p> <p>Desired number of primaries <u>1</u></p> <p>Failback? <input type="checkbox"/> Yes <input checked="" type="checkbox"/> No</p> <p><i>(Will this resource group switch back to the primary node, after the primary node has failed)</i></p> <p>Node names (1) _____ (2) _____ (3) _____ (4) _____ <i>(ordered list)</i> (5) _____ (6) _____ (7) _____ (8) _____</p> <p><i>(Indicate the cluster nodes that may host this resource group. The first node in this list should be the primary, with others being the secondaries. The order of the secondaries will indicate preference for becoming primaries.)</i></p> <p>Dependencies _____</p>

Shared Address Resource Worksheet

Resource name _____Resource group name **sa-rg** _____

Resource type:

☐ Logical host name ☒ Shared address☐ Data service/other

Host names used _____

Network name _____

Adapter or NAFO group:

Node name	Adapter/NAFO group name

Resource type name _____

Dependencies _____

Extension properties:

Name	Value

Scalable Application Resource Worksheet

Resource name apache-resResource group name web-rg

Resource type:

☐ Logical host name ☐ Shared address☒ Data service/otherHost names used _____

Network name _____

Adapter or NAFO group:

Node name	Adapter/NAFO group name

Resource type name SUNW.apache

Dependencies _____

Extension properties:

Name	Value
-x Confdir_list	/etc/apac he
-x Bin_dir	/usr/apac he/bin
-y Scalable	TRUE
-y Network_Resources_Used	web- server

Configuring Multi-Initiator SCSI

This appendix contains information that can be used to configure multi-initiator Small Computer System Interface (SCSI) storage devices including the Sun StorEdge MultiPack and Sun StorEdge D1000 storage arrays.

Multi-Initiator Overview

This section applies only to SCSI storage devices and not to Fibre Channel storage used for the multihost disks.

In a standalone server, the server node controls the SCSI bus activities using the SCSI host adapter circuit connecting this server to a particular SCSI bus. This SCSI host adapter circuit is referred to as the *SCSI initiator*. This circuit initiates all bus activities for this SCSI bus. The default SCSI address of SCSI host adapters in Sun systems is 7.

Cluster configurations share storage between multiple server nodes. When the cluster storage consists of singled-ended or differential SCSI devices, the configuration is referred to as multi-initiator SCSI. As this terminology implies, more than one SCSI initiator exists on the SCSI bus.

The SCSI specification requires that each device on a SCSI bus has a unique SCSI address. (The host adapter is also a device on the SCSI bus.) The default hardware configuration in a multi-initiator environment results in a conflict because all SCSI host adapters default to 7.

To resolve this conflict, on each SCSI bus, leave one of the SCSI host adapters with the SCSI address of 7, and set the other host adapters to unused SCSI addresses. Proper planning dictates that these “unused” SCSI addresses include both currently and eventually unused addresses. An example of addresses unused in the future is the addition of storage by installing new drives into empty drive slots. In most configurations, the available SCSI address for a second host adapter is 6.

You can change the selected SCSI addresses for these host adapters by setting the `scsi-initiator-id` Open Boot Programmable read-only memory (PROM) property. You can set this property globally for a node or on a per-host-adapter basis. Instructions for setting a unique `scsi-initiator-id` for each SCSI host adapter are included in the chapter for each disk enclosure in the *SunTM Cluster 3.0 07/01 Hardware Guide*.

Installing a Sun StorEdge MultiPack Enclosure

This section provides the procedure for an initial installation of a Sun StorEdge MultiPack enclosure.

Use this procedure to install a Sun StorEdge MultiPack enclosure in a cluster prior to installing the Solaris Operating Environment and Sun Cluster software. Perform this procedure with the procedures in *SunTM Cluster 3.0 07/01 Installation Guide* and your server hardware manual.

1. Ensure that each device in the SCSI chain has a unique SCSI address.

The default SCSI address for host adapters is 7. Reserve SCSI address 7 for one host adapter in the SCSI chain. This procedure refers to the host adapter you choose for SCSI address 7 as the host adapter on the `second` node. To avoid conflicts, in Step 7, you change the `scsi-initiator-id` of the remaining host adapter in the SCSI chain to an available SCSI address. This procedure refers to the host adapter with an available SCSI address as the host adapter on the `first` node. Depending on the device and configuration settings of the device, either SCSI address 6 or 8 is usually available.



Caution – Even though a slot in the enclosure might not be in use, you should avoid setting `scsi-initiator-id` for the first node to the SCSI address for that disk slot. This precaution minimizes future complications if you install additional disk drives.

For more information, refer to the *OpenBootTM 3.x Command Reference Manual* and the labels inside the storage device.

2. Install the host adapters in the nodes that will be connected to the enclosure.

For the procedure on installing host adapters, refer to the documentation that shipped with your nodes.

3. Connect the cables to the enclosure, as shown in Figure B-1.

Make sure that the *entire* SCSI bus length to each enclosure is less than 6 meters. This measurement includes the cables to both nodes, as well as the bus length internal to each enclosure, node, and host adapter. Refer to the documentation that shipped with the enclosure for other restrictions regarding SCSI operation.

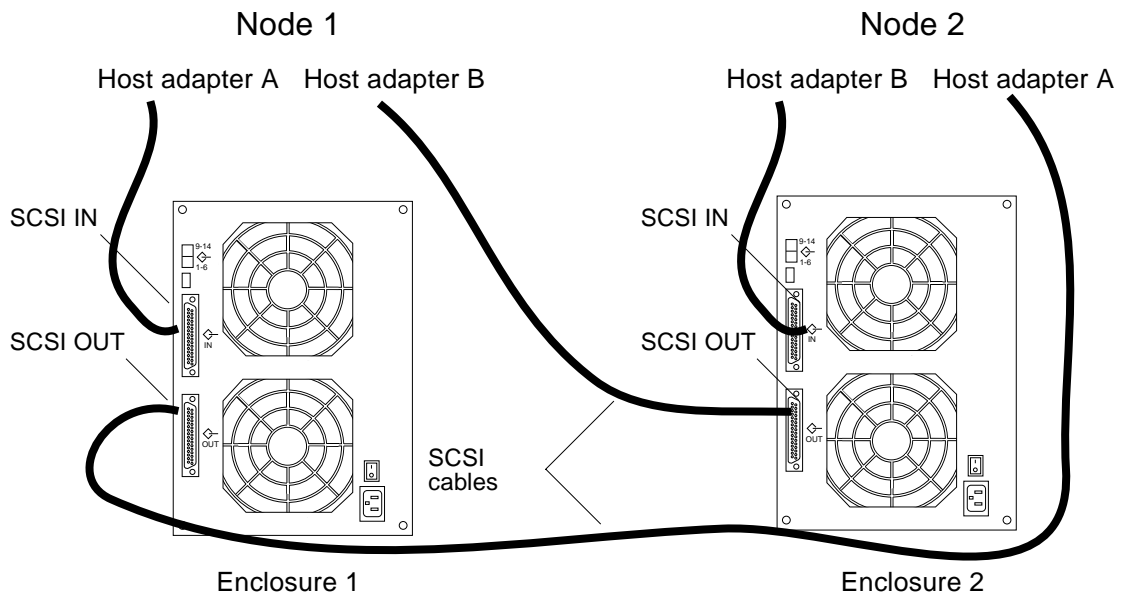


Figure B-1 Example of a Sun StorEdge MultiPack Enclosure Mirrored Pair

4. Connect the AC power cord for each enclosure of the pair to a different power source.
5. Without allowing the node to boot, power on the first node. If necessary, abort the system to continue with OpenBoot PROM Monitor tasks.
6. Find the paths to the host adapters.

ok **show-disks**

Identify and record the two controllers that will be connected to the storage devices, and record these paths. Use this information to change the SCSI addresses of these controllers in the `nvrnrc` script. Do not include the `/sd` directories in the device paths.

7. Edit the `nvrामrc` script to set the `scsi-initiator-id` for the host adapter on the first node.

For a list of `nvrामrc` editor and `nvedit` keystroke commands, see the "NVRAMRC Editor and NVEDIT Keystroke Commands" on page B-11.

The following example sets the `scsi-initiator-id` to 6. The OpenBoot PROM Monitor prints the line numbers (0:, 1:, and so on).

```
nvedit
0: probe-all
1: cd /sbus@1f,0/
2: 6 encode-int " scsi-initiator-id" property
3: device-end
4: cd /sbus@1f,0/SUNW,fas@2,8800000
5: 6 encode-int " scsi-initiator-id" property
6: device-end
7: install-console
8: banner <Control C>
ok
```



Caution – Insert exactly one space after the first double quotation mark and before `scsi-initiator-id`.

8. Store the changes.

The changes you make through the `nvedit` command are done on a temporary copy of the `nvrामrc` script. You can continue to edit this copy without risk. After you complete your edits, save the changes. If you are not sure about the changes, discard them.

- To discard the changes, type:

```
ok nvquit
ok
```

- To store the changes, type:

```
ok nvstore
ok
```

9. Verify the contents of the `nvrarc` script you created in Step 7.

```
ok printenv nvrarc
nvrarc = probe-all
cd /sbus@1f,0/
6 encode-int " scsi-initiator-id" property
device-end
cd /sbus@1f,0/SUNW,fas@2,8800000
6 encode-int " scsi-initiator-id" property
device-end
install-console
banner
ok
```

10. Instruct the OpenBoot PROM Monitor to use the `nvrarc` script.

```
ok setenv use-nvrarc? true
use-nvrarc? = true
ok
```

11. Without allowing the node to boot, power on the second node. If necessary, abort the system to continue with OpenBoot PROM Monitor tasks.

12. Verify that the `scsi-initiator-id` for the host adapter on the second node is set to 7.

```
ok cd /sbus@1f,0/SUNW,fas@2,8800000
ok .properties
scsi-initiator-id      00000007
...
```

13. Continue with the Solaris Operating Environment, Sun Cluster software, and volume management software installation tasks.

For software installation procedures, refer to the *SunTM Cluster 3.0 07/01 Installation Guide*.

Installing a Sun StorEdge D1000 Disk Array

This section provides the procedure for an initial installation of a Sun StorEdge D1000 disk array.

Use this procedure to install a Sun StorEdge D1000 disk array in a cluster prior to installing the Solaris Operating Environment and Sun Cluster software. Perform this procedure with the procedures in the *SunTM Cluster 3.0 07/01 Installation Guide* and your server hardware manual.

1. Ensure that each device in the SCSI chain has a unique SCSI address.

The default SCSI address for host adapters is 7. Reserve SCSI address 7 for one host adapter in the SCSI chain. This procedure refers to the host adapter you choose for SCSI address 7 as the host adapter on the `second` node. To avoid conflicts, in Step 7 you change the `scsi-initiator-id` of the remaining host adapter in the SCSI chain to an available SCSI address. This procedure refers to the host adapter with an available SCSI address as the host adapter on the `first` node. SCSI address 6 is usually available.



Caution – Even though a slot in the enclosure might not be in use, you should avoid setting the `scsi-initiator-id` for the first node to the SCSI address for that disk slot. This precaution minimizes future complications if you install additional disk drives.

For more information, refer to the *OpenBootTM 3.x Command Reference Manual* and the labels inside the storage device.

2. Install the host adapters in the node that will be connected to the disk array.

For the procedure on installing host adapters, refer to the documentation that shipped with your nodes.

3. Connect the cables to the disk arrays, as shown in Figure B-2.

Make sure that the *entire* bus length connected to each disk array is less than 25 meters. This measurement includes the cables to both nodes, as well as the bus length internal to each disk array, node, and the host adapter.

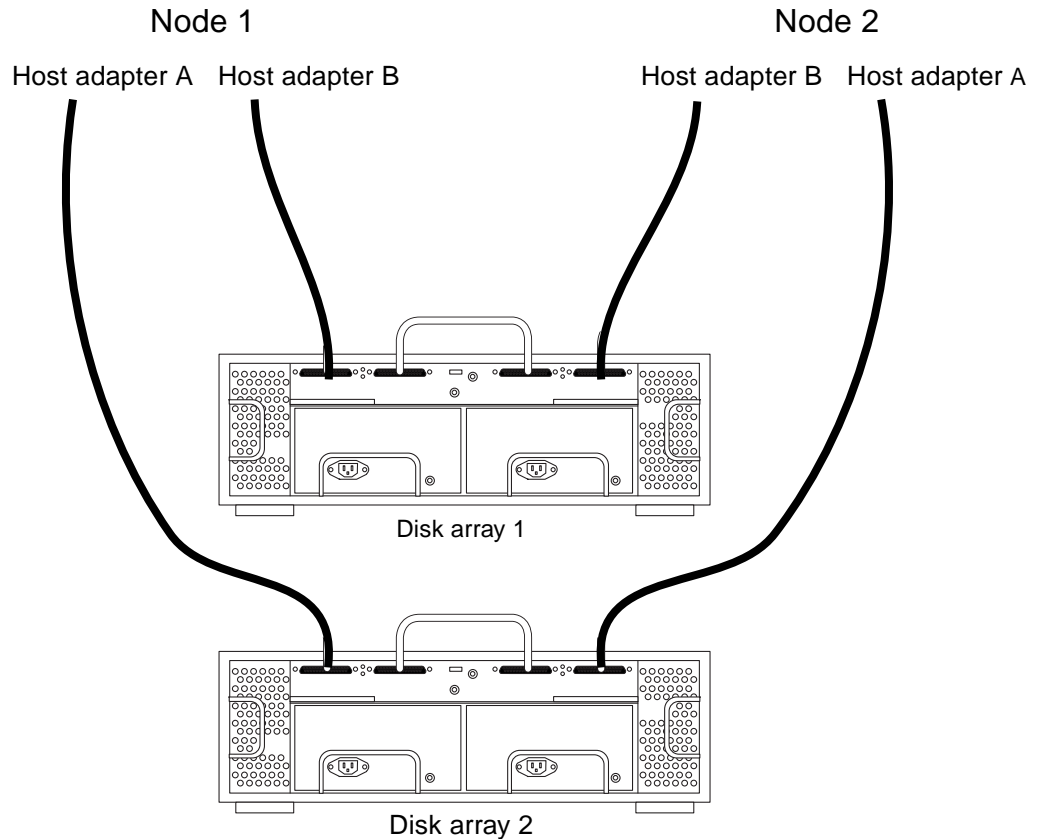


Figure B-2 Example of a Sun StorEdge D1000 Disk Array Mirrored Pair

4. Connect the AC power cord for each disk array of the pair to a different power source.
5. Power on the first node and the disk arrays.

6. Find the paths to the host adapters.

```
ok show-disks
```

Identify and record the two controllers that will be connected to the storage devices and record these paths. Use this information to change the SCSI addresses of these controllers in the `nvrामrc` script. Do not include the `/sd` directories in the device paths.

7. Edit the `nvrामrc` script to change the `scsi-initiator-id` for the host adapter on the first node.

For a list of `nvrामrc` editor and `nvedit` keystroke commands, see the "NVRAMRC Editor and NVEDIT Keystroke Commands" on page B-11.

The following example sets the `scsi-initiator-id` to 6. The OpenBoot PROM Monitor prints the line numbers (0:, 1:, and so on).

```
nvedit
0: probe-all
1: cd /sbus@1f,0/QLGC,isp@3,10000
2: 6 encode-int " scsi-initiator-id" property
3: device-end
4: cd /sbus@1f,0/
5: 6 encode-int " scsi-initiator-id" property
6: device-end
7: install-console
8: banner [Control C]
ok
```



Caution – Insert exactly one space after the first double quotation mark and before `scsi-initiator-id`.

8. Store or discard the changes.

The edits are done on a temporary copy of the `nvrामrc` script. You can continue to edit this copy without risk. After you complete your edits, save the changes. If you are not sure about the changes, discard them.

- To store the changes, type:

```
ok nvstore
```

```
ok
```

- To discard the changes, type:

```
ok nvquit
```

```
ok
```

9. Verify the contents of the `nvrामrc` script you created in Step 7.

```
ok printenv nvrामrc
nvrामrc = probe-all
cd /sbus@1f,0/QLGC,isp@3,10000
6 encode-int " scsi-initiator-id" property
device-end
cd /sbus@1f,0/
6 encode-int " scsi-initiator-id" property
device-end
install-console
banner
ok
```

10. Instruct the OpenBoot PROM Monitor to use the `nvrामrc` script.

```
ok setenv use-nvrामrc? true
use-nvrामrc? = true
ok
```

11. Without allowing the node to boot, power on the second node. If necessary, abort the system to continue with OpenBoot PROM (OBP) Monitor tasks.
12. Verify that the `scsi-initiator-id` for each host adapter on the second node is set to 7.

```
ok cd /sbus@1f,0/QLGC,isp@3,10000
ok .properties
scsi-initiator-id          00000007
differential
isp-fcode                  1.21 95/05/18
device_type                scsi
...
```

13. Continue with the Solaris Operating Environment, Sun Cluster software, and volume management software installation tasks.

For software installation procedures, refer to the *SunTM Cluster 3.0 07/01 Installation Guide*.

NVRAMRC Editor and NVEDIT Keystroke Commands

The OBP Monitor builds its own device tree based on the devices attached to the system when the boot sequence is invoked. The OBP Monitor has a set of default aliases for the commonly occurring devices in the system.

An `nvrामrc` script contains a series of OBP commands that are executed during the boot sequence. The procedures in this guide assume that this script is empty. If your `nvrामrc` script contains data, add the entries to the end of the script. To edit an `nvrामrc` script or merge new lines in an `nvrामrc` script, you must use `nvedit` editor and `nvedit` keystroke commands.

Table B-1 and Table B-2 on page B-12 list useful `nvedit` editor and `nvedit` keystroke commands. For an entire list of `nvedit` editor and `nvedit` keystroke commands, refer to the *OpenBoot™ 3.x Command Reference Manual*.

Table B-1 NVEDIT Editor Commands

Command	Description
<code>nvedit</code>	Enter the <code>nvrामrc</code> editor. If the data remains in the temporary buffer from a previous <code>nvedit</code> session, resume editing previous contents. Otherwise, read the contents of <code>nvrामrc</code> into the temporary buffer and begin editing it. This command works on a buffer, and you can save the contents of this buffer by using the <code>nvstore</code> command.
<code>nvstore</code>	Copy the contents of the temporary buffer to <code>nvrामrc</code> , and discard the contents of the temporary buffer.
<code>nvquit</code>	Discard the contents of the temporary buffer, without writing it to <code>nvrामrc</code> . Prompt for confirmation.
<code>nvrecover</code>	Attempts to recover the content of the <code>nvrामrc</code> if the content was lost as a result of the execution of <code>set-defaults</code> , then enters the <code>nvrामrc</code> editors as with <code>nvedit</code> . This command fails if <code>nvedit</code> is executed between the time the content of <code>nvrामrc</code> was lost and the time the content of the <code>nvrामrc</code> was executed.
<code>nvrुn</code>	Executes the contents of the temporary buffer.

Table B -2 NVEDIT Keystroke Commands

Keystroke	Description
^A	Moves to the beginning of the line.
^B	Moves backward one character.
^C	Exits the script editor.
^F	Moves forward one character.
^K	Deletes until end of line.
^L	Lists all lines.
^N	Moves to the next line of the <code>nvrामrc</code> editing buffer.
^O	Inserts a new line at the cursor position and stay on the current line.
^P	Moves to the previous line of the <code>nvrामrc</code> editing buffer.
^R	Replaces the current line.
Delete	Deletes previous character.
Return	Inserts a new line at the cursor position and advances to the next line.

Sun Cluster Administrative Command Summary

This appendix contains brief overviews of the Sun™ Cluster 3.0 07/01 command formats and examples.

Command Overview

Sun™ Cluster 3.0 07/01 comes with a number of commands and utilities that you can use to administer the cluster. Refer the man pages for more detailed information on each of these commands and utilities.

The Sun Cluster 3.0 administrative commands and utilities are:

- `scinstall` – Installs cluster software and initializes cluster nodes
- `scconf` – Updates the Sun Cluster software configuration
- `scsetup` – Interactive Sun Cluster configuration tool
- `sccheck` – Checks and validates Sun Cluster configuration
- `scstat` – Displays the current status of the Cluster
- `scgdevs` – Administers the global device namespace
- `scdidadm` – Disk ID configuration and administration utility
- `scshutdown` – Utility to shut down a cluster
- `scrgadm` – Manages registration and configuration of resource types, resources, and resource groups
- `scswitch` – Performs ownership or state changes of Sun Cluster resource groups and disk device groups
- `pnmset` – Sets up and updates the configuration for Public Network Management (PNM)
- `pnmstat` – Reports status for network automatic failover (NAFO) groups managed by PNM
- `pnmptor`, `pnmrtp` – Maps pseudo adapter to real adapter name (`pnmptor`) or real adapter to pseudo adapter name (`pnmrtp`) in NAFO groups
- `ccp` – Cluster control panel (administrative console)
- `cconsole`, `ctelnet`, `crlogin` – Multi-window, multi-machine remote administration console tools

The scinstall Utility

Use the `scinstall` utility to install Sun Cluster software and initialize new cluster nodes. When run with no arguments, `scinstall(1M)` runs in an interactive fashion, presenting the user with menus and prompts to perform its tasks. The `scinstall` utility can also be run in a non-interactive mode by supplying the proper command-line arguments.

The `scinstall` utility is located in the `Tools` directory of the SunTM Cluster 3.0 07/01 CD or in `/usr/cluster/bin` on a node where Sun Cluster has already been installed.

All forms of `scinstall` affect only the node it is run on.

Command Formats

- To run `scinstall` interactively:

```
scinstall
```

- To install Sun Cluster software and/or initialize a node as a new Sun Cluster member:

```
scinstall -i [-k] [-d <cdimage_dir>] [-s <svrc>,...]
           [-N <clusternode>
```

```
           [-C <clustername>]
           [-T <authentication_options>]
           [-G {<special> | <filesystem>} ]
           [-A <adapter_options>]
           [-B <junction_options>]
           [-m <cable_options>]
           [-w [<netaddr_options>]
```

```
]
```

- To upgrade a Sun Cluster node:

```
scinstall -u [-d <cdimage_dir>] [-s <svrc>,...]
           [-N <clusternode>
```

```
           [-C <clustername>]
           [-G {<special> | <filesystem>} ]
           [-T authentication_options]
```

```
]
```

- To set up a Sun Cluster install server (copies the CD image to an install directory):

```
scinstall -a <install_dir> [-d <cdimage_dir>]
```

- To establish the given node name as an installation client of the install server:

```
scinstall -c <jumpstart_dir> -h <nodename>  
          [-d <cdimage_dir>] [-s <srvc>, ...]  
          [-N <clusternode>  
           [-C <clustername>]  
           [-G {<special> | <filesystem>}]  
           [-T <authentication_options>]  
           [-A <adapter_options>]  
           [-B <junction_options>]  
           [-m <cable_options>]  
           [-w <netaddr_options>]
```

- To print the release and package version information:

```
scinstall -p [-v]
```

The `scconf` Command

Use the `scconf(1M)` command to manage the cluster software configuration. Use it to add new items to the configuration, change the properties of already configured items and remove items from the configuration. You can run the `scconf` command from any node that is a member of the cluster (and is usually only run on one node). Items that `scconf` manages include:

- Quorum options
- Disk device groups (Solstice DiskSuite disksets, VxVM disk groups or global raw devices)
- The name of the cluster
- Adding or removing cluster nodes
- Cluster transport adapters, junctions and cables
- Private host names for the nodes (host name used over the cluster transport)
- Node authentication options

When used with the `-p` option, `scconf` prints the current cluster configuration.

The `scconf` command is located in the `/usr/cluster/bin` directory.

Command Formats

To add or initialize a new item to the software configuration (for example, a new node, transport adapter, junction or cable, quorum device, device group or authentication option):

```
scconf -a [-Hv] [-h <node_options>]
          [-A <adapter_options>]
          [-B <junction_options>]
          [-m <cable_options>]
          [-p <privatehostname_options>]
          [-q <quorum_options>] [-D <devicegroup_options>]
          [-T <authentication_options>]
```

- To change the options for an existing item in the software configuration (that is, the cluster name, a transport adapter, junction or cable, the private host names, quorum devices, device groups options and authentication options):

```
scconf -c [-Hv] [-c <cluster_options>][-A <adapter_options>]
        [-B <junction_options>] [-m <cable_options>]
        [-P <privatehostname_options>]
        [-q <quorum_options>]
        [-D <devicegroup_options>]
        [-T <authentication_options>]
```

- To remove an item from the software configuration (that is, a node, adapter, junction, cable, quorum device, device group or authentication):

```
scconf -r [-Hv] [-h <node_options>] [-A <adapter_options>]
        [-B <junction_options>]
        [-m <cable_options>] [-q <quorum_options>]
        [-D <devicegroup_options>]
        [-T <authentication_options>]
```

- To print out the current configuration:

```
scconf -p [-Hv]
```

- To print help information about the command options:

```
scconf [-H]
```

Each form of the command accepts a `-H` option. If present, this option causes `scconf` to print help information (specific to the form of the command used) and ignore any other options given.

Command Example

To add a new adapter, `hme3`, on node `venus`:

```
# scconf -a -A trtype=dlpi,name=hme3,node=venus
```

The `scsetup` Utility

The `scsetup` utility is an interactive, menu-driven utility that can perform most of the postinstallation cluster configuration tasks that are handled by `scconf`. `scsetup`. Run it immediately after the cluster software has been installed and all nodes have joined the cluster (`scsetup` automatically detects the new installation and prompts for the proper quorum configuration information).

In the Sun™ Cluster 3.0 07/01 release, a new function has been added to the `scsetup` utility: Resource Groups. Using the `scsetup` utility, you can create failover and shared address resource groups.

You can run the `scsetup` utility from any node of the cluster.

Command Example

```
# scsetup

*** Main Menu ***

Please select from one of the following options:

    1) Quorum
    2) Resource groups
    3) Cluster interconnect
    4) Device groups and volumes
    5) Private hostnames
    6) New nodes
    7) Other cluster properties

    ?) Help with menu options
    q) Quit

Option:
```



Note – The first time `scsetup` is run on a new cluster installation, it does not display the previous menu.

The `sccheck` Utility

The `sccheck` utility, when run on a node of the cluster (can be run on any node currently in the cluster), checks the validity of the cluster configuration. It checks to make sure that the basic configuration of the cluster is correct and consistent across all nodes.

Command Format

Options can be given to `sccheck` to invoke a brief check, print verbose messages, suppress warning messages, or to perform the check on only certain nodes of the cluster.

```
sccheck [-bvW] [-h <hostlist>
  -b : perform a brief check
  -v : verbose mode
  -W : disable warnings
  -h : Run check on specific hosts
```

Command Example

```
# sccheck -v
vfstab-check: CHECKED - Check for node id
vfstab-check: CHECKED - Check for node id
vfstab-check: CHECKED - Check for
/global/.devices/node@<id>
vfstab-check: CHECKED - Check for mount point
vfstab-check: CHECKED - Check for identical global entries
vfstab-check: CHECKED - Check for option 'syncdir'
vfstab-check: CHECKED - Check for physical connectivity
vfstab-check: CHECKED - Check for option 'logging' for raw
device
vfstab-check: CHECKED - vfstab check completed.
```


The `scstat` Command

The `scstat` command prints the current status of various cluster components. You can use it to display the following information:

- The cluster name
- A list of cluster members
- The status of each cluster member
- The status of resource groups and resources
- The status of every path in the cluster interconnect
- The status of every disk device group
- The status of every quorum device

Command Format

```
scstat [-DWgnpq] [-h node]
-D - Disk group status
-W - interconnect status,
-g - resource group status,
-n node status,
-p - all components status,
-q - quorum device status
```

Command Example

```
# scstat -g
Resource Group
Resource Group Name:          netscape-rg
Status
Node Name:                    venus
Resource Group State:         Online

Node Name:                    mars
Resource Group State:         Offline

Resource
Resource Name:                netscape-server
Status
Node Name:                    venus
```

```
Resource Monitor Status/Message:   Online -
    SharedAddress online
Resource State:                     Online

Node Name:                          mars
Resource Monitor Status/Message:   Offline -
    SharedAddress offline
Resource State:                     Offline

Resource Group Name:                netscape-rg-2
Status
Node Name:                          venus
Resource Group State:               Online

Node Name:                          mars
Resource Group State:               Online

Resource
Resource Name:                      netscape-res
Status
Node Name:                          venus
Resource Monitor Status/Message:   Online -
    Successfully started Netscape Web Server
    for resource <netscape-res>.
Resource State:                     Online

Node Name:                          mars
Resource Monitor Status/Message:   Online -
    Successfully started Netscape Web Server
    for resource <netscape-res>.
Resource State:                     Online
```

The `scgdevs` Utility

Use the `scgdevs` utility to manage the global devices namespace. The global devices namespace is mounted under `/global` and consists of a set of symbolic links to physical device files.

By calling `scgdevs`, an administrator can attach new global devices (such as a tape drive, CD-ROM drive, or disk drive) to the global devices namespace without requiring a system reboot. The `drvconfig`, `disks`, `tapes`, or `devlinks` commands must be run prior to running `scgdevs`. Also run `devfsadm` before running `scgdevs`.

Run this command on the node (the node must be a current cluster member) where the new device is being installed.

Command Example

```
# drvconfig
# disks
# devfsadm
# scgdevs
Configuring DID devices
Configuring the /dev/global directory (global devices)
obtaining access to all attached disks
reservation program successfully exiting
```

The `scdidadm` Command

Use the `scdidadm` command to administer the disk ID (DID) pseudo device driver. It can create driver configuration files, modify entries in the configuration file, load the current configuration files into the kernel, and list the mapping between DID devices and the physical devices.

Run the `scdidadm` command during cluster startup to initialize the DID driver. It is also used by the `scgdevs` command to update the DID driver. The primary use of the `scdidadm` command by the administrator is to list the current DID device mappings.

Command Formats

The `scdidadm` command runs from any node of the cluster.

- To perform a consistency check against the kernel representation of the devices and the physical devices:

```
scdidadm -c
```

- To remove all DID references to underlying devices that have been detached from the current node (use after running the normal Solaris device commands to remove references to non-existent devices):

```
scdidadm -C
```

- To print out the DID device mappings:

```
scdidadm -l | -L [-h] [-o <fmt>,...] [<path> |  
<DID_instance>]  
    fmt can be instance, path, fullpath, host, name,  
    fullname, diskid or asciidiskid
```

- To reconfigure the DID database to add any new devices (this is performed by `scgdevs`):

```
scdidadm -r
```

- To replace a disk device in the DID database:

```
scdidadm -R <path> | <DID_instance>
```

- To run `scgdevs` on each member of the cluster:

```
scdidadm -S
```

- To initialize and load the DID configuration into the kernel:

```
scdidadm -ui
```

- To print the version number of this program:

```
scdidadm -v
```

Command Example

```
# scdidadm -hlo instance,host,path,name
```

Instance	Host	Physical Path	Pseudo Path
1	venus	/dev/rdisk/c0t0d0	d1
2	venus	/dev/rdisk/c1t2d0	d2
3	venus	/dev/rdisk/c1t3d0	d3
4	venus	/dev/rdisk/c1t4d0	d4
5	venus	/dev/rdisk/c1t5d0	d5
6	venus	/dev/rdisk/c2t2d0	d6
7	venus	/dev/rdisk/c2t3d0	d7
8	venus	/dev/rdisk/c2t4d0	d8
9	venus	/dev/rdisk/c2t5d0	d9

The `scswitch` Command

Use the `scswitch` command to perform the following tasks:

- Switch resource groups or disk device groups to new primary nodes:
`scswitch -z ...`
- Bring resource groups or disk device groups online or offline:
`scswitch -z ...` or `scswitch -m ...`
- Restart a resource group on a node:
`scswitch -R ...`
- Enable or disable resources and resource monitors:
`scswitch -e|-n ...`
- Switch resource groups to or from an “unmanaged” state:
`scswitch -o|-u ...`
- Clear error flags on resources `scswitch -c ...`
- Bring resource group offline on all nodes:
`scswitch -F -g`
- Enable all resources, make resource group managed, and bring resource group online on default masters:
`scswitch -Z -g [optional]...`

The `scswitch` command runs on any node of the cluster.

Command Formats

To switch the primary for a resource group (or bring the resource group online if it is not online on any node):

```
scswitch -z -g <resource_grp>[,<resource_grp>...] -h  
<node>[,<node>...]
```

To switch the primary for a disk device group (or bring the disk device group online if it is currently offline):

```
scswitch -z -D <device_group_name>[,<device_group_name>...]
-h <node>[,<node>...]
```

To place a resource group offline:

```
scswitch -z -g <resource_grp>[,<resource_grp>...] -h ""
```

To place a disk device group offline (places the disk device group into “maintenance mode”):

```
scswitch -m -D <device_group_name>[,<device_group_name>...]
```

To restart a resource group on a node:

```
scswitch -R -g <resource_group>[,<resource_group>...] -h
<node>[,<node>...]
```

To enable a resource or resource monitor:

```
scswitch -e [-M] -j <resource>[,<resource>...]
```

To disable a resource or resource monitor:

```
scswitch -n [-M] -j <resource>[,<resource>...]
```

To make a resource group “managed” (that is, bring the resource group under cluster control):

```
scswitch -o -g <resource_grp>[,<resource_grp>...]
```

To make a resource group “unmanaged” (that is, take the resource group away from cluster control):

```
scswitch -u -g <resource_grp>[,<resource_grp>...]
```

To clear a resource’s error flags:

```
scswitch -c -h <node>[,<node>...] -j
<resource>[,<resource>...] -f <flag_name>
(flag_name can be: BOOT_FAILED, UPDATE_FAILED, INIT_FAILED,
FINI_FAILED, or STOP_FAILED)
```

Before clearing a `STOP_FAILED` flag, make sure that the data service is actually down.



Note – Only `STOP_FAILED` is currently implemented.

The `scshutdown` Command

Use the `scshutdown` command to shut down the entire cluster. It runs from any active cluster node.

When shutting down the cluster, `scshutdown` performs the following tasks:

1. It places all of the functioning resource groups on the cluster into an offline state. If any of the transitions fail, `scshutdown` aborts.
2. It unmounts all of the cluster file systems. If any of the unmounts fail, `scshutdown` aborts.
3. It shuts down all of the active device services. If any of the transitions fail, `scshutdown` aborts.
4. It runs `/usr/sbin/init 0` on all nodes.

Command Format

- To shut down all nodes in the cluster:

```
scshutdown [-y] [-g <grace_period>] [<message>]
```

Command Example

Use the following command to shut down a cluster in 60 seconds and issue a warning message.

```
# scshutdown -y -g 60 "Log Off Now"
```

The `scrgadm` Command

Use the `scrgadm` command for the following tasks:

- Add, change, or remove resource types
- Create or change the properties of or remove resource groups
- Add or change the properties of or remove resources within resource groups, including logical host name or shared address resources
- Print the properties of resource groups and their resources

The `scrgadm` command runs on any node that is a member of the cluster.

Command Formats

- To register a resource type:

```
scrgadm -a -t <resource_type_name> [-h  
<RT_installed_node_list>  
[-f <registration_file_path>]
```

- To deregister a resource type:

```
scrgadm -r -t <resource_type_name>
```

- To create a new resource group:

```
scrgadm -a -g <RG_name> [-h <nodelist>] [-y  
<property=value> [...]]
```

Use `-y Maximum primaries=n` to create a scalable resource group.

- To add a logical host name or shared address resource to a resource group:

```
scrgadm -a -g <RG_name> -l <hostnamelist> [-n  
<netiflist>]
```

- To add a resource to a resource group:

```
scrgadm -a -j <resource_name> -t <resource_type_name> -  
g <RG_name>  
[-y <property=value> [...]] [-x  
<extension_property=value> [...]]
```

- To change the properties of a resource group:

```
scrgadm -c -g <RG_name> -y <property=value> [-y
<property=value>]
```

- To change the properties of a resource:

```
scrgadm -c -j <resource_name> [-y <property=value>
[...]] [-x <extension_property=value> [...]]
```

- To remove a resource from a resource group:

```
scrgadm -r -j <resource_name>
```

A resource must be disabled (using `scswitch -n`) before it can be removed.

- To remove a resource group:

```
scrgadm -r [-L|-S] -g <RG_name>
```

Before removing a resource group, perform the following steps:

- Place the resource group offline:
`scswitch -z -g <RG_name> -h ""`
- Disable all resources:
`scswitch -n -j <resource_name>`
- Remove all resources:
`scrgadm -r -j <resource_name>`
- Make the resource group unmanaged:
`scswitch -u -g <RG_name>`

- To print out the resource types, resource groups and resources (and their properties) in the cluster:

```
scrgadm -p[v[v]]
```

The additional `-v` flags provide more verbose output.

The `pnmset` Utility

Use the `pnmset` utility to configure NAFO groups on a node. It can be used to:

- Create, change, or remove a NAFO group
- Migrate IP addresses from the active adapter to a configured standby adapter
- Print out the current NAFO group configuration

You can run the `pnmset` utility interactively or, if backup groups have already been configured, non-interactively.

The `pnmset` utility only affects the node it is run on. It must be run separately on each node of the cluster.

Command Formats

- To create NAFO groups (initial setup):

```
pnmset [-f <filename>] [-n[-t]] [-v]
```

Where:

`-f <filename>` indicates a filename to save or read the configuration to/from (see `pnmconfig(4)`). Default is `/etc/cluster/pnmconfig`.

`-n` Do not run interactively, instead read configuration file for NAFO group information (see `pnmconfig(4)` for file format)

`-t` Do not run a test of the interfaces before configuring the NAFO groups (only valid with the `-n` option)

`-v` Do not start or restart the `pnmd` daemon, verify only. Any new groups will not be active until the daemon is restarted.

- To reconfigure PNM (after the PNM service has already been started):

```
pnmset -c <NAFO_group> -o <subcommand> [<subcommand  
args> ...]
```

Subcommands:

`create [<adp1> <adp2> ...]` - creates a new NAFO group

`delete` - deletes the NAFO group

`add <adp>` - adds the specified adapter to the NAFO group

`remove <adp>` - removes the specified adapter from the NAFO group

`switch <adp>` - moves the IP addresses from the current live adapter to the specified adapter

- To print out the current NAFO group configuration:

```
pnmset -p
```

The `pnmstat` Command

The `pnmstat` command reports the current status of the NAFO groups configured on a node. It reports the following information:

- Status of the NAFO groups:
 - a. OK – The NAFO groups are working.
 - b. DOUBT – The NAFO groups are currently in a transition state. PNM has not determined if the group is healthy or down.
 - c. DOWN – The NAFO group is down, no adapters in the group are capable of hosting the configured IP addresses.
- Seconds since the last failover
- Currently active adapter

If run without any arguments, `pnmstat` displays the overall status of PNM on the node. When run with a specific NAFO group (`-c`) or with the `-l` option, it displays all three statistics.

The `pnmstat` command reports only on the node it is run on unless the `-h` option is given, in which case, it reports the NAFO group status on the specified host.

Command Formats

- To report the general status of PNM on a node:
`pnmstat`
- To report the status of all the NAFO groups on a node:
`pnmstat -l`
- To report the status of a particular NAFO group on a node:
`pnmstat -c <NAFO_group>`
- To report the status of the NAFO groups on another node:
`pnmstat -h <host> [-s] [-c <NAFO_group>] [-l]`

The `-s` option indicates that the cluster interconnect should be used instead of the public network.

Command Examples

```
# pnmstat  
OK
```

```
# pnmstat -c nafo0  
OK  
NEVER  
hme0
```

```
# pnmstat -l  
group    adapters          status  fo_time  act_adp  
nafo0    hme0             OK      NEVER    hme0
```

```
# pnmstat -h venus -c nafo0  
OK  
NEVER  
hme0
```

The `pnmptor` and `pnmrtp` Commands

The `pnmrtp` and `pnmptor` commands map NAFO group names (pseudo adapter names) to actual network adapter names and vice versa.

These commands only report on NAFO groups that are configured on the local node.

Command Formats and Examples

- To convert a NAFO group name to the name of the currently active adapter:

```
pnmptor <NAFO_group>
```

```
# pnmptor nafo0  
hme0
```

- To display the NAFO group for a specified network adapter:

```
pnmrtp <adp>
```

```
# pnmrtp hme0  
nafo0
```


Sun Cluster Node Replacement

This appendix describes the process used to replace a node that has suffered a catastrophic failure.

Node Replacement Overview

If a node in the cluster completely fails and requires replacement, it first must be removed from the cluster, and then re-added to the cluster as a new node. You can also use this procedure to reestablish a node that had a complete failure of its boot disk and no mirror or backup was available. The procedure for replacing a failed node is:

1. Remove the failed node from the cluster framework.

This involves determining and removing all references to the node in the various cluster subsystems (device group access, quorum devices, resource groups, and so on.)

2. Physically replace the failed node.

Verify that the new node has the proper hardware installed to serve as a replacement for the failed node. This includes the transport adapters, public network interfaces, and storage adapters.

3. Add the new node to the existing cluster.

This involves configuring the existing cluster to accept the new node into the cluster and performing a standard Sun Cluster installation on the “new” node. After installation, the resource group, quorum, and device group relationships can be reestablished.

Replacement Preparation

To simulate a complete node failure, shut down one of your nodes and initiate a JumpStart operation to reload the Solaris 8 10/00 Operating Environment.

```
ok boot net - install
```

While the JumpStart operation proceeds, you can start the replacement procedure on the remaining active cluster node.

Logically Removing a Failed Node

The removal process is a set of steps in which you remove all references to a node from the existing cluster framework. This allows the cluster to later accept the replacement node as a “new” node, making it easier to keep the framework consistent across all the nodes of the cluster.

Perform the following steps to remove a failed node from the cluster framework.

1. Place the node into a maintenance state

Use the `scconf` command to place the node into a maintenance state. This removes it from the quorum vote count total, which helps minimize possible quorum loss problems while you work on replacing the node.

```
# scconf -c -q node=node_name,maintstate
```

2. Remove the node from all resource groups.

- a. Determine the node ID of the node being removed:

```
# scconf -pv | grep "Node ID"
(venus) Node ID: 1
(saturn) Node ID: 2
```

- b. Determine the set of resource groups that reference the node ID for the node being removed. The node ID is referred to in the `NetIfList` property of the network resource in the resource group.

```
# scrgadm -pvv | grep -i netiflist | grep "property value"
(nfs-rg:nfs-server:NetIfList) Res property value: nafo0@1
nafo0@2
```

- c. Update the `NetIfList` property for the resources that refer to the node being removed. Change the `netiflist` property to leave off the node being removed.

```
# scrgadm -c -j nfs-server -x netiflist=nafo0@1
```

- d. Determine the set of resource groups that refer to the node in its `nodelist` property:

```
# scrgadm -pvv | grep "Res Group Nodelist"
(nfs-rg) Res Group Nodelist:          venus saturn
```

- e. Delete the node being removed from the `nodelist` property of the resource groups:

```
# scrgadm -c -g nfs-rg -y nodelist=venus
```

3. Remove the node from all VxVM device group node lists.

- a. Determine the VxVM device groups and node lists.

```
# scconf -p | grep "Device group"
```

- b. Use the `scconf` command to remove the failed node from the device group node lists.

```
# scconf -r -D name=nfs_dg_1,nodelist=saturn
```

4. Remove the failed node from all Solstice DiskSuite diskset node lists.

- a. Determine the Solstice DiskSuite diskset node lists.

```
# metaset
```

- b. Use the `metaset` command to remove the failed node from the diskset node lists.

```
# metaset -s nfs_ds_1 -f -d -h saturn
```

5. Reset the `localonly` flag for the failed node's boot disk DID instance.

```
# scconf -pvv | grep Local_Disk
(dsk/dl0) Device group type:  Local_Disk
(dsk/dl1) Device group type:  Local_Disk

# scdidadm -L dl0 dl1
10    saturn:/dev/rdisk/c0t0d0  /dev/did/rdisk/dl0
1     venus:/dev/rdisk/c0t0d0   /dev/did/rdisk/dl1

# scconf -c -D name=dsk/dl0,localonly=false
```

6. Remove the node from all raw disk devices.

```
# scconf -pvv | grep saturn | grep Device
(dsk/dl2) Device group node list:  saturn
(dsk/dl1) Device group node list:  venus, saturn
(dsk/dl0) Device group node list:  venus, saturn
(dsk/d9) Device group node list:  venus, saturn

# scconf -r -D name=dsk/dl2,nodelist=saturn
# scconf -r -D name=dsk/dl1,nodelist=saturn
# scconf -r -D name=dsk/dl0,nodelist=saturn
# scconf -r -D name=dsk/d9,nodelist=saturn
```

7. Remove all transport connections to the node.

- a. Use `scsetup` or `scconf` to remove all transport cables that have an endpoint on the node being removed. You can use the `scconf -pvv` command to show which transport cables have endpoints on the node being removed.

```
# scconf -pvv | grep -i "transport cable"
Transport cable:  saturn:hme0@0    venus:hme1@0  Enabled
Transport cable:  saturn:hme1@0    venus:hme2@0  Enabled

# scconf -r -m endpoint=saturn:hme0
# scconf -r -m endpoint=saturn:hme1
```

- b. Use the `scsetup` or `scconf` commands to remove all transport adapters in the node being removed. You can use the `scconf -pvv` command to show which transport adapters are configured for the node.

```
# scconf -pvv | grep -i "transport adapters"
(venus) Node transport adapters: hme1 hme2
(saturn) Node transport adapters: hme0 hme1
# scconf -r -A name=hme0,node=saturn
# scconf -r -A name=hme1,node=saturn
```

8. Remove all quorum devices to which the node is connected.

All quorum devices to which the node being removed is connected must be removed from the cluster configuration. Use the `scconf` command to remove the quorum device.

If the quorum device being removed is the last quorum device in a two-node cluster, the cluster must first be placed back into `installmode` prior to removing the device. This can be done by using `scconf`.

```
# scconf -pvv | grep -i quorum | grep saturn
(saturn) Node quorum vote count: 0
(d2) Quorum device hosts (enabled): venus saturn
# scconf -r -q globaldev=d2
scconf: Make sure you are not attempting to remove the
last required quorum device.
scconf: Failed to remove quorum device (d2) - quorum could
be compromised.
# scconf -c -q installmode
# scconf -r -q globaldev=d2
```

9. Remove the node from the cluster framework

Finally, remove the node itself from the cluster configuration using the `scconf` command. At this point, all traces of the node have been removed from the cluster.

```
# scconf -r -h saturn
```

Physically Replacing a Failed Node

Now that the failed node has been completely removed from the cluster framework, the node can be physically replaced with a new server system.

1. Remove the failed system, taking care to properly label any network, storage, and power cables so they can be easily identified and correctly plugged into the replacement system.
2. Verify that the replacement system has the proper network interfaces, storage interfaces, memory, and so on, to adequately serve as a replacement node in the cluster.
3. Connect the cluster interconnect cables, public network cables, and storage cables to the replacement system.
4. Connect the power cable to the new system.
5. Power up the replacement system.

Logically Adding a Replacement Node

Add the replacement node to the existing cluster. As far as the existing cluster is concerned, this is to be treated as a brand new node.

1. On one of the existing nodes of the cluster, configure the cluster to accept a new member.

```
# scconf -s -T node=saturn
```

2. Install the Solaris 8 Operating Environment on the new node and install the Sun Cluster software. Use option 2, Add this machine as a node in an established cluster.

During the installation, the proper raw device groups should be automatically built along with the cluster interconnect components for the node (adapters and cables).

3. After the new node has joined the cluster, configure any quorum devices required. If the cluster is still in install mode, make sure to reset the quorum configuration using `scconf` or `scsetup`.

```
# scconf -a -q globaldev=d2
# scconf -c -q reset
```

4. Add the node to any volume management groups of which it should be a part.

For VxVM: Use `scsetup` or `scconf` to add the node into any VxVM disk groups of which the node should be a part.

```
# scconf -a -D name=nfs_dg_1,nodelist=saturn
```

For Solstice DiskSuite: Use the `metaset` command to add the host to any disksets it needs to be a member of. *This must be done from a node other than the one you are replacing.*

```
# metaset -s nfs_ds_1 -a -h saturn
```

5. Create NAFO groups for the node's public network interfaces. Try to mirror the configuration of the failed node.

```
# pnmset -c nafo0 -o create hme0 qfe0
# pnmset -c nafo1 -o create hme4 qfe1
```

6. Add the node to any resource groups that might be mastered by the new node.

Refer to the list of resource groups from which you removed the node from the `nodelist` parameter. Reconstruct the `nodelist` parameter for these resource groups, adding the node name for the “new” node.

Also, add the node’s NAFO group back into each of the resources from which it was deleted when the node was removed. When rebuilding the `netiflist` for each resource, be sure to use the format of `nafo<NAFO Instance>@<NodeName or NodeID>`. Make sure to assign the proper NAFO group based on the IP address of the resource.

```
# scrgadm -c -g nfs-rg -y nodelist=venus,saturn
# scrgadm -c -j nfs-server -x netiflist=nafo0@venus,nafo0@saturn
```

7. At this point, the replacement node should be fully operational. You can use `scswitch` to switch over any device or resource groups to help balance the cluster.

```
# scswitch -z -D nfs_ds_1 -h saturn
# scswitch -z -g nfs-rg -h saturn
```


Sun Cluster HA for Oracle Installation

This appendix describes the basic process of installing and configuring a highly available Oracle database.

Installation Process

Following is a summary of a Sun Cluster HA for Oracle data service installation using Oracle 8.1.6. The installation is intended for a two-node cluster, a single pair configuration.

1. Install the Sun™ Cluster 3.0 07/01 software on both nodes.
2. Install the Sun Cluster HA for Oracle software on both nodes.
3. Install Oracle 8.1.6 in the following configuration:
 - a. The Oracle binaries are local to each node under `/oracle`.
 - a. The database is in VERITAS file system volumes that are globally mounted under `/global/oracle`.
4. Edit the `listener.ora` and `tnsnames.ora` to point to the logical host name (`ford-ora`) you are using for this data service.
5. Register the `SUNW.oracle_server` and `SUNW.oracle_listener` resource types.

```
# scrgadm -a -t SUNW.oracle_server
```

```
May 12 12:26:14 mustang Cluster.CCR: resource type  
SUNW.oracle_server added.
```

```
# scrgadm -a -t SUNW.oracle_listener
```

```
May 12 12:26:41 mustang Cluster.CCR: resource type  
SUNW.oracle_listener added.
```

6. Create a blank resource group, `oracle-rg` with a node list.

```
# scrgadm -a -g oracle-rg -h mustang,cobra
```

```
May 12 12:28:11 mustang Cluster.CCR: resource group  
oracle-rg added.
```

7. Try to register the `SUNW.HAStorage` resource type. The registration fails because it is a preregistered resource type.

```
# scrgadm -a -t SUNW.HAStorage
```

```
SUNW.HAStorage: resource type exists; can't create
```

8. Add the logical host name resource along with the appropriate NAFO group to use for each node in the node list.

```
# scrgadm -a -L -g oracle-rg -l ford-ora \  
-n nafo2@mustang,nafo1@cobra
```

```
May 12 12:33:46 mustang Cluster.RGM.rgmd: launching  
method <hafoip_validate> for resource <ford-ora>,  
resource group <oracle-rg>, timeout <300> seconds
```

```
May 12 12:33:46 mustang Cluster.RGM.rgmd: method  
<hafoip_validate> completed successfully for resource  
<ford-ora>, resource group <oracle-rg>
```

```
May 12 12:33:47 mustang Cluster.CCR: resource ford-ora  
added.
```

9. Add the SUNW.HAstorage resource type. Specify the Oracle data path and enable affinity so the data storage must follow the data service if just the data service fails over to another node.

```
# scrgadm -a -j hastorage-res -g oracle-rg \
-t SUNW.HAStorage \
-x ServicePaths=/global/oracle \
-x AffinityOn=TRUE
```

```
May 12 12:36:16 mustang Cluster.RGM.rgmd: launching
method <hastorage_validate> for resource <hastorage-
res>, resource group <oracle-rg>, timeout <300> seconds
```

```
May 12 12:36:17 mustang Cluster.RGM.rgmd: method
<hastorage_validate> completed successfully for
resource <hastorage-res>, resource group <oracle-rg>
```

```
May 12 12:36:17 mustang Cluster.CCR: resource
hastorage-res added.
```

10. Create local Oracle alert logs on both nodes.

```
# mkdir /var/oracle (on both nodes)
# touch /var/oracle/alert.log (on both nodes)
```

11. Add the SUNW.oracle_server resource, and set standard and extended resource properties.

```
# scrgadm -a -j ora_server_1 -t SUNW.oracle_server \
-g oracle-rg \
-x CONNECT_STRING=scott/tiger \
-y Resource_dependencies=hastorage-res \
-x ORACLE_SID=test \
-x ORACLE_HOME=/oracle \
-x Alert_log_file=/var/oracle/alert.log
```

```
May 12 12:50:33 mustang Cluster.RGM.rgmd: launching
method <bin/oracle_server_validate> for resource
<ora_server_1>, resource group <oracle-rg>, timeout
<120> seconds
```

```
May 12 12:50:34 mustang Cluster.RGM.rgmd: method
<bin/oracle_server_validate> completed successfully for
resource <ora_server_1>, resource group <oracle-rg>
```

```
May 12 12:50:34 mustang Cluster.CCR: resource
ora_server_1 added.
```

12. Add the `SUNW.oracle_listener` resource, and set standard and extended resource properties.

```
# scrgadm -a -j ora_listener_1 \  
-t SUNW.oracle_listener -g oracle-rg \  
-y Resource_dependencies=ora_server_1 \  
-x ORACLE_HOME=/oracle \  
-x LISTENER_NAME=listener \  

```

```
May 12 12:56:59 mustang Cluster.RGM.rgmd: launching  
method <bin/oracle_listener_validate> for resource  
<ora_listener_1>, resource group <oracle-rg>, timeout  
<60> seconds
```

```
May 12 12:56:59 mustang Cluster.RGM.rgmd: method  
<bin/oracle_listener_validate> completed successfully  
for resource <ora_listener_1>, resource group <oracle-  
rg>
```

```
May 12 12:57:00 mustang Cluster.CCR: resource  
ora_listener_1 added.
```

13. Bring the resource group online.

```
# scswitch -Z -g oracle-rg
```

Glossary

A

active server

A node in the Sun™ Cluster 3.0 07/01 configuration that is providing highly available data services.

administration console

A workstation that is outside the cluster that is used to run cluster administrative software.

API

Application programming interface.

ATM

Asynchronous transfer mode.

B

backup group

Used by NAFO. A set of network adapters on the same subnet. Adapters within a set provide backup for each other.

C

CCR

(cluster configuration repository) A highly available, replicated database that can be used to store data for HA data services and other Sun™ Cluster 3.0 07/01 configuration needs.

cluster

Two to four nodes configured together to run either parallel database software or highly available data services.

cluster interconnect

The private network interface between cluster nodes.

cluster node

A physical machine that is part of a Sun cluster. It is also referred to as a cluster host or cluster server.

cluster quorum

The set of cluster nodes that can participate in the cluster membership.

cluster reconfiguration

An ordered multistep process that is invoked whenever there is a significant change in cluster state. During cluster reconfiguration, the Sun™ Cluster 3.0 07/01 software coordinates all of the physical hosts that are up and communicating. Those hosts agree on which logical hosts should be mastered by which physical hosts.

cluster pair topology

Two pairs of Sun™ Cluster 3.0 07/01 nodes operating under a single cluster administrative framework.

CMM

(cluster membership monitor) The software that maintains a consistent cluster membership roster to avoid database corruption and subsequent transmission of corrupted or inconsistent data to clients. When nodes join or leave the cluster, thus changing the membership, CMM processes on the nodes coordinate global reconfiguration of various system services.

concatenation

A VERITAS volume or a Solstice DiskSuite metadvice created by sequentially mapping data storage space to a logical virtual device. Two or more physical components can be concatenated. The data accessed sequentially rather than interlaced (as with stripes).

D

data service

A network service that implements read-write access to disk-based data from clients on a network. NFS is an example of a data service. The data service may be composed of multiple processes that work together.

DES

Data Encryption Standard.

DID

Disk ID.

disk group

A well-defined group of multihost disks that move as a unit between two servers in an HA configuration. This can be either a Solstice DiskSuite diskset or a VERITAS Volume Manager disk group.

diskset

See disk group.

DiskSuite state database

A replicated database that is used to store the configuration of metadevices and the state of these metadevices.

DLM

(distributed lock management) Locking software used in a shared disk OPS environment. The DLM enables Oracle processes running on different nodes to synchronize database access. The DLM is designed for high availability; if a process or node crashes, the remaining nodes do not have to be shut down and restarted. A quick reconfiguration of the DLM is performed to recover from such a failure.

DLPI

Data Link Provider Interface.

DNS

Domain Name Service.

DR

Dynamic Reconfiguration.

DRL

Dirty Region log.

E**EEPROM**

Electrically erasable programmable read-only memory.

F**fault detection**

Sun™ Cluster 3.0 07/01 programs that detect two types of failures. The first type includes low-level failures, such as system panics and hardware faults (that is, failures that cause the entire server to be inoperable). These failures can be detected quickly. The second type of failures are related to data service. These types of failures take longer to detect.

fault monitor

A fault daemon and the programs used to probe various parts of data services.

FC-AL

Fibre Channel-Arbitrated Loop.

FCOM

Fibre-Channel Optical Module.

FDDI

Fiber Distributed Data Interface.

FF

Failfast.

Fibre-Channel connections

Fibre-Channel connections connect the nodes with the SPARCstorage™ arrays.

G

GBIC

Gigabit interface converter.

golden mediator

In Solstice DiskSuite configurations, the in-core state of a mediator host set if specific conditions were met when the mediator data was last updated. The state permits take operations to proceed even when a quorum of mediator hosts is not available.

GUI

Graphical user interface.

H

HA

High availability.

HA administrative file system

A special file system created on each logical host when Sun™ Cluster 3.0 07/01 is first installed. It is used by Sun™ Cluster 3.0 07/01 and by layered data services to store copies of their administrative data.

HA-NFS

Highly Availability-Network File System.

heartbeat

A periodic message sent between the several membership monitors to each other. Lack of a heartbeat after a specified interval and number of retries might trigger a takeover.

HFS

High Sierra file system.

highly available data service

A data service that appears to remain continuously available, despite single-point failures of server hardware or software components.

host

A physical machine that can be part of a Sun cluster. In Sun™ Cluster 3.0 07/01 documentation, host is synonymous with node.

hot standby server

In an N+1 configuration, the node that is connected to all multihost disks in the cluster. The hot standby is also the administrative node. If one or more active nodes fail, the data services move from the failed node to the hot standby. However, there is no requirement that the +1 node cannot run data services in normal operation.

HTML

Hypertext Markup Language.

HTTP

Hypertext Transfer Protocol.

I**IDLM**

(integrated distributed lock manager) A data access coordination scheme used by newer versions of the Oracle Parallel Server database. Also see **Distributed Lock Manager**.

I/O

Input/output.

IP

Internet Protocol.

L**LAN**

Local area network.

LANE

Local area network emulation.

LDAP

Lightweight Directory Access Protocol.

local disks

Disks attached to a HA server but not included in a diskset. The local disks contain the Solaris Operating Environment distribution and the Sun™ Cluster 3.0 07/01 and volume management software packages. Local disks must not contain data exported by the Sun™ Cluster 3.0 07/01 data service.

logical host

A set of resources that moves as a unit between HA servers. In the current product, the resources include a collection of network host names and their associated IP addresses plus a group of disks (a diskset). Each logical host is mastered by one physical host at a time.

logical host name

The name assigned to one of the logical network interfaces. A logical host name is used by clients on the network to refer to the location of data and data services. The logical host name is the name for a path to the logical host. Because a host might be on multiple networks, there might be multiple logical host names for a single logical host.

logical network interface

In the Internet architecture, a host may have one or more IP addresses. HA configures additional logical network interfaces to establish a mapping between several logical network interfaces and a single physical network interface. This allows a single physical network interface to respond to multiple logical network interfaces. This also enables the IP address to move from one HA server to the other in the event of a takeover or `haswitch(1M)`, without requiring additional hardware interfaces.

M**MAC address**

(media access control address) The worldwide unique identifying address assigned to every Ethernet interface card.

master

The server with exclusive read and write access to a diskset. The current master host for the diskset runs the data service and has the logical IP addresses mapped to its Ethernet address.

mediator

In a dual-string configuration, provides a “third vote” in determining whether access to the metadvice state database replicas can be granted or must be denied. It is used only when exactly half of the metadvice state database replicas are accessible.

mediator host

A host that is acting in the capacity of a “third vote” by running the `rpc.metamed(1M)` daemon and that has been added to a diskset.

mediator quorum

The condition achieved when half + 1 of the mediator hosts are accessible.

membership monitor

A process running on all HA servers that monitors the servers. The membership monitor sends and receives heartbeats to its sibling hosts. The monitor can initiate a takeover if the heartbeat stops. It also keeps track of which servers are active.

metadvice

A group of components accessed as a single logical device by concatenating, striping, mirroring, or logging the physical devices. Metadevices are sometimes called pseudo devices.

metadvice state database

Information kept in nonvolatile storage (on disk) for preserving the state and configuration of metadevices.

metadvice state database replica

A copy of the state database. Keeping multiple copies of the state database protects against the loss of state and configuration information. This information is critical to all metadvice operations.

MI

Multi-initiator.

mirroring

Replicating all writes made to a single logical device (the mirror) to multiple devices (the submirrors), while distributing read operations. This provides data redundancy in the event of a failure.

multihomed host

A host that is on more than one public network.

multihost disk

A disk configured for potential accessibility from multiple servers. Sun™ Cluster 3.0 07/01 software enables data on a multihost disk to be exported to network clients using a highly available data service.

N

NAFO

Network adapter failover.

NFS

Network File System.

NIS

Network Information Service.

N-to-N topology

All nodes are directly connected to a set of shared disks.

N+1 topology

Some number (N) of active servers and one (+1) hot-standby server. The active servers provide on-going data services and the hot-standby server takes over data service processing if one or more of the active servers fail.

node

A physical machine that can be part of a Sun cluster. In Sun™ Cluster 3.0 07/01 documentation, it is synonymous with host or node.

nodelock

The mechanism used in greater than two-node clusters using Cluster Volume Manager or VERITAS Volume Manager to failure fence failed nodes.

NTP

Network Time Protocol.

NVRAM

Nonvolatile random access memory.

O

OBP

OpenBoot programmable read-only memory.

OGMS

Oracle Group Membership Services.

OLAP

Online Analytical Processing.

OLTP

Online Transaction Processing.

OPS

Oracle Parallel Server.

P**parallel database**

A single database image that can be accessed concurrently through multiple hosts by multiple users.

partial failover

Failing over a subset of logical hosts mastered by a single physical host.

PCI

Peripheral component interconnect.

PDB

Parallel database.

PGA

Program global area.

PMON

Process monitor.

PNM

Public Network Management.

potential master

Any physical host that is capable of mastering a particular logical host.

PROM

Programmable read-only memory.

primary logical host name

The name by which a logical host is known on the primary public network.

primary physical host name

The name by which a physical host is known on the primary public network.

primary public network

A name used to identify the first public network.

private links

The private network between nodes used to send and receive heartbeats between members of a server set.

Q

quorum device

In SSVM or CVM configurations, the system votes by majority quorum to prevent network partitioning. Because it is impossible for two nodes to vote by majority quorum, a quorum device is included in the voting. This device could be either a controller or a disk.

R

RAID

Redundant arrays of independent disks.

RDBMS

Relational database management system.

replica

See **metadevice state database replica**.

replica quorum

A Solstice DiskSuite concept; the condition achieved when $\text{HALF} + 1$ of the metadevice state database replicas are accessible.

ring topology

One primary and one backup server is specified for each set of data services.

RP

Remote probes.

ROM

Read-only memory.

RPC

Remote procedure call.

S

SAP

Service access point.

SCI

(scalable coherent interface) A high-speed interconnect used as a private network interface.

SCSI

Small Computer System Interface.

scalable topology

See **N-to-N topology**.

secondary logical host name

The name by which a logical host is known on a secondary public network.

secondary physical host name

The name by which a physical host is known on a secondary public network.

secondary public network

A name used to identify the second or subsequent public networks.

server

A physical machine that can be part of a Sun cluster. In Sun™ Cluster 3.0 07/01 documentation, it is synonymous with host or node.

SGA

System global area.

Sibling host

One of the physical servers in a symmetric HA configuration.

SIMM

Single inline memory module.

SNMP

Simple Network Management Protocol.

SOC

Serial optical channel.

Solstice DiskSuite

A software product that provides data reliability through disk striping, concatenation, mirroring, UFS logging, dynamic growth of metadevices and file systems, and metadvice state database replicas.

SPARC

Scalable Processor Architecture.

SSP

System service processor.

stripe

Similar to concatenation, except the addressing of the component blocks is non-overlapped and interlaced on the slices (partitions), rather than placed sequentially. Striping is used to gain performance. By striping data across disks on separate controllers, multiple controllers can access data simultaneously.

submirror

A metadvice that is part of a mirror. See also **mirroring**.

Sun™ Cluster 3.0 07/01

Software and hardware that enables several machines to act as read-write data servers while acting as backups for each other.

Switch Management Agent

The software component that manages sessions for the SCI and Ethernet links and switches.

switchover

The coordinated moving of a logical host from one operational HA server to the other. A switchover is initiated by an administrator using the `haswitch(1M)` command.

symmetric configuration

A two-node configuration where one server operates as the hot-standby server for the other.

T

takeover

The automatic moving of a logical host from one HA server to another after a failure has been detected. The HA server that has the failure is forced to give up mastery of the logical host.

TC

(terminal concentrator) A device used to enable an administrative workstation to securely communicate with all nodes in the Sun™ Cluster 3.0 07/01.

TCP

Transmission Control Protocol.

TCP/IP

Transmission Control Protocol/Internet Protocol.

TPE

Twisted-pair Ethernet.

trans device

In Solstice DiskSuite configurations, a pseudo device responsible for managing the contents of a UFS log.

U**UDLM**

UNIX distributed lock manager.

UDP

User Data Protocol.

UFS

UNIX file system.

UFS logging

Recording UFS updates to a log (the logging device) before the updates are applied to the UFS (the master device).

UFS logging device

In Solstice DiskSuite configurations, the component of a transdevice that contains the UFS log.

UFS master device

In Solstice DiskSuite configurations, the component of a transdevice that contains the UFS file system.

UPS

Uninterruptable Power Supply.

V**VMSA**

Volume Manager Storage Administrator. A VERITAS graphical storage administration application.

VxFS

Veritas file system.

