

CXFS™ MultiOS for CXFS Client-Only
Nodes: Installation and Configuration
Guide

007-4507-007

CONTRIBUTORS

Written by Lori Johnson

Illustrated by Chrystie Danzer

Production by Glen Traefald

Engineering contributions to the book by Vlad Apostolov, Neil Bannister, Dale Brantly, David Chatterton, Mark Cruciani, Tad Dolphay, Dave Ellis, Eric Epe, Andrew Gildfind, Dennis Kender, Aaron Mantel, Troy McCorkell, Ken McDonell, Bill Mckevitt, Terry Merth, Daniel Moore, Max Matveev, Fujio Nakajima, Barry Naujok, Mike Raskie, Eric Sandeen, Tim Sirianni, Wesley Smith, Michael Umansky, Madan Valluri, Geoffrey Wehrman

COPYRIGHT

© 2002, 2003 Silicon Graphics, Inc. All rights reserved; provided portions may be copyright in third parties, as indicated elsewhere herein. No permission is granted to copy, distribute, or create derivative works from the contents of this electronic documentation in any manner, in whole or in part, without the prior written permission of Silicon Graphics, Inc., in the United States and/or other countries worldwide.

LIMITED RIGHTS LEGEND

The electronic (software) version of this document was developed at private expense; if acquired under an agreement with the USA government or any contractor thereto, it is acquired as "commercial computer software" subject to the provisions of its applicable license agreement, as specified in (a) 48 CFR 12.212 of the FAR; or, if acquired for Department of Defense units, (b) 48 CFR 227-7202 of the DoD FAR Supplement; or sections succeeding thereto. Contractor/manufacturer is Silicon Graphics, Inc., 1600 Amphitheatre Pkwy 2E, Mountain View, CA 94043-1351.

TRADEMARKS AND ATTRIBUTIONS

Silicon Graphics, SGI, the SGI logo, IRIX, O2, Origin, and XFS are registered trademarks and CXFS, FailSafe, IRIS FailSafe, SGI Altix, SGI ProPack, and Trusted IRIX are trademarks of Silicon Graphics, Inc., in the United States and/or other countries worldwide.

Active Directory, Microsoft, Windows, and Windows NT are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. AIX and IBM are registered trademarks of IBM Corporation. Brocade and Silkworm are trademarks of Brocade Communication Systems, Inc. AMD, AMD Athlon, and AMD Duron are trademarks of Advanced Micro Devices, Inc. Disk Manager is a registered trademark of ONTRACK Data International, Inc. FibreStar and JNI are registered trademarks and EZ Fibre is a trademark of JNI Corporation. LSI Logic is a trademarks or registered trademark of LSI Logic Corp. FLEXIm is a registered trademark of Macrovision Corporation. HP-UX is a trademark of Hewlett-Packard Company. InstallShield is a registered trademark of InstallShield Software Corporation in the United States and/or other countries. Intel and Pentium are registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. Legato NetWorker is a registered trademark of Legato Systems, Inc. Linux is a registered trademark of Linus Torvalds. OpenLDAP is a registered trademark of OpenLDAP Foundation. Red Hat is a registered trademark and RPM is a trademark of Red Hat, Inc. SANsurfer and QLogic are registered trademarks of QLogic Corporation. Sun, Sun Fire, SunOS, Solaris, and Ultra Enterprise are trademarks or registered trademarks of Sun Microsystems, Inc. UNIX and the X device are registered trademarks of The Open Group in the United States and other countries.

Screen snaps of the EZ Fibre product are provided by express permission of JNI Corporation.

Cover design by Sarah Bolles, Sarah Bolles Design, and Dany Galgani, SGI Technical Publications.

New Features in this Guide

This guide contains the following new features:

- Support for the IBM AIX platform, Linux on supported IA32 platforms, SGI ProPack for Linux on Altix servers.
- Support for a cluster of up to 48 nodes, 16 of which can be CXFS administration nodes; the rest must be client-only nodes. See "Requirements", page 6.
- For Windows nodes, user identification with lightweight directory access protocol (LDAP); see "User Identification Mapping Methods", page 117.
- Support of forced unmount of filesystems on Windows nodes; see "Forced Unmount on a Windows Node", page 114.
- Information about protecting data integrity if JNI Fibre Channel cables are disconnected or fail; see "Protecting Data Integrity", page 85.
- Support for the SGI TP9500 RAID; see "Required SGI TP9500 RAID Firmware", page 17.
- Support for the QLogic 2342 host bus adapter; see "Requirements Specific to Windows", page 110
- Information about new `cxfs-reprobe` scripts on AIX, IRIX, Linux, and Solaris nodes. These scripts are run by either `clconfd` or `cxfs_client` when they need to reprobe the Fibre Channel controllers. The administrator may modify these scripts if needed. See "Cluster Administration", page 3.
- Information about setting the `ntcp_nodelay` system tunable parameter in order to provide adequate performance on file deletes; see "Requirements", page 6.
- Automatic detection of HBAs is provided for Linux, Solaris, and Windows nodes. See:
 - "Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File", page 59
 - "Postinstallation Steps for Solaris: Creating the I/O Fencing File", page 104
 - "Creating the Windows I/O Fencing File", page 147

Record of Revision

Version	Description
001	March 2002 Original publication with the CXFS MultiOS Clients 2.0 release for IRIX 6.5.16f.
002	May 2002 Revised to support the CXFS MultiOS Clients 2.1 release for IRIX 6.5.16f. This release supports the Sun Microsystems Solaris and Microsoft Windows NT platforms.
003	June 2002 Revised to support the CXFS MultiOS Clients 2.1.1 release for IRIX 6.5.16f. This release supports the Sun Microsystems Solaris and Microsoft Windows NT platforms.
004	August 2002 Revised to support the CXFS MultiOS 2.2 Clients release for IRIX 6.5.17f. This release supports the Sun Microsystems Solaris, Microsoft Windows NT, and Microsoft Windows 2000 platforms.
005	November 2002 Revised to support the CXFS MultiOS Clients 2.3 release for IRIX 6.5.18f. This release supports the Sun Microsystems Solaris, Microsoft Windows NT, and Microsoft Windows 2000 platforms.
006	February 2003 Revised to support the CXFS MultiOS Clients 2.4 release for IRIX 6.5.19f. This release supports the Sun Microsystems Solaris, Microsoft Windows NT, and Microsoft Windows 2000 platforms.
007	May 2003 Revised to support the CXFS MultiOS Clients 2.5 release for IRIX 6.5.20f. This release supports the IBM AIX platform, Linux on supported IA32 platforms, SGI ProPack for Linux on SGI Altix platforms, Sun Microsystems Solaris platform, Microsoft Windows NT platform, and Microsoft Windows 2000 platform.

Contents

About This Guide	xix
Prerequisites	xix
Related Publications	xix
Obtaining Publications	xxii
Conventions	xxii
Reader Comments	xxiii
1. Introduction	1
When to Use CXFS	2
CXFS on Client-Only Nodes	2
CXFS Processes	3
Licenses	3
Cluster Administration	3
User Administration for CXFS	4
Performance Considerations	5
Requirements	6
Recommendations	8
Overview of the Installation and Configuration Steps	10
AIX Overview	11
Linux on IA32 Platforms Overview	12
SGI ProPack for Linux on SGI Altix Overview	13
Solaris Overview	14
Windows Overview	15
2. SGI RAID Firmware	17
Required SGI RAID Firmware	17
007-4507-007	vii

Required SGI TP9500 RAID Firmware	17
Required SGI TP9400 RAID Firmware	17
Required SGI TP9100 RAID Firmware	18
RAID Firmware Verification	18
For More Information	18
3. Brocade Fibre Channel Switch Verification	19
Required Brocade Fibre Channel Switch Firmware and License	19
Verifying the Brocade License	20
Verifying the Brocade Switch Firmware Version	20
Configuring the Brocade Silkworm 3900	21
Configuring the Brocade Silkworm 12000	21
Changing the Brocade FC Cable Connections	22
4. Obtaining CXFS and XVM FLEXlm Licenses	25
Obtaining the Host Information Required for the License	25
AIX Host Information	25
Linux on IA32 Platforms Host Information	26
SGI ProPack Host Information	27
Solaris Host Information	27
Windows Host Information	28
Obtaining and Installing the Licenses	28
For More Information	29
5. AIX Platform	31
CXFS on AIX	31
Requirements Specific to AIX	31
CXFS Commands Installed on AIX	32
Log Files on AIX	33

Limitations on AIX	33
Maximum CXFS Filesystem and File Size on AIX	34
Access Control Lists and AIX	35
FLEXlm License Verification for AIX	35
Host Bus Adapter Installation and Configuration for AIX	36
Preinstallation Steps for AIX	36
Hostname Resolution and Network Configuration Rules for AIX	37
Adding a Private Network for AIX Nodes	37
Verifying the Private and Public Network for AIX	40
Client Software Installation Steps for AIX	41
AIX Installation Overview	41
Verifying the AIX Installation	43
Postinstallation Steps for AIX: Creating the I/O Fencing File	43
Manual CXFS Start/Shutdown for AIX	45
Software Maintenance for AIX	45
Upgrading the CXFS Software on an AIX System	46
Modifying the CXFS Software on an AIX System	46
6. Linux on IA32 Platforms	47
CXFS on Linux on IA32 Platforms	47
Requirements Specific to Linux on IA32 Platforms	48
CXFS Commands Installed on Linux on IA32 Platforms	49
Log Files on Linux on IA32 Platforms	49
Limitations and Considerations for Linux on IA32 Platforms	49
Maximum CXFS Filesystem and File Size on Linux on IA32 Platforms	50
Access Control Lists and Linux on IA32 Platforms	50
FLEXlm License Verification for Linux on IA32 Platforms	50
Host Bus Adapter Installation and Configuration for Linux on IA32 Platforms	51

Preinstallation Steps for Linux on IA32 Platforms	51
Hostname Resolution and Network Configuration Rules for Linux	52
Adding a Private Network for Linux Nodes	52
Verifying the Private and Public Networks for Linux	56
Client Software Installation Steps for Linux on IA32 Platforms	57
Installation Overview	57
Verifying the Linux Installation	59
Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File	59
Manual CXFS Startup/Shutdown for Linux on IA32 Platforms	60
Software Maintenance: Modifying the CXFS Software on a Linux on IA32 Platforms	60
7. SGI ProPack for Linux Platform	63
CXFS on SGI ProPack for Linux	63
Requirements Specific to SGI ProPack for Linux	64
CXFS Commands Installed on SGI ProPack for Linux	64
Log Files on SGI ProPack for Linux	64
Limitations and Considerations for SGI ProPack for Linux	65
Maximum CXFS Filesystem and File Size on SGI ProPack for Linux	65
Access Control Lists and SGI ProPack for Linux	66
FLEXlm License Verification for SGI ProPack for Linux	66
Host Bus Adapter Installation and Configuration for SGI ProPack for Linux	66
Preinstallation Steps for SGI ProPack for Linux	67
Hostname Resolution and Network Configuration Rules for SGI ProPack for Linux	67
Adding a Private Network for SGI ProPack for Linux Nodes	68
Verifying the Private and Public Networks for SGI ProPack for Linux	71
Client Software Installation Steps for SGI ProPack for Linux	72
Installation Overview	72
Verifying the SGI ProPack for Linux Installation	73

Postinstallation Steps for SGI ProPack for Linux	73
Manual CXFS Startup/Shutdown on SGI ProPack for Linux	74
Software Maintenance: Modifying the CXFS Software on an SGI ProPack for Linux System	74
8. Solaris Platform	77
CXFS on Solaris	77
Requirements Specific to Solaris	78
CXFS Commands Installed on Solaris	79
Log Files on Solaris	79
Limitations and Considerations on Solaris	79
Maximum CXFS Filesystem and File Size on Solaris	81
Access Control Lists and Solaris	81
FLEXlm License Verification for Solaris	82
Host Bus Adapter Installation and Configuration for Solaris	83
Installing the JNI HBA	83
Protecting Data Integrity	85
Installing and Running the EZ Fibre Configuration GUI	85
Verifying the JNI HBA Installation	94
Preinstallation Steps for Solaris	96
Hostname Resolution and Network Configuration Rules for Solaris	96
Adding a Private Network for Solaris Nodes	97
Verifying the Private and Public Networks for Solaris	102
Client Software Installation Steps for Solaris	102
Solaris Installation Overview	103
Verifying the Solaris Installation	104
Postinstallation Steps for Solaris: Creating the I/O Fencing File	104
Manual CXFS Startup/Shutdown for Solaris	106
Software Maintenance for Solaris	107

Upgrading the CXFS Software on a Solaris System	107
Modifying the CXFS Software on a Solaris System	107
9. Windows NT and Windows 2000 Platforms	109
CXFS on Windows	109
Requirements Specific to Windows	110
CXFS Commands Installed on Windows	111
Windows Log Files and Cluster Status	111
Functional Limitations Specific to Windows	112
UNIX Perspective of CXFS on a Windows Node	112
Windows Perspective of CXFS on a Windows Node	113
Forced Unmount on a Windows Node	114
Maximum CXFS Filesystem and File Size on Windows	114
Performance Considerations on a CXFS Windows Node	114
Access Controls on a Windows Node	115
User Identification on a Windows Node	116
User Identification Mapping Methods	117
Enforcing Access to Files and Directories	119
Viewing and Changing File Attributes with Windows Explorer	120
Viewing and Changing File Permissions with Windows Explorer	121
Windows NT	121
Windows 2000	122
Viewing and Changing File Access Control Lists (ACLs)	124
Windows NT	124
Windows 2000	125
Effective Access	126
Windows NT	126
Windows 2000	126
Restrictions with file ACLs on Window nodes	126

Inheritance and Default ACLs on a Windows node	127
Host Bus Adapter Installation for Windows	129
Confirming the QLogic HBA Installation	130
Support for More than 8 LUNs under Windows NT	130
Configuring Two HBAs for Failover Operation	131
Preinstallation Steps for Windows	133
Upgrading the QLogic BIOS	134
Hostname Resolution and Network Configuration Rules for Windows	135
Adding a Private Network for Windows Nodes	135
Adding a Private Network for Windows NT Nodes	135
Adding a Private Network for Windows 2000 Nodes	136
Verifying the Private and Public Networks for Windows	137
Client Software Installation Steps for Windows	138
Postinstallation Steps for Windows	145
Configuring the FLEXlm License for Windows	146
Creating the Windows I/O Fencing File	147
Performing User Configuration	149
Checking Permissions on the Password, Group, and Fencing Files	150
Creating a New Hardware Profile	151
Manual CXFS Startup/Shutdown for Windows	154
Software Maintenance for Windows	154
Modifying the CXFS for Windows Software	155
Upgrading the CXFS Software on a Windows System	156
Removing the CXFS Software from a Windows System	157
Downgrading the CXFS Software on a Windows System	157
10. Cluster Configuration	159

Defining the Client-Only Nodes	159
Adding the Client-Only Nodes to the Cluster	161
Defining the Switch for I/O Fencing	162
Starting CXFS Services on the Client-Only Nodes	164
Verifying LUN Masking	164
Mounting Filesystems on the Client-Only Nodes	164
Restarting the Windows Node	165
Verifying the Cluster	165
Forced Unmount of CXFS Filesystems	167
11. Troubleshooting	169
Identifying Problems on AIX Nodes	169
Is the AIX Node in the Cluster?	169
Are there Error Messages for the AIX Node?	169
Identifying Problems on Linux on IA32 Nodes	170
Is the Linux Node in the Cluster?	170
Are there Error Messages for the Linux Node?	170
Identifying Problems on SGI ProPack Nodes	171
Is the SGI ProPack Node in the Cluster?	171
Are there Error Messages for the SGI ProPack Node?	171
Identifying Problems on Solaris Nodes	171
Is the Solaris Node in the Cluster?	171
Are there Error Messages for the Solaris Node?	172
Identifying Problems on Windows Nodes	172
Is the CXFS Software Running Correctly on the Windows Node?	172
Windows NT CXFS Software Verification	172
Windows 2000 CXFS Software Verification	173
Is the Windows Node in the Cluster?	174

Are There Error Messages for the Windows Node?	174
Windows Error Message Explanations	175
Common Problems and Solutions	177
Incorrect Configuration	177
Determining If a Client-Only Node Is Fenced	177
Common Host Bus Adapter Problems	177
Common Linux on IA32 Problems	178
Common Windows Problems	179
Filesystems Are Not Displayed on a Windows Node	179
Large Log Files on Windows	179
Windows Failure on Restart	180
Memory Configuration of the Windows Node	180
Reporting Problems	180
Reporting AIX Problems	181
Reporting Linux on IA32 Problems	182
Reporting SGI ProPack Problems	182
Reporting Solaris Problems	183
Reporting Windows Problems	184
Retain Windows NT Information	185
Retain Windows 2000 Information	186
Save Crash Dumps for Windows	187
Glossary	189
Index	195

Figures

Figure 8-1	Example: Second Window: EZ Fibre Configuration Utility - Standalone . . .	87
Figure 8-2	Location of icon (+) to Display the HBA	88
Figure 8-3	Example: After Clicking + to Display the HBA	89
Figure 8-4	Location of the Icon to Display the Adapter Parameters	90
Figure 8-5	Example: After Clicking the HBA Icon to Show the Adapter Parameters . . .	91
Figure 8-6	After Clicking the Adapter Information Tab	92
Figure 8-7	After Clicking on LUN-Level Zoning	93
Figure 8-8	Example: After Mapping the LUNs and Committing the Changes	94
Figure 9-1	Select All Software Components	140
Figure 9-2	Enter CXFS Details	141
Figure 9-3	Active Directory Details	142
Figure 9-4	Generic LDAP Details	143
Figure 9-5	Review the Settings	144
Figure 9-6	Start CXFS Driver	145
Figure 9-7	Modify the CXFS for Windows	156

About This Guide

This publication documents the CXFS MultiOS Clients 2.5 release for IRIX 6.5.20f. This release supports IBM AIX, Linux on supported IA32 platforms, SGI ProPack for Linux on SGI Altix servers, Sun Microsystems Solaris, Microsoft Windows NT, and Microsoft Windows 2000 nodes.

Prerequisites

This guide assumes the following:

- The IRIX CXFS cluster is installed and operational.
- The CXFS client-only nodes have the appropriate platform-specific operating system software installed.
- The reader is familiar with the information presented in the *CXFS Version 2 Software Installation and Administration Guide* and the platform's operating system documentation.

Related Publications

The following documents contain additional information (if you are viewing this document online, you can click on TPL Link below to link to the book on the SGI TechPubs library):

- CXFS documentation:
 - Platform-specific release notes
 - *CXFS Version 2 Software Installation and Administration Guide* (TPL link)
- SGI TP9400 and SGI TP9500 documentation :
 - *SGI TP9400 and SGI TP9500 Software Concepts Guide* (TPL link)
 - *SGI TP9400 and SGI TP9500 RAID Owner's Guide* (TPL link)
 - *SGI TP9400 and SGI TP9500 RAID Administration Guide* (TPL link)

The SGI TP9400 documentation is available on the release CD in the following files:

- `tp9400_sw_concepts_guide.pdf`
- `tp9400_owners_guide.pdf`
- `tp9400_admin_guide.pdf`
- SGI TP9100:
 - *TPM Installation Instructions and User's Guide for TP9100*
- JNI host bus adapter (HBA) card and driver documentation:
 - *Installation Guide, FCE-6460 and FCE2-6460 PCI-to-Fibre Channel Host Bus Adapters (Solaris, Windows NT/2000, Novell, AIX, HP-UX, MAC-OS, Linux) JNI FibreStar*
 - *Quick Installation Guide, Solaris, AIX and Windows JNI EZ Fibre*

Also see the JNI website at:

<http://www.jni.com>

- QLogic HBA card and driver documentation:
 - *Hardware Installation Guide for the QLA2xxx Board Family*
 - *Software Installation Guide for the QLA2xxx Board Family*

See the QLogic website at:

<http://www.qlogic.com>

- AIX documentation:
 - *AIX 5L V5.1 Installation Guide*
 - *AIX V5.1 Network Installation Management Guide and Reference*
 - *AIX 5L V5.1 Installation in a Partitioned Environment Guide*

See the IBM website at:

<http://www.ibm.com>

- Linux on IA32 platforms documentation:
 - *Red Hat Linux 7.3 Release Notes*
 - *The Official Red Hat Linux x86 Installation Guide*
 - *The Official Red Hat Linux Reference Guide*
- SGI ProPack and SGI Altix documentation:
 - *SGI ProPack for Linux Start Here*
 - *SGI Altix 3000 User's Guide*
- Solaris 8 documentation:
 - *Solaris 8 Installation Guide*
 - *Solaris 8 System Administration Collection*
 - *Solaris 8 Advanced Installation Guide*

See the Sun Microsystems website at:

<http://www.sun.com>
- Solaris 9 documentation:
 - *Solaris 9 Installation Guide*
 - *Solaris 9 System Administration Collection*

See the Sun Microsystems website at:

<http://www.sun.com>
- Sun Microsystems hardware documentation:
 - *Ultra Enterprise 450 Systems Manual*
 - *Ultra Enterprise 6000/5000/4000 Systems Manual*
 - *Ultra Enterprise 6000/5000/4000 Systems Installation Guide*
 - *Ultra Enterprise 10000 SSP 3.0 User's Guide*

- Windows software documentation:
 - *Start Here Microsoft Windows NT Workstation: Basics and Installation*
 - *Microsoft Windows 2000 Quick Start Guide*
- Hardware documentation for the Intel platform
- *Flexible License Manager End User Manual* from Macrovision Corporation.

Obtaining Publications

You can obtain SGI documentation in the following ways:

- See the SGI Technical Publications Library at <http://docs.sgi.com>. Various formats are available. This library contains the most recent and most comprehensive set of online books, release notes, man pages, and other information.
- If it is installed on your SGI system, you can use InfoSearch, an online tool that provides a more limited set of online books, release notes, and man pages. With an IRIX system, select **Help** from the Toolchest, and then select **InfoSearch**. Or you can type `infosearch` on a command line.
- You can also view release notes by typing either `grelnotes` or `relnotes` on a command line.
- You can also view man pages by typing `man title` on a command line.

Conventions

Note: This guide uses *Windows* to refer to both Microsoft Windows NT and Microsoft Windows 2000 nodes when the information applies equally to both. Information that applies to only one of these types of nodes is identified.

The following conventions are used throughout this document:

Convention	Meaning
<code>command</code>	This fixed-space font denotes literal items such as commands, files, routines, path names, signals, messages, and programming language structures.
<i>variable</i>	Italic typeface denotes variable entries and words or concepts being defined.
user input	This bold, fixed-space font denotes literal items that the user enters in interactive sessions. (Output is shown in nonbold, fixed-space font.)
GUI	This font denotes the names of graphical user interface (GUI) elements such as windows, screens, dialog boxes, menus, toolbars, icons, buttons, boxes, fields, and lists.
[]	Brackets enclose optional portions of a command or directive line.
...	Ellipses indicate that a preceding element can be repeated.

Reader Comments

If you have comments about the technical accuracy, content, or organization of this publication, contact SGI. Be sure to include the title and document number of the publication with your comments. (Online, the document number is located in the front matter of the publication. In printed publications, the document number is located at the bottom of each page.)

You can contact SGI in any of the following ways:

- Send e-mail to the following address:
techpubs@sgi.com
- Use the Feedback option on the Technical Publications Library Web page:
<http://docs.sgi.com>
- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.

- Send mail to the following address:

Technical Publications
SGI
1600 Amphitheatre Parkway, M/S 535
Mountain View, California 94043-1351

- Send a fax to the attention of "Technical Publications" at +1 650 932 0801.

SGI values your comments and will respond to them promptly.

Introduction

This guide provides an overview of the installation and configuration procedures for CXFS client-only nodes running SGI CXFS clustered filesystems. A *CXFS client-only node* runs a subset of CXFS processes and services.

This release supports client-only nodes running the following operating systems:

- IBM AIX 5L version 5.1
- Red Hat 7.3 Linux on supported IA32 platforms (see "Requirements Specific to Linux on IA32 Platforms", page 48)
- SGI ProPack 2.2 for Linux on the SGI Altix platform
- Sun Microsystems Solaris 8
- Sun Microsystems Solaris 9
- Microsoft Windows NT 4.0 Service Pack 6
- Microsoft Windows 2000 Service Pack 2 or later

Note: This guide uses *Windows* to refer to both Microsoft Windows NT and Microsoft Windows 2000 nodes when the information applies equally to both. Information that applies to only one of these types of nodes is identified.

A cluster running multiple operating systems is known as a *multiOS cluster*.

Many of the procedures mentioned in this guide will be performed by SGI personnel or other qualified service personnel. Details for these procedures are provided in other documents.



Caution: CXFS is a complex product. To ensure that CXFS is installed and configured in an optimal manner, it is **mandatory** that you purchase SGI installation services developed for CXFS. Contact your local SGI sales representative for details.

For general information about CXFS terminology, concepts, and configuration, see the *CXFS Version 2 Software Installation and Administration Guide*.

This chapter discusses the following:

- "When to Use CXFS"
- "CXFS on Client-Only Nodes"
- "Overview of the Installation and Configuration Steps", page 10

When to Use CXFS

You should use CXFS when you have multiple hosts running applications that require high-bandwidth access to common filesystems.

CXFS performs best under the following conditions:

- Data I/O operations are greater than 16 KB.
- All processes that perform read/write operations for a given file reside on the same host.
- Multiple processes on multiple hosts read the same file.
- Direct-access I/O is used for read/write operations for multiple processes on multiple hosts.
- Large files and file accesses are being used.

For most filesystem loads, the preceding scenarios represent the bulk of the file accesses. Thus, CXFS delivers fast local file performance. CXFS is also useful when the amount of data I/O is larger than the amount of meta-data I/O. (*Meta data* is information that describes a file, such as the file's name, size, location, and permissions.) CXFS is faster than NFS because the data does not go through the network.

CXFS on Client-Only Nodes

This section contains the following:

- "CXFS Processes", page 3
- "Licenses", page 3
- "Cluster Administration", page 3

- "User Administration for CXFS", page 4
- "Performance Considerations", page 5
- "Requirements", page 6
- "Recommendations", page 8

CXFS Processes

When CXFS is started on a client-only node, a user-space daemon/service is started that provides the required processes. This is a subset of the processes needed on an IRIX node.

Licenses

You must have the following licenses:

- Brocade license. See "Required Brocade Fibre Channel Switch Firmware and License", page 19.
- CXFS FLEXlm license installed on every node in the cluster; see Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.

Note: XVM provides a mirroring feature. If you want to access a mirrored volume from a given node in the cluster, you must install a FLEXlm mirroring license on that node. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your SGI sales representative.

Cluster Administration

There must be at least one IRIX server-capable administration node in the cluster that is responsible for updating that filesystem's meta data. This node is referred to as the *CXFS meta-data server*. Only IRIX server-capable administration nodes can be meta-data servers; client-only nodes cannot be meta-data servers. The CXFS cluster database is not stored on client-only nodes; only IRIX administration nodes contain the cluster database.

An IRIX administration node is required to perform administrative tasks, using either the `cmgr(1M)` command or the CXFS graphical user interface (GUI). For more

information about using these tools, see the *CXFS Version 2 Software Installation and Administration Guide*.

Note: The NFS export scripts are supported on AIX, IRIX, Linux, SGI ProPack, and Solaris nodes; they are not supported on Windows nodes. The scripts behave the same on these platforms, but the pathnames differ between client-only nodes and administration nodes:

On client-only nodes running AIX, IRIX, Linux, SGI ProPack, and Solaris:

```
/var/cluster/cxfs_client-scripts/cxfs-pre-mount  
/var/cluster/cxfs_client-scripts/cxfs-post-mount  
/var/cluster/cxfs_client-scripts/cxfs-pre-umount  
/var/cluster/cxfs_client-scripts/cxfs-post-umount
```

On administration nodes running IRIX:

```
/var/cluster/clconfd-scripts/cxfs-pre-mount  
/var/cluster/clconfd-scripts/cxfs-post-mount  
/var/cluster/clconfd-scripts/cxfs-pre-umount  
/var/cluster/clconfd-scripts/cxfs-post-umount
```

The following `cxfs-probe` scripts are run by either `clconfd` or `cxfs_client` when they need to reprobe the Fibre Channel controllers:

```
/var/cluster/cxfs_client-scripts/cxfs-reprobe  
/var/cluster/clconfd-scripts/cxfs-reprobe
```

The administrator may modify these scripts if needed.

For information about using these scripts, see the *CXFS Version 2 Software Installation and Administration Guide*.

User Administration for CXFS

A CXFS cluster requires a consistent user identification scheme across all hosts in the cluster so that one person using different cluster nodes has the same access to the files on the cluster.

The following must be observed to achieve this consistency:

- Users must have the same usernames on all nodes in the cluster. An individual user identifier (UID) should not be used by two different people anywhere in the cluster. Ideally, group names should also be the same on all nodes in the cluster.
- Each CXFS client and server node must have access to the same UID and GID information. The simplest way to achieve this is to maintain the same `/etc/passwd` and `/etc/group` files on all CXFS nodes, but other mechanisms may be supported on different platforms.

Performance Considerations

CXFS may not give optimal performance under the following circumstances:

- When you are using NFS to export a CXFS filesystem from a CXFS client. Performance will be much better when the export is performed from the active CXFS meta-data server than when it is performed from a CXFS client node.
- When access would be as slow with CXFS as with network filesystems, such as with the following:
 - Small files.
 - Low bandwidth.
 - Lots of meta-data transfer. Meta-data operations can take longer to complete through CXFS than on local filesystems. Meta-data transaction examples include the following:
 - Opening and closing a file
 - Changing file size (usually extending a file)
 - Creating and deleting files
 - Searching a directory

In addition, multiple processes on multiple hosts that are reading and writing the same file using buffered I/O can be slower when using CXFS than when using a local filesystem. This performance difference comes from maintaining coherency among the distributed file buffers; a write into a shared, buffered file will invalidate data (pertaining to that file) that is buffered in other hosts.

- When distributed applications write to shared files that are memory mapped.

Also see "Functional Limitations Specific to Windows", page 112.

Requirements

Using a client-only node in a CXFS cluster requires the following:

- A supported SAN hardware configuration.

Note: For details about supported hardware, see the Entitlement Sheet that accompanies the base CXFS release materials. Using unsupported hardware constitutes a breach of the CXFS license. CXFS does **not** support the Silicon Graphics O2 workstation as a CXFS node nor does it support JBOD.

- At least one IRIX server-capable administration node to act as the meta-data server and from which to perform cluster administration tasks. CXFS should be installed on the IRIX administration node before CXFS is installed on the client-only nodes.
- A private 100baseT (or greater) TCP/IP network connected to each node, to be dedicated to the CXFS private heartbeat and control network. All nodes must be configured to use the same subnet.
- IRIX 6.5.20f or later, plus any required patches. For more details, see the platform-specific CXFS MultiOS Clients release notes.
- A FLEXlm license key for CXFS and optionally XVM. The CXFS license is required for all nodes in the pool; a license is required for each node from which you want to access a mirrored XVM volume.
- A Brocade Fibre Channel 2400, 2800, 3800, 3900, or 12000 switch that is supported by SGI. The switch is required to protect data integrity.

IRIX nodes with system controllers use serial reset lines or I/O fencing to protect the integrity of the data stored in the cluster. (One of these methods is mandatory for the IRIX nodes in a cluster with only two server-capable nodes. Larger clusters should have an odd number of server-capable nodes.)

The *I/O fencing* feature isolates a problem node so that it cannot access I/O devices and therefore cannot corrupt data in the shared CXFS filesystem. This feature can only be used with a Brocade Fibre Channel switch; therefore, the Brocade switch is a required piece of hardware in a multiOS cluster.

I/O fencing differs from zoning:

- *Fencing* is a generic cluster term that means to erect a barrier between a host and shared cluster resources.
- *Zoning* is the ability to define logical subsets of the switch (zones), with the ability to include or exclude hosts and media from a given zone. A host can only access media that are included in its zone. Zoning is one possible implementation of fencing.

Zoning implementation is complex and does not have uniform availability across switches. Therefore, SGI chose to implement a simpler form of fencing, enabling/disabling a host's Brocade ports.

If there are problems with a node, the I/O fencing software sends a message via the `telnet` protocol to the appropriate Fibre Channel switch. The switch only allows one `telnet` session at a time; therefore, if you are using I/O fencing, you must keep the `telnet` port on the Fibre Channel switch free at all times.



Caution: Do not perform a `telnet` to the switch and leave the session connected.

- At least one IRIX node must be a server-capable administration node in order to be a CXFS meta-data server; other nodes can be CXFS client-only nodes. A cluster may contain as many as 48 nodes, of which as many as 16 can be administration nodes; the rest must be client-only nodes. All AIX, Linux, Solaris, and Windows nodes are CXFS client-only nodes.

A cluster in which both CXFS and IRIS FailSafe 2.1 or later are run (known as *coexecution*) is supported with a maximum of 48 nodes, as many as 8 of which can run FailSafe. However, FailSafe cannot run on AIX, Linux, Solaris, or Windows nodes.

- No nodes within the cluster running Trusted IRIX. A multiOS cluster cannot contain Trusted IRIX nodes.
- Ensure that the appropriate IRIX software is installed on the potential meta-data server nodes. For example, if you want to use quotas and access control lists (ACLs) on any cluster node, the `coe.sw.quotas`, `nfs.sw.acl_nfs`, and `coe.sw.acl` subsystems must be installed on the IRIX nodes listed as potential meta-data servers. For more information, see *IRIX Admin: Disks and Filesystems*, *IRIX Admin: Backup, Security and Accounting*, and your site's IRIX system administrator.

- Set the `ntcp_nodelay` system tunable parameter to 1 on IRIX potential meta-data servers in order to provide adequate performance on file deletes.

Also see "Requirements Specific to Solaris", page 78, and "Requirements Specific to Windows", page 110.

Recommendations

SGI recommends the following when running CXFS on a client-only node:

- Fix any network issues on the private network before trying to use CXFS.
- Use an Ethernet network switch rather than a hub for performance and control.
- Configure a production cluster with an odd number of server-capable administration nodes.
- For large clusters, SGI recommends that you define only the first three server-capable administration nodes and then continue on with the steps to define the cluster. After you have a successful small cluster, go back and add the remaining nodes.
- Launch any task initiated using `cron` on a CXFS filesystem from a single node in the cluster, preferably from the active meta-data server.

The `cron` daemon can cause severe stress on a CXFS filesystem if multiple nodes in a cluster start the same filesystem-intensive task simultaneously. An example of such a task is one that uses the `find` command to search files in a filesystem.

- Do not run any defragmentation software on CXFS filesystems. This includes the IRIX `fsr(1M)` command and any similar commands on other supported operating systems.
- Be very careful when running IRIX `xfs_repair` on CXFS filesystems. Only use `xfs_repair` on meta-data servers and only when you have verified that all other cluster nodes have unmounted the filesystem. SGI recommends that you contact SGI technical support before using `xfs_repair`. For more details, see the *CXFS Version 2 Software Installation and Administration Guide*.
- Only those nodes that you want to be potential meta-data servers should be CXFS administration nodes (installed with the `cluster_admin` software product). CXFS client administration nodes should only be used when necessary for coexecution with IRIS FailSafe. All other nodes should be client-only nodes (installed with `cxfs_client`). Use an odd number of server-capable administration nodes.

- Shut down cluster services before maintenance.
- In this release, relocation is disabled by default and recovery is supported only when using standby nodes.

A *standby node* is a meta-data server-capable administration node that is configured as a potential meta-data server for a given filesystem, but does not currently run any applications that will use that filesystem. To use recovery, you must not run any applications on any of the potential meta-data servers for a given filesystem; after the active meta-data server has been chosen by the system, you can then run applications that use the filesystem on the active meta-data server and client-only nodes.

Relocation and recovery are fully implemented, but the number of associated problems prevents full support of these features in the current release. Although data integrity is not compromised, cluster node panics or hangs are likely to occur. Relocation and recovery will be fully supported in a future release when these issues are resolved.

- Use the following good practices:
 - Unmount the filesystems from the meta-data server, shut down the node, and remount the filesystem when possible.
 - Do the following before shutting down a node:
 - Unmount filesystems.
 - Shut down cluster services.
- Do not run power management software, which may interfere with the CXFS cluster.
- Enable the *forced unmount* feature for CXFS filesystems, which is turned off by default. Normally, an unmount operation will fail if any process has an open file on the filesystem. However, a forced unmount allows the unmount to proceed regardless of whether the filesystem is still in use.

Many sites have found that enabling this feature improves the stability of their CXFS cluster, particularly in situations where the filesystem must be unmounted.

The method used to implement this feature is platform-specific:

- On IRIX nodes, this feature uses the `umount -k` option. The `-k` option attempts to kill processes that have open files or current directories in the

appropriate filesystems and then unmount them. That is, it attempts to terminate any I/O going to the filesystem, so that it can unmount it promptly, rather than having to wait for the I/O to finish on its own, causing the unmount to possibly fail.

- On AIX nodes, a similar function is performed with the `fuser -k` command and `umount` command.
- On Linux and SGI ProPack nodes, a similar function is performed with the `fuser -m -k` command and the `umount` command.
- On Solaris nodes, a similar function is performed with the `fuser -c -k` command and the `umount -f` command.
- On Windows nodes, all processes with open files on the CXFS filesystem are killed. For details, see "Forced Unmount on a Windows Node", page 114.

This feature is available on an IRIX node with the following CXFS GUI menu:

Tasks

> **Filesystems**

> **Unmount a Filesystem**

In the CXFS GUI, click the **Force** toggle in the **Unmount Filesystem** task.

You can also specify this feature using the `cmgr(1M)` commands to define the filesystem. For more information, see "Forced Unmount of CXFS Filesystems", page 167.

You must use `cmgr` from an IRIX node, and the GUI must be connected to an IRIX node.

For more information, see the *CXFS Version 2 Software Installation and Administration Guide*, and the `fuser` and `umount` man pages.

- Enable system dumps; see the operating system documentation for instructions.

Overview of the Installation and Configuration Steps

This section provides an overview of the installation, verification, and configuration steps for each platform type:

- "AIX Overview", page 11

- "Linux on IA32 Platforms Overview", page 12
- "SGI ProPack for Linux on SGI Altix Overview", page 13
- "Solaris Overview", page 14
- "Windows Overview", page 15

AIX Overview

Note: For additional details, see the CXFS MultiOS Clients release notes for AIX. If you run into problems, see Chapter 11, "Troubleshooting", page 169.

Following is the order of installation and configuration steps for a CXFS AIX node:

1. Install the AIX 5L version 5.1 operating system according to the directions in the AIX documentation (if not already done).
2. Install and verify the SGI RAID. See Chapter 2, "SGI RAID Firmware", page 17.
3. Install and verify the Brocade Fibre Channel switch. See Chapter 3, "Brocade Fibre Channel Switch Verification", page 19.
4. Obtain and install the CXFS license. If you want to access an XVM mirrored volume from a given node in the cluster, you must purchase the mirroring software option and obtain and install a FLEXlm license. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your sales representative. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.
5. Install and verify the host bus adapter (HBA). See "Host Bus Adapter Installation and Configuration for AIX", page 36.
6. Prepare the AIX node, including adding a private network. See "Preinstallation Steps for AIX", page 36.
7. Install the CXFS software. See "Client Software Installation Steps for AIX", page 41.
8. Create the I/O fencing file. See "Postinstallation Steps for AIX: Creating the I/O Fencing File", page 43.

9. Configure the cluster to define the new AIX node in the pool, add it to the cluster, start CXFS services, and mount filesystems. See Chapter 10, "Cluster Configuration", page 159.

Linux on IA32 Platforms Overview

Note: For additional details, see the CXFS MultiOS Clients release notes for Linux. If you run into problems, see Chapter 11, "Troubleshooting", page 169.

Following is the order of installation and configuration steps for a CXFS Linux node on supported IA32 platforms:

1. Install the Red Hat 7.3 operating system according to the directions in the Red Hat documentation (if not already done).
2. Install and verify the SGI RAID. See Chapter 2, "SGI RAID Firmware", page 17.
3. Install and verify the Brocade Fibre Channel switch. See Chapter 3, "Brocade Fibre Channel Switch Verification", page 19.
4. Obtain and install the CXFS license. If you want to access an XVM mirrored volume from a given node in the cluster, you must purchase a mirroring software option and obtain and install a FLEXlm license. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your sales representative. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.
5. Install and verify the host bus adapter (HBA). See "Host Bus Adapter Installation and Configuration for Linux on IA32 Platforms", page 51.
6. Prepare the Linux node, including adding a private network. See "Adding a Private Network for Linux Nodes", page 52.
7. Install the CXFS software. See "Client Software Installation Steps for Linux on IA32 Platforms", page 57.
8. Only if necessary, create the I/O fencing file. See "Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File", page 59.
9. Configure the cluster to define the new Linux node in the pool, add it to the cluster, start CXFS services, and mount filesystems. See Chapter 10, "Cluster Configuration", page 159.

SGI ProPack for Linux on SGI Altix Overview

Note: For additional details, see the CXFS MultiOS Clients release notes for SGI ProPack. If you run into problems, see Chapter 11, "Troubleshooting", page 169.

Following is the order of installation and configuration steps for a CXFS SGI ProPack node:

1. Install SGI ProPack 2.2 for Linux release, according to the directions in the SGI ProPack documentation (if not already done).
2. Install and verify the SGI RAID. See Chapter 2, "SGI RAID Firmware", page 17.
3. Install and verify the Brocade Fibre Channel switch. See Chapter 3, "Brocade Fibre Channel Switch Verification", page 19.
4. Obtain and install the CXFS license. If you want to access an XVM mirrored volume from a given node in the cluster, you must purchase a mirroring software option and obtain and install a FLEXlm license. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your sales representative. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.
5. Install and verify the host bus adapter (HBA). See "Host Bus Adapter Installation and Configuration for SGI ProPack for Linux", page 66.
6. Prepare the SGI ProPack node, including adding a private network. See "Adding a Private Network for SGI ProPack for Linux Nodes", page 68.
7. Install the CXFS software. See "Client Software Installation Steps for SGI ProPack for Linux", page 72.
8. Only if necessary, create the I/O fencing file. See "Postinstallation Steps for SGI ProPack for Linux", page 73.
9. Configure the cluster to define the new SGI ProPack node in the pool, add it to the cluster, start CXFS services, and mount filesystems. See Chapter 10, "Cluster Configuration", page 159.

Solaris Overview

Note: For additional details, see the CXFS MultiOS Clients release notes for Solaris. If you run into problems, see Chapter 11, "Troubleshooting", page 169.

Following is the order of installation and configuration steps for a CXFS Solaris node:

1. Install the Solaris 8 or Solaris 9 operating system according to the directions in the Solaris documentation (if not already done).
2. Install and verify the SGI RAID. See Chapter 2, "SGI RAID Firmware", page 17.
3. Install and verify the Brocade Fibre Channel switch. See Chapter 3, "Brocade Fibre Channel Switch Verification", page 19.
4. Obtain and install the CXFS license. If you want to access an XVM mirrored volume from a given node in the cluster, you must purchase a mirroring software option and obtain and install a FLEXlm license. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your sales representative. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.
5. Install and verify the JNI host bus adapter (HBA). You will install the JNIC146x package, which provides software for the Fibre Channel card and driver. See "Host Bus Adapter Installation and Configuration for Solaris", page 83.
6. Prepare the Solaris node, including adding a private network. See "Preinstallation Steps for Solaris", page 96.
7. Install the CXFS software. See "Client Software Installation Steps for Solaris", page 102.
8. Only if necessary, create the I/O fencing file. See "Postinstallation Steps for Solaris: Creating the I/O Fencing File", page 104.
9. Configure the cluster to define the new Solaris node in the pool, add it to the cluster, start CXFS services, and mount filesystems. See Chapter 10, "Cluster Configuration", page 159.

Windows Overview

This information applies to both Windows NT and Windows 2000 nodes unless otherwise noted.

Note: For additional details, see the CXFS MultiOS release notes for Windows. If you run into problems, see Chapter 11, "Troubleshooting", page 169.

Following is the order of installation and configuration steps for a CXFS Windows node:

1. Install the Windows operating system according to the directions in the Windows documentation (if not already done).
2. Install Windows NT Service Pack 6 or Windows 2000 Service Pack 2 or later according to the directions in the Windows documentation (if not already done).
3. Install and verify the SGI RAID. See Chapter 2, "SGI RAID Firmware", page 17.
4. Install and verify the Brocade Fibre Channel switch. See Chapter 3, "Brocade Fibre Channel Switch Verification", page 19.
5. Obtain the CXFS license. If you want to access an XVM mirrored volume from a given node in the cluster, you must purchase a mirroring software option and obtain and install a FLEXlm license. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your sales representative. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.
6. Install and verify the QLogic host bus adapter (HBA) and driver. See "Host Bus Adapter Installation for Windows", page 129.
7. Prepare the Windows node, including adding a private network. See "Preinstallation Steps for Windows", page 133.
8. Install the CXFS software. See "Client Software Installation Steps for Windows", page 138.
9. Perform post-installation configuration steps:
 - "Configuring the FLEXlm License for Windows", page 146
 - "Creating the Windows I/O Fencing File", page 147 (only if necessary)

- "Performing User Configuration", page 149
 - "Creating a New Hardware Profile", page 151
10. Configure the cluster to define the new Solaris node in the pool, add it to the cluster, start CXFS services, and mount filesystems. See Chapter 10, "Cluster Configuration", page 159.
 11. Start CXFS services on the Windows node to see the mounted filesystems under the configured drive letter.

SGI RAID Firmware

The SGI RAID will be initially installed and configured by SGI personnel.

Required SGI RAID Firmware

This section describes the required RAID firmware.

Required SGI TP9500 RAID Firmware

The TP9400/TP9500 6.0 CD contains the required controller firmware and NVSRAM files for the 5884 unit. The 05.30.xx.xx code or later must be installed.

Note: By default, the TP9500 supports 32 logical units (LUNs). If additional LUNs are required, you must obtain a separate software-enabling key; this key will support a maximum of 2048 LUNs in separate partitions, which requires that the Fibre Channel ports be mapped to a partition. Contact your SGI sales representative for the SGI software partitioning key.

Required SGI TP9400 RAID Firmware

The TP9400 4.0 CD contains the required controller firmware and NVSRAM files for the 4774 or 4884 units:

- If you have a 4774 unit, the 04.01.xx.xx code or later must be installed according to FCO 1056.
- If you have a 4884 unit, the 04.02.xx.xx code or later is installed by default.

Note: By default, the TP9400 supports 32 LUNs. If additional LUNs are required, you must obtain a separate software-enabling key; this key will support a maximum number of LUNs in separate partitions, which requires that the Fibre Channel ports be mapped to a partition. The maximum depends upon the code installed:

- 04.01.xx.xx code supports a maximum of 128 LUNs
- 04.02.xx.xx code supports a maximum of 128 LUNs
- 05.30.xx.xx code supports a maximum of 1024 LUNs

Contact your SGI sales representative for the SGI software partitioning key.

Required SGI TP9100 RAID Firmware

The TP9100 4.0 CD contains the required version 7.75 controller firmware and NVSRAM files for the 1-Gbit TP9100. The TP9100 5.0 CD contains the required version 8.29 firmware and NVSRAM files for the 2-Gbit TP9100.

RAID Firmware Verification

To verify that the SGI RAID is properly installed and ready for use with CXFS, you can dump the RAID's profile and verify the controller software revisions.

For More Information

The following documentation is used to install and verify the RAID:

- SGI TP9400 and SGI TP9500:
 - *SGI TP9400 and SGI TP9500 Software Concepts Guide*
 - *SGI TP9400 and SGI TP9500 RAID Owner's Guide*
 - *SGI TP9400 and SGI TP9500 RAID Administration Guide*
- SGI TP9100:
 - *TPM Installation Instructions and User's Guide for TP9100*

Brocade Fibre Channel Switch Verification

In order to protect data integrity, AIX, Linux, Solaris, and Windows nodes must use the *I/O fencing* feature, which isolates a problem node so that it cannot access I/O devices and therefore cannot corrupt data in the shared CXFS filesystem. I/O fencing is also available for IRIX and SGI ProPack nodes on the SGI Altix platform. This feature can only be used with a Brocade Fibre Channel switch supported by SGI; therefore, the Brocade switch is a required piece of hardware in a multiOS cluster.

The Brocade Fibre Channel switches will be initially installed and configured by SGI personnel. You can use the information in this chapter to verify the installation.

Required Brocade Fibre Channel Switch Firmware and License

This release supports Brocade Silksworm Fibre Channel switches that are supported by SGI:

- 2400 (8-port)
- 2800 (16-port)
- 3200 (8-port, 2-Gbit)
- 3800 (16-port, 2-Gbit)
- 3900 (32-port, 2-Gbit)
- 12000 (32-, 64-, or dual 64-port, 2-Gbit)

All Brocade switches contained within the SAN fabric must have the appropriate Brocade license key installed. The following firmware is required:

- 2400 and 2800 switches: 2.6.0d or later
- 3200 and 3800 switches: 3.0.2c or later
- 3900 and 12000 switches: v4.0.2c or later

If the current firmware level of the switches must be upgraded, please contact your local SGI service representative or customer support center.

The Brocade switch must be configured so that its Ethernet interface is accessible from all IRIX nodes using `telnet`. The fencing network connected to the Brocade switch must be physically separate from the private heartbeat network.



Caution: The `telnet` port must be kept free in order for I/O fencing to succeed.

The 3900 and 12000 series switches permit multiple `telnet` sessions. However, CXFS I/O fencing requires a `telnet` lockout for global mutual exclusion when a fencing race occurs. Therefore, you must configure the 3900 and 12000 series switches to set the maximum allowed simultaneous `telnet` sessions for the `admin` user to one. Only the 3900 and 12000 switches require this configuration (other Brocade switch models are shipped with the required restrictions configured by default). See "Configuring the Brocade Silkworm 3900", page 21, and "Configuring the Brocade Silkworm 12000", page 21.

Verifying the Brocade License

To verify the Brocade license, log into the switch as user `admin` and use the `licenseshow` command, as shown in the following example:

```
brocade:admin> licenseshow
dcRyzyScSedSz0p:
  Web license
  Zoning license
  SES license
  Fabric license
SQQQSyddQ9TRRdUP:
  Release v2.2 license
```

Verifying the Brocade Switch Firmware Version

To verify the firmware version, log into the switch as user `admin` and use the `version` command, as shown in the following example:

```
brocade:admin> version
Kernel:      5.3.1
Fabric OS:   v2.2.1c                <== Firmware Revision
Made on:     Mon Dec 18 11:39:26 PST 2000
```

```
Flash:      Mon Dec 18 11:40:01 PST 2000
BootProm:   Thu Jun 17 15:20:39 PDT 1999
```

Configuring the Brocade Silkworm 3900

To limit the maximum allowed simultaneous telnet sessions for the admin user to one on the Brocade Silkworm 3900, do the following:

1. Connect to the switch via the telnet command and login as root.
2. Issue the sync command to avoid filesystem corruption:

```
# sync
```

3. Edit the /etc/profile file to change the max_telnet_sessions from 2 to 1 and place the information in a new file. For example:

```
# cd /etc
# sed -e 's/max_telnet_sessions=2/max_telnet_sessions=1/' profile >profile.new
```

4. Distribute the edited profile file to both partitions on both central processors. For example:

```
# cp profile.new profile
# cp profile.new /mnt/etc/profile
```

5. Issue the sync command again to avoid filesystem corruption:

```
# sync
```

Configuring the Brocade Silkworm 12000

To limit the maximum allowed simultaneous telnet sessions for the admin user to one on the Brocade Silkworm 12000, do the following:

1. Connect to the switch via the telnet command and login as root.
2. Use the haShow command to make sure that both central processors are up. This is indicated by the message Heartbeat Up within the output of the haShow command. If it is not up, wait a couple of minutes and run haShow again to check for the status.

3. Issue the `sync` command on the filesystems to avoid filesystem corruption:

```
# rsh 10.0.0.5 sync
# rsh 10.0.0.6 sync
```

4. Edit the `/etc/profile` file to change the `max_telnet_sessions` from 2 to 1 and place the information in a new file. For example:

```
# cd /etc
# sed -e 's/max_telnet_sessions=2/max_telnet_sessions=1/' profile >profile.new
```

5. Distribute the new profile to both partitions and central processors. For example:

```
# rcp /etc/profile.new 10.0.0.5:/etc/profile
# rcp /etc/profile.new 10.0.0.5:/mnt/etc/profile
# rcp /etc/profile.new 10.0.0.6:/etc/profile
# rcp /etc/profile.new 10.0.0.6:/mnt/etc/profile
```

6. Issue the `sync` command again to avoid filesystem corruption:

```
# rsh 10.0.0.5 sync
# rsh 10.0.0.6 sync
```

Changing the Brocade FC Cable Connections

To change Brocade Fibre Channel cable connections used by nodes in the CXFS cluster, do the following:

1. Cleanly shut down CXFS services on the nodes affected by the cable change using either the CXFS GUI or the `cmgr` command.
2. Rearrange the cables as required.
3. Restart CXFS services.
4. Reconfigure I/O fencing if required. You must perform this step if I/O fencing is enabled on the cluster and if you added/removed any Brocade switches. You must use the CXFS GUI or the `cmgr(1M)` command to add/remove switches from the CXFS configuration as required.
5. If any CXFS client nodes are connected to a new (or different) Brocade switch, restart CXFS services on those nodes. This will ensure that the IRIX servers can correctly identify the Brocade ports used by all clients.
6. If connected to a different RAID device, restart the Windows NT node.

Consult the *CXFS Version 2 Software Installation and Administration Guide* for instructions to configure I/O fencing.

Obtaining CXFS and XVM FLEXlm Licenses

The software licensing used by CXFS is based on the FLEXlm product from Macrovision Corporation. For all supported platforms, a FLEXlm license is required to use CXFS. Perform the procedures in this chapter to satisfy this requirement.

XVM provides a mirroring feature. If you want to access a mirrored volume from a given node in the cluster, you must install a FLEXlm mirroring license on that node. Only those nodes that will access the mirrored volume must be licensed. For information about purchasing this license, see your SGI sales representative.

Obtaining the Host Information Required for the License

When you order CXFS, you will receive an entitlement ID. You must submit the system host ID, host name, and entitlement ID when requesting your permanent CXFS license. The method used to obtain this information is platform-specific.

AIX Host Information

To obtain the host identifier (ID) and hostname of the system on which you will run CXFS, execute the following AIX commands:

```
/usr/bin/uname -m  
/usr/bin/hostname
```

The `uname -m` command lists the host ID as a 12-digit number. For CXFS license purposes, you will drop the first two digits and the last two digits and supply the remaining eight digits.

For example:

```
aix# /usr/bin/uname -m  
000276513100  
aix# /usr/bin/hostname  
cxfsaix3
```

When you are asked for the license manager host identifier, you would provide the following information:

- Host ID: 02765131

That is, the output from the `uname -m` command minus the first two digits (00) and the last two digits (00)

- Hostname: `cxfsaix3`

You must have a separate license for each host on which CXFS is installed.

Linux on IA32 Platforms Host Information

To obtain the host identifier and hostname of the system on which you will run CXFS, execute the following Linux commands:

```
/sbin/ifconfig eth0  
/bin/hostname
```

The host identifier is the value for the `HWaddr` field, minus the colons.

For example:

```
[root@linux root]# /bin/hostname  
cxfslinux  
[root@linux root]# /sbin/ifconfig eth0  
eth0      Link encap:Ethernet  HWaddr 00:E0:81:24:77:D1  
          inet addr:128.162.240.135  Bcast:128.162.240.255  Mask:255.255.255.0  
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
          RX packets:17941247 errors:2 dropped:0 overruns:0 frame:2  
          TX packets:16373834 errors:10 dropped:0 overruns:0 carrier:10  
          collisions:0 txqueuelen:100  
          RX bytes:1539518033 (1468.1 Mb)  TX bytes:1573522873 (1500.6 Mb)  
          Interrupt:19 Base address:0xb880 Memory:fe3fe000-fe3fe038
```

In this case, the host identifier is `00E0812477D1`

When you are asked for the license manager host identifier, provide this information. You must have a separate license for each host on which CXFS is installed.

SGI ProPack Host Information

For SGI Altix systems, you must obtain the host identifier, hostname, and serial number of the system on which you will run CXFS. Execute the following commands:

```
cat /proc/sgi_sn/system_serial_number
/sbin/ifconfig eth0
/bin/hostname
```

The system serial number allows SGI to generate multiple host ID keys for Altix partitioning.

The host identifier is the value for the HWaddr field, minus the colons.

For example:

```
[root@linux root]# cat /proc/sgi_sn/system_serial_number
N0000002
[root@linux root]# /bin/hostname
cxfslinux
[root@linux root]# /sbin/ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:E0:81:24:77:D1
          inet addr:128.162.240.135  Bcast:128.162.240.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:17941247 errors:2 dropped:0 overruns:0 frame:2
          TX packets:16373834 errors:10 dropped:0 overruns:0 carrier:10
          collisions:0 txqueuelen:100
          RX bytes:1539518033 (1468.1 Mb)  TX bytes:1573522873 (1500.6 Mb)
          Interrupt:19 Base address:0xb880 Memory:fe3fe000-fe3fe038
```

In this case, the host identifier is 00E0812477D1

When you are asked for the license manager host identifier, provide this information. You must have a separate license for each host on which CXFS is installed.

Solaris Host Information

To obtain the host identifier and hostname of the system on which you will run CXFS, execute the following Solaris commands:

```
/usr/bin/hostid
/usr/bin/hostname
```

For example:

```
solaris# /usr/bin/hostid
830dad77
solaris# /usr/bin/hostname
cxfssun2
```

When you are asked for the license manager host identifier, provide this information. You must have a separate license for each host on which CXFS is installed.

Windows Host Information

FLEXlm requires that you supply the Ethernet (MAC) address in order to generate the FLEXlm license. This address is known as the *Physical Address* in Windows. You can obtain this information in one of the following ways:

- View the network adapter properties in the **Windows Control Panel**
- Open a command prompt window and run the following command:

```
C:\> ipconfig /all
```

If the machine has more than one network interface, you should use the Physical Address of the private network interface.

Note: Windows NT licenses cannot be used under Windows 2000 and vice versa. If you are upgrading a Windows node to Windows 2000, you must obtain a new license.

Obtaining and Installing the Licenses

Along with your entitlement number, you will receive a URL to a key generation page. To obtain your permanent CXFS and XVM licenses, follow the instructions on the key generation page. After the required information is provided, a key will be generated and displayed on the webpage along with installation instructions.

See also:

- "FLEXlm License Verification for AIX", page 35
- "FLEXlm License Verification for Linux on IA32 Platforms", page 50

- "FLEXlm License Verification for Solaris", page 82
- "Configuring the FLEXlm License for Windows", page 146

For More Information

For more information about licensing, see the following webpage:

<http://www.sgi.com/support/licensing>

If you cannot use the web key generation page, you can contact the SGI order desk at 800 800 4SGI (800 800 4744).

For more information on FLEXlm, you may order the *Flexible License Manager End User Manual* from Macrovision Corporation.

AIX Platform

CXFS supports a client-only node running the AIX operating system.

This chapter contains the following sections:

- "CXFS on AIX"
- "FLEXlm License Verification for AIX", page 35
- "Host Bus Adapter Installation and Configuration for AIX", page 36
- "Preinstallation Steps for AIX", page 36
- "Client Software Installation Steps for AIX", page 41
- "Postinstallation Steps for AIX: Creating the I/O Fencing File", page 43
- "Manual CXFS Start/Shutdown for AIX", page 45
- "Software Maintenance for AIX", page 45

CXFS on AIX

This section contains the following information about CXFS on AIX:

- "Requirements Specific to AIX"
- "CXFS Commands Installed on AIX", page 32
- "Log Files on AIX", page 33
- "Limitations on AIX", page 33

Requirements Specific to AIX

In addition to the items listed in "Requirements", page 6, using an AIX node to support CXFS requires the following:

- An AIX 5L version 5.1 operating system (64-bit kernel mode)
- IBM FC6228 2-Gbit Fibre Channel host bus adapters (HBAs)

- One or more of the following IBM hardware platforms:
 - pSeries 610
 - pSeries 620
 - pSeries 630
 - pSeries 640
 - pSeries 650
 - pSeries 660
 - pSeries 670
 - pSeries 680
 - pSeries 690

Note: This release was tested with the following:

- pSeries 610 Model 6E1
 - pSeries 630 Model 6E4
-

IRIX nodes do not permit nested mount points on CXFS filesystems; that is, you cannot mount an IRIX XFS or CXFS filesystem on top of an existing CXFS filesystem. Although it is possible to mount a JFS or NFS filesystem on top of an AIX CXFS filesystem, this is not recommended.

CXFS Commands Installed on AIX

The following commands are shipped as part of the CXFS for AIX package:

- `/usr/cxfs_cluster/bin/cxfs_client`
- `/usr/cxfs_cluster/bin/cxfs_info`
- `/usr/cxfs_cluster/bin/cxfslicense`
- `/usr/cxfs_cluster/bin/xvmprobe`
- `/usr/cxfs_cluster/bin/xvm`

These commands provide all of the services needed to include an AIX node in a CXFS cluster. The `ls1pp` output lists all of the software added; see "AIX Installation Overview", page 41.

For more information, see the `cxfs_client`, `cxfs_info`, `xvm`, and `xvmprobe` man pages.

Log Files on AIX

The `cxfs_client` command creates a `/var/tmp/cxfs_client` log file. (There is no `/var/cluster` log on AIX nodes.) This log file is not rotated or truncated.

Some daemons working in the kernel related to CXFS output a message in the console log, which is rotated. To see the contents of this log file, use the following command:

```
alog -o -t console
```

For information about the log files created on IRIX nodes, see the *CXFS Version 2 Software Installation and Administration Guide*.

Limitations on AIX

CXFS on AIX has the following limitations:

- CXFS AIX nodes cannot view or edit user and group quotas because CXFS administration must be performed from an IRIX node. However, user and group quotas are enforced correctly by the meta-data server.

To view or edit your quota information, you must log in to an IRIX cluster node and make any necessary changes. If you want to provide a viewing command, such as `repquota`, you can construct an AIX shell script similar to the following:

```
# ! /bin/sh
#
# Where repquota lives on IRIX
repquota=/usr/etc/repquota

# The name of an IRIX node in the cluster
irixnode=cain

rsh $irixnode "$repquota $*"
exit
```

- The block size supported is 4 KB, which is the XFS default block size.
- Unlike IRIX, there is no default access control list (ACL) in AIX. Therefore, the setup and display of the default ACL cannot be completed using the following commands:

```
aclget  
aclput  
acledit
```

If an IRIX ACL exists, the ACL becomes effective when the default ACL is set up by IRIX and a file and a directory are made under that directory in AIX.

- There is no MASK entry in AIX, but the access permissions in AIX follow those established when an ACL set up by IRIX contains a MASK entry. If the default ACL is set up for a given directory and the MASK entry exists, then that MASK entry is used when a file or a subdirectory is made by AIX. When the MASK entry does not exist, `rxw` is used.
- The ACL control of the following, which the AIX JFS filesystem has, cannot be applied to CXFS:
 - The access to a certain user or the group is rejected (`deny`)
 - When a user belongs to the specific group, access is permitted or rejected (`specify`)

If `deny` or `specify` is used, an error occurs (`EINVAL`) because these features are not in IRIX.

- Socket files cannot be copied. The following error is returned:

```
AIX:The socket does not allow the requested operation.
```
- You can use the `fuser` command to extract process information about the mounted filesystem, but you cannot extract process information about the file or the directory.

Maximum CXFS Filesystem and File Size on AIX

The maximum size of a CXFS filesystem on AIX is 2^{64} . The maximum size of a file is 1 terabyte (TB). An attempt to write beyond this limit will result in an `Invalid argument` or a `File too large` error.

Access Control Lists and AIX

All CXFS files have UNIX mode bits (read, write, and execute) and optionally an ACL. For more information, see the AIX `chmod`, `aclaudit`, `aclget`, and `aclput` man pages.

If you want to use an AIX node to restore a CXFS file with an ACL, you should use the `backup` and `restore` commands. If you use the `tar`, `cpio`, or `pax` command, the ACL will not be used because these tools behave "intelligently" by not calling `acl` subroutines to set an ACL. These tools will only set the file mode.

When using the `ls` command to display access permissions for a file with an ACL, the mode reported for a CXFS file follows IRIX semantics instead of AIX JFS semantics.

The IRIX model calls for reporting the ACL MASK for the group permission in the mode. Therefore, if the GROUP entry is `r-x` and the MASK entry is `rw-`, the group permission will be reported as `rw-`. Although it appears that the group has write permission, it does not and an attempt to write to the file will be rejected. You can obtain the real (that is, effective) group permission by using the AIX `aclget` command.

Note: Normally, AIX filesystem ACLs can have up to one memory page (4096 bytes) for a file and a directory. However, CXFS filesystems on AIX nodes in a multiOS cluster must maintain compatibility with the IRIX meta-data server. The CXFS filesystems on an AIX node are limited to a maximum of 25 ACL entries converted to IRIX ACL type for a file and a directory.

FLEXlm License Verification for AIX

Use the `cxfslicense` command with the `-d` option to verify that the FLEXlm licenses have been installed properly.

If the CXFS license is properly installed, you will see the following:

```
# /usr/cxfs_cluster/bin/cxfslicense -d
XVM_AIX license granted.
CXFS_AIX license granted.
```

If you do not have the CXFS license properly installed, you will see the following error on the console when trying to run CXFS:

```
Starting CXFS services> ....
CXFS not properly licensed for this host.  Run
  '/usr/cxfs_cluster/bin/cxfslicense -d'
for detailed failure information.  After fixing the
license, please run '/usr/cxfs_cluster/bin/cxfs_cluster restart'.
```

An error such as the following example will appear in the *SYSLOG* file (line breaks added here for readability):

```
Jan 25 10:24:03 ncc1701:Jan 25 10:24:03 cxfs_client:
cis_main FATAL: cxfs_client failed the CXFS license check.
Use the cxfslicense command to diagnose the license problem
```

Host Bus Adapter Installation and Configuration for AIX

For more information about installing and configuring the host bus adapter (HBA), see the IBM HBA documentation.

You are required to have a second network interface that must be used for the private meta-data network. If you do not already have a second interface installed, you must install a network interface card. You may wish to install that card at this time.

Preinstallation Steps for AIX

When you install the CXFS software on the client-only node, you must modify certain system files. The network configuration is critical. Each node in the cluster must be able to communicate with every node in the cluster by both logical name and IP address without going through any other network routing; proper name resolution is key. SGI recommends static routing.

This section provides an overview of the steps that you or a qualified IBM service representative will perform on your AIX nodes prior to installing the CXFS software. It contains the following sections:

- "Hostname Resolution and Network Configuration Rules for AIX"
- "Adding a Private Network for AIX Nodes", page 37

- "Verifying the Private and Public Network for AIX", page 40

Hostname Resolution and Network Configuration Rules for AIX



Caution: It is critical that you understand these rules before attempting to configure a CXFS cluster.

The following hostname resolution rules and recommendations apply to client-only nodes:

- Hostnames cannot begin with an underscore (_) or include any whitespace characters.
- The private network IP addresses on a running node in the cluster cannot be changed while cluster services are active.
- You must be able to communicate directly between every node in the cluster (including client-only nodes) using IP addresses and logical names, without routing.
- A private network must be dedicated to be the heartbeat and control network. No other load is supported on this network.
- The heartbeat and control network must be connected to all nodes, and all nodes must be configured to use the same subnetwork for that network.

Adding a Private Network for AIX Nodes

The following procedure provides an overview of the steps required to add a private network to the AIX system.

Note: A private network is required for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site. For details about any of these steps, see the AIX documentation.

1. If your system is already operational and on the network, skip to step 2. If the AIX operating system has not been installed, install it in accordance with the *AIX 5L V5.1 Installation Guide*.

2. Edit the `/etc/hosts` file so that it contains entries for every node in the cluster and their private interfaces.

The `/etc/hosts` file has the following format, where *primary_hostname* can be the simple hostname or the fully qualified domain name:

```
IP_address      primary_hostname      aliases
```

You should be consistent when using fully qualified domain names in the `/etc/hosts` file. If you use fully qualified domain names on a particular node, then all of the nodes in the cluster should use the fully qualified name of that node when defining the IP/hostname information for that node in the `/etc/hosts` file.

The decision to use fully qualified domain names is usually a matter of how the clients (such as NFS) are going to resolve names for their client server programs, how their default resolution is done, and so on.

Even if you are using the domain name service (DNS) or the network information service (NIS), you must add every IP address and hostname for the nodes to `/etc/hosts` on all nodes.

For example:

```
190.0.2.1 server1-company.com server1
190.0.2.3 stocks
190.0.3.1 priv-server1
190.0.2.2 server2-company.com server2
190.0.2.4 bonds
190.0.3.2 priv-server2
```

You should then add all of these IP addresses to `/etc/hosts` on the other nodes in the cluster.

Note: Exclusive use of NIS or DNS for IP address lookup for the nodes will reduce availability in situations where the NIS or DNS service becomes unreliable.

For more information, see "Hostname Resolution and Network Configuration Rules for AIX", page 37, and the `hosts`, `named`, and `nis` man pages.

3. (Optional) Edit the `/etc/netsvc.conf` file so that local files are accessed before either NIS or DNS. That is, the `hosts` line in `/etc/netsvc.conf` must list `local` first. For example:

```
hosts = local,nis,bind
```

(The order of `nis` and `bind` is not significant to CXFS, but `local` must be first.)

4. Determine the name of the private interface by using the `ifconfig` command as follows, to list the available networks. For example:

```
# ifconfig -l
en0 en1 lo0
```

However, if the second network interface (`en1`) does not appear, then the network interface must be set up in accordance with the AIX documentation.

You can set up an IP address by using `ifconfig` after restarting the system. If it is set up properly, the following information is output (line breaks added here for readability):

```
# ifconfig -a
en0: flags=4e080863<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GRUPT,64BIT,PSEG>
  inet 10.208.148.61 netmask 0xffffffff broadcast 10.208.148.255
en1: flags=7e080863,10<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GRUPT,64BIT,
  CHECKSUM_OFFLOAD,CHECKSUM_SUPPORT,RSEG>
  inet 192.168.10.61 netmask 0xffffffff broadcast 192.168.10.255
lo0: flags=e08084b<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GRUPT,64BIT>
  inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
```

5. (Optional) Edit the `/.rhosts` file if you want to use remote access or if you want to use the connectivity diagnostics with CXFS. Make sure that the `/.rhosts` file on each AIX node allows all of the nodes in the cluster to have access to each other. The connectivity tests execute a `ping` command from the local node to all nodes and from all nodes to the local node. To execute `ping` on a remote node, CXFS uses `rsh` as user `root`.

For example, suppose you have a cluster with three nodes: `irix0`, `aix1`, and `aix2`. The `/.rhosts` files could be as follows (where the prompt denotes the node name):

```
irix0# cat /.rhosts
aix1 root
aix2 root
```

```
aix1# cat /.rhosts
irix0 root
aix2 root
```

```
aix2# cat /.rhosts
irix0 root
aix1 root
```

Verifying the Private and Public Network for AIX

For each private network on each AIX node in the pool, verify access with the AIX ping command. Enter the following, where *nodeIPAddress* is the IP address of the node:

```
/usr/sbin/ping -c 3 nodeIPAddress
```

For example:

```
aix# /usr/sbin/ping -c 3 192.168.10.61
PING 192.168.10.61: (192.168.10.61): 56 data data bytes
64 bytes from 192.168.10.61 icmp_seq=0 ttl=255 time=0 ms
64 bytes from 192.168.10.61 icmp_seq=1 ttl=255 time=0 ms
64 bytes from 192.168.10.61 icmp_seq=2 ttl=255 time=0 ms
----192.168.10.61 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/00 ms
```

You should also execute a ping on the public networks. If that ping fails, follow these steps:

1. Verify that the network interface was configured up using `ifconfig`. For example:

```
aix# /usr/sbin/ifconfig en0
en0: flgs=4e08086<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT,PSEG>
inet 10.208.148.61 netmask 0xffffffff0 broadcast 10.208.148.255
```

In the first output line above, `UP` indicates that the interface was configured up.

2. Verify that the cables are correctly seated. Repeat this procedure on each node.

Client Software Installation Steps for AIX

The CXFS software initially will be installed and configured by SGI personnel. This section provides the following:

- "AIX Installation Overview"
- "Verifying the AIX Installation ", page 43

AIX Installation Overview

Installing the CXFS client CD for AIX requires approximately 20 MB of space. To install the required software on an AIX node, SGI personnel will do the following:

1. Verify that the node has been upgraded to AIX 5.1L according to the *AIX 5L V5.1 Installation Guide*. Use the following command to display the currently installed system:

```
aix# uname -rv
1 5
```

This output indicates 1 is the revision and 5 is the version.

2. Insert the CXFS multiOS client CD-ROM.
3. Mount the CD-ROM:

```
aix# mount -v cdrfs -o ro /dev/cd0 /mnt/cdrom
```

4. Install the CXFS software (the example output below is truncated):

```
aix# installp -a -d /mnt/cdrom/SGIcxfs-aix5L all
+-----+
+-----+
Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
Results...

SUCSESSES
-----
Filesets listed in this section passed pre-installation verification
and will be installed.
```

Selected Filesets

SGIcxf5-aix5L 2.5.0.0 # CXFS CLIENT for AIX

<< End of Success Section >>

FILESET STATISTICS

1 Selected to be installed, of which:
1 Passed pre-installation verification

1 Total to be installed

+-----+
Installing Software...
+-----+

installp: APPLYING software for:
SGIcxf5-aix5L 2.5.0.0

. << Copyright notice for SGIcxf5-aix5L >>

.

Finished processing all filesets. (Total time: 2 secs).

+-----+
Summaries:
+-----+

Installation Summary

Name	Level	Part	Event	Result
SGIcxf5-aix5L	2.5.0.0	USR	APPLY	SUCCESS

5. Verify that the CXFS license key has been installed. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25. For example:

```
# /usr/cxfs_cluster/bin/cxfslicense -d
CXFS license granted.
```

Verifying the AIX Installation

To verify that the CXFS software has been installed properly, use the `lslpp` command as follows:

```
aix# lslpp -L SGIcxfs-aix5L
```

For example, the following output (showing a state of C, for “committed”) indicates that the CXFS package installed properly:

```
aix# lslpp -L SGIcxfs-aix5L
Fileset                                Level  State  Type  Description (Uninstaller)
-----
SGIcxfs-aix5L                          2.5.0.0  C      F      CXFS CLIENT for AIX
```

State codes:

```
A -- Applied.
B -- Broken.
C -- Committed.
O -- Obsolete. (partially migrated to newer version)
? -- Inconsistent State...Run lppchk -v.
```

Type codes:

```
F -- Installp Fileset
P -- Product
C -- Component
T -- Feature
R -- RPM Package
```

Postinstallation Steps for AIX: Creating the I/O Fencing File

I/O fencing is required to protect data integrity for AIX nodes. To use I/O fencing, you must create the AIX `/etc/fencing.conf` file, which lists the worldwide port names (WWPNs) of any supported host bus adapters (HBAs) in the system that are

connected to a switch that is configured in the cluster database. These HBAs will then be available for fencing.

If you want to use the `/etc/fencing.conf` file, you must update it whenever the HBA configuration changes, including the replacement of an HBA.

The `/etc/fencing.conf` file enumerates the WWPN for all of the host bus adapters (HBA) that will be used to mount a CXFS filesystem. The `/etc/fencing.conf` file must contain a simple list of WWPNs, as a 64-bit hexadecimal number, one per line.

If you are not completely certain which number you should use, do the following:

1. Follow the Fibre Channel cable on the back of the AIX host to determine the port to which it is connected in the Brocade Fibre Channel switch. Ports are numbered beginning with 0. (For example, if there are 8 ports, they will be numbered 0 through 7.)
2. Use the `telnet` command to connect to the Brocade Fibre Channel switch and log in as user `admin` (the password is `password` by default).
3. Execute the `switchshow` command to display the switches and their WWPNs. For example:

```
brocade04:admin> switchshow
switchName:      brocade04
switchType:      2.4
switchState:     Online
switchRole:      Principal
switchDomain:    6
switchId:        fffc06
switchWwn:       10:00:00:60:69:12:11:9e
switchBeacon:    OFF
port  0:  sw  Online      F-Port  20:00:00:01:73:00:2c:0b
port  1:  cu  Online      F-Port  21:00:00:e0:8b:02:36:49
port  2:  cu  Online      F-Port  21:00:00:e0:8b:02:12:49
port  3:  sw  Online      F-Port  20:00:00:01:73:00:2d:3e
port  4:  cu  Online      F-Port  21:00:00:e0:8b:02:18:96
port  5:  cu  Online      F-Port  21:00:00:e0:8b:00:90:8e
port  6:  sw  Online      F-Port  20:00:00:01:73:00:3b:5f
port  7:  sw  Online      F-Port  20:00:00:01:73:00:33:76
port  8:  sw  Online      F-Port  21:00:00:e0:8b:01:d2:57
port  9:  sw  Online      F-Port  21:00:00:e0:8b:01:0c:57
port 10: sw  Online      F-Port  20:08:00:a0:b8:0c:13:c9
```

```
port 11: sw Online      F-Port  20:0a:00:a0:b8:0c:04:5a
port 12: sw Online      F-Port  20:0c:00:a0:b8:0c:24:76
port 13: sw Online      L-Port  1 public
port 14: sw No_Light
port 15: cu Online      F-Port  21:00:00:e0:8b:00:42:d8
```

The WWPN is the hexadecimal string to the right of the port number. For example, the WWPN for port 0 is 2000000173002c0b (you must remove the colons from the WWPN reported in the `switchshow` output to produce the string to be used in the `/etc/fencing.conf` file).

4. Edit or create the `/etc/fencing.conf` file on the AIX node and add the WWPN for the port determined in step 1. (Comment lines begin with a # character.) For example, if you determined that port 0 is the port connected to the Brocade Fibre Channel switch, your `/etc/fencing.conf` file should appear as follows:

```
2000000173002c0b
```

5. After the AIX node is added to the cluster (see Chapter 10, "Cluster Configuration", page 159), enable the fencing feature by using the CXFS GUI or `cmgr` command on an IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Manual CXFS Start/Shutdown for AIX

The `/usr/cxfs_cluster/bin/cxfs_cluster` script will be invoked automatically during normal system startup and shutdown procedures. This script starts and stops the processes required to run CXFS.

To start up CXFS process manually on your AIX node, enter the following:

```
# /usr/cxfs_cluster/bin/cxfs_cluster start
```

To stop CXFS processes manually, enter the following:

```
# /usr/cxfs_cluster/bin/cxfs_cluster stop
```

Software Maintenance for AIX

This section contains information about upgrading and modifying the CXFS software on an AIX system.

Upgrading the CXFS Software on an AIX System

To upgrade the CXFS software on an AIX system, do the following:

1. Make sure that no applications on the node are accessing files on a CXFS filesystem.
2. Determine the name of the CXFS package that is installed. For example:

```
aix# lsllp -L | grep cxfs  
SGIcxfs-aix5L          2.5.0.0    C    F    CXFS CLIENT for AIX
```

3. Uninstall the old version by using the following command:

```
installp -u packagename
```

For example, given a package name of SGIcxfs-aix5L:

```
aix# installp -u SGIcxfs-aix5L
```

4. Install the new version. See "Client Software Installation Steps for AIX", page 41.

Modifying the CXFS Software on an AIX System

You can modify the CXFS client service (`/usr/cxfs_cluster/bin/cxfs_client`) by placing options in the `/usr/cxfs_cluster/bin/cxfs_client.options` file. The available options are documented in the `cxfs_client` man page.



Caution: Some of the options are intended to be used internally by SGI only for testing purposes and do not represent supported configurations. Consult your SGI service representative before making any changes.

Linux on IA32 Platforms

CXFS supports a client-only node running the Linux operating system on supported IA32 platforms.

Note: On Linux systems, the use of XVM is supported only with CXFS; XVM does not support local Linux IA32 disk volumes.

This chapter contains the following sections:

- "CXFS on Linux on IA32 Platforms"
- "FLEXlm License Verification for Linux on IA32 Platforms", page 50
- "Host Bus Adapter Installation and Configuration for Linux on IA32 Platforms", page 51
- "Preinstallation Steps for Linux on IA32 Platforms", page 51
- "Client Software Installation Steps for Linux on IA32 Platforms", page 57
- "Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File", page 59
- "Manual CXFS Startup/Shutdown for Linux on IA32 Platforms", page 60
- "Software Maintenance: Modifying the CXFS Software on a Linux on IA32 Platforms", page 60

CXFS on Linux on IA32 Platforms

This section contains the following information about CXFS on Linux:

- "Requirements Specific to Linux on IA32 Platforms"
- "CXFS Commands Installed on Linux on IA32 Platforms", page 49
- "Log Files on Linux on IA32 Platforms", page 49
- "Limitations and Considerations for Linux on IA32 Platforms", page 49

- "Maximum CXFS Filesystem and File Size on Linux on IA32 Platforms", page 50
- "Access Control Lists and Linux on IA32 Platforms", page 50

Requirements Specific to Linux on IA32 Platforms

In addition to the items listed in "Requirements", page 6, using a Linux node to support CXFS requires the following:

- Red Hat Linux 7.3 operating system
- A choice of at least one Fibre Channel host bus adapter (HBA):
 - QLogic 2200, QLogic 2310, or QLogic 2342 HBA
 - JNI FCX-6562-N 2 Gb 133 MHz PCI-X-to-Fibre Channel HBAs

Note: 1-Gbit HBAs and Sbus HBAs are not supported.

- LSI Logic LS17202XP-LC dual channel PCI-X HBA
- A CPU of the following type:
 - Intel Pentium III
 - Intel Pentium 4
 - Advanced Micro Devices AMD Athlon
 - Advanced Micro Devices AMD Duron

The machine must have at least the following minimum requirements:

- 256 MB of RAM memory
- Two Ethernet 100baseT interfaces
- One empty PCI slot (to receive the HBA)

IRIX nodes do not permit nested mount points on CXFS filesystems; that is, you cannot mount an IRIX XFS or CXFS filesystem on top of an existing CXFS filesystem. Although it is possible to mount other filesystems on top of a Linux CXFS filesystem, this is not recommended.

CXFS Commands Installed on Linux on IA32 Platforms

The following commands are shipped as part of the CXFS Linux package:

- `/usr/cluster/bin/cxfs_client` (the CXFS client service)
- `/usr/cluster/bin/cxfslicense`
- `/sbin/xvm`

These commands provide all of the services needed to include a Linux node in a CXFS cluster. The `rpm` output lists all software added; see "Installation Overview ", page 57.

For more information, see the `cxfs_client` and `xvm` man pages.

Log Files on Linux on IA32 Platforms

The `cxfs_client` command creates a `/var/log/cxfs_client` log file. (There is no `/var/cluster` log on Linux nodes.)

For information about the log files created on IRIX nodes, see the *CXFS Version 2 Software Installation and Administration Guide*.

Limitations and Considerations for Linux on IA32 Platforms

CXFS for Linux has the following limitations and considerations:

- CXFS Linux nodes cannot view or edit user and group quotas because CXFS administration must be performed from an IRIX node. However, user and group quotas are enforced correctly by the meta-data server.
- To view or edit your quota information, you must log in to an IRIX cluster node and make any necessary changes. If you would like to provide a viewing command such as `repquota`, you could construct a Linux shell script similar to the following:

```
#!/bin/sh
#

# Where repquota lives on IRIX
repquota=/usr/etc/repquota
```

```
# The name of an IRIX node in the cluster
irixnode=cain

rsh $irixnode "$repquota $*"
exit
```

- The maximum block size supported is 4 KB, determined by the kernel page size. (XFS uses a default block size of 4 KB unless overridden by an administrator to a different blocksize value, for example 1 KB or 2 KB.)
- Due to Linux Kernel limitations, CXFS filesystems cannot be mounted with the `inode64` mount option.
- CXFS filesystems with XFS version 1 directory format cannot be mounted on Linux nodes.

Maximum CXFS Filesystem and File Size on Linux on IA32 Platforms

The maximum size of a CXFS filesystem on a supported Linux IA32 client is 2 terabytes; this is due to a Linux kernel restriction. The maximum size of a file is 16 terabytes, using Large File Support (`O_LARGEFILE`). An attempt to write beyond this limit will result in an `Invalid argument` or a `File too large` error.

Access Control Lists and Linux on IA32 Platforms

All CXFS files have UNIX mode bits (read, write, and execute) and optionally an access control list (ACL). For more information, see the `chmod` and `setfacl` man pages.

Red Hat Linux 7.3 file utilities do not provide ACL support, so commands such as `ls` and `cp` will not show or preserve ACLs. However, the commands in the `acl` package will allow manipulation of the ACLs of files on CXFS filesystems.

FLEXlm License Verification for Linux on IA32 Platforms

Use the `cxfslicense` command with the `-d` option to verify that the FLEXlm licenses have been installed properly.

If the CXFS license is properly installed, you will see the following:

```
[root@linux root]# /usr/cluster/bin/cxfslicense -d
CXFS license granted.
```

If you do not have the CXFS license properly installed, you will see the following error on the console and in the `cxfs client` logs when trying to run CXFS:

```
May 12 14:40:17 cxfs_client: cis_main FATAL: cxfs_client failed the CXFS
license check. Use the cxfslicense command to diagnose the license
problem.
May 12 14:40:17 cxfs_client: FATAL: aborting on fatal error
```

Host Bus Adapter Installation and Configuration for Linux on IA32 Platforms

This section provides an overview of the Fibre Channel host bus adapter (HBA) installation information for Linux nodes.

The installation may be performed by you or by a qualified service representative for your hardware. See the Red Hat documentation and the documentation for your hardware platform.

The driver requirements are as follows:

- QLogic card: the drivers are supplied with the Red Hat Linux kernel. The module names are `qla2200.o` and `qla2300.o`.
- LSI Logic card: the drivers are supplied with the Red Hat Linux kernel. The module name is `mptscsih.o`.
- JNI card: the drivers are supplied on the JNI website: <http://www.jni.com>

Preinstallation Steps for Linux on IA32 Platforms

When you install the CXFS software on the client-only node, you must modify certain system files. The network configuration is critical. Each node in the cluster must be able to communicate with every other node in the cluster by both logical name and IP address without going through any other network routing; proper name resolution is key. SGI recommends static routing.

This section provides an overview of the steps that you will perform on your Linux nodes prior to installing the CXFS software. It contains the following sections:

- "Hostname Resolution and Network Configuration Rules for Linux"
- "Adding a Private Network for Linux Nodes", page 52
- "Verifying the Private and Public Networks for Linux", page 56

Hostname Resolution and Network Configuration Rules for Linux



Caution: It is critical that you understand these rules before attempting to configure a CXFS cluster.

The following hostname resolution rules and recommendations apply to client-only nodes:

- Hostnames cannot begin with an underscore (`_`) or include any whitespace characters.
- The private network IP addresses on a running node in the cluster cannot be changed while cluster services are active.
- You must be able to communicate directly between every node in the cluster (including client-only nodes) using IP addresses and logical names, without routing.
- A private network must be dedicated to be the heartbeat and control network. No other load is supported on this network.
- The heartbeat and control network must be connected to all nodes, and all nodes must be configured to use the same subnet for that network.

If you change hostname resolution settings in the `/etc/nsswitch.conf` file after you have defined the first IRIX node (which creates the cluster database), you must recreate the cluster database.

Adding a Private Network for Linux Nodes

The following procedure provides an overview of the steps required to add a private network to the Linux system.

Note: A private network is required for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site. For details about any of these steps, see the Red Hat Linux documentation.

1. Edit the `/etc/hosts` file so that it contains entries for every node in the cluster and their private interfaces as well.

The `/etc/hosts` file has the following format, where *primary_hostname* can be the simple hostname or the fully qualified domain name:

```
IP_address    primary_hostname    aliases
```

You should be consistent when using fully qualified domain names in the `/etc/hosts` file. If you use fully qualified domain names on a particular node, then all of the nodes in the cluster should use the fully qualified name of that node when defining the IP/hostname information for that node in their `/etc/hosts` file.

The decision to use fully qualified domain names is usually a matter of how the clients (such as NFS) are going to resolve names for their client server programs, how their default resolution is done, and so on.

Even if you are using the domain name service (DNS) or the network information service (NIS), you must add every IP address and hostname for the nodes to `/etc/hosts` on all nodes. For example:

```
190.0.2.1 server1-company.com server1
190.0.2.3 stocks
190.0.3.1 priv-server1
190.0.2.2 server2-company.com server2
190.0.2.4 bonds
190.0.3.2 priv-server2
```

You should then add all of these IP addresses to `/etc/hosts` on the other nodes in the cluster.

For more information, see the `hosts` and `resolver` man pages.

Note: Exclusive use of NIS or DNS for IP address lookup for the nodes will reduce availability in situations where the NIS or DNS service becomes unreliable.

For more information, see "Hostname Resolution and Network Configuration Rules for Linux", page 52.

2. Edit the `/etc/nsswitch.conf` file so that local files are accessed before either NIS or DNS. That is, the `hosts` line in `/etc/nsswitch.conf` must list `files` first.

For example:

```
hosts:      files nis dns
```

(The order of `nis` and `dns` is not significant to CIFS, but `files` must be first.)

For example:

3. Configure your private interface according to the instructions in the Network Configuration section of your Linux distribution manual. To verify that the private interface is operational, issue the following command:

```
[root@linux root]# ifconfig -a

eth0      Link encap:Ethernet  HWaddr 00:50:81:A4:75:6A
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:13782788  errors:0  dropped:0  overruns:0  frame:0
          TX packets:60846  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0  txqueuelen:100
          RX bytes:826016878 (787.7 Mb)  TX bytes:5745933 (5.4 Mb)
          Interrupt:19  Base address:0xb880  Memory:fe0fe000-fe0fe038

eth1      Link encap:Ethernet  HWaddr 00:81:8A:10:5C:34
          inet addr:10.0.0.10  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0  errors:0  dropped:0  overruns:0  frame:0
          TX packets:0  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0  txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:19  Base address:0xef00  Memory:febfd000-febfd038
```

```
lo      Link encap:Local Loopback
        inet addr:127.0.0.1  Mask:255.0.0.0
        UP LOOPBACK RUNNING  MTU:16436  Metric:1
        RX packets:162 errors:0 dropped:0 overruns:0 frame:0
        TX packets:162 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:11692 (11.4 Kb)  TX bytes:11692 (11.4 Kb)
```

This example shows that two ethernet interfaces, `eth0` and `eth1`, are present and running (as indicated by `UP` in the third line of each interface description).

If the second network does not appear, it may be that a network interface card must be installed in order to provide a second network, or it may be that the network is not yet initialized.

4. (Optional) Edit the `/root/.rhosts` file if you want to use the connectivity diagnostics provided with CXFS.

Ensure that the `/root/.rhosts` file on each Linux node allows all of the nodes in the cluster to have access to each other. The connectivity tests execute a `ping` command from the local node to all nodes and from all nodes to the local node. To execute `ping` on a remote node, CXFS uses `rsh` as user `root`.

For example, suppose you have a cluster with three nodes: `irix0`, `linux1`, and `linux2`. The `.rhosts` files could be as follows (the prompt denotes the node name):

```
irix0# cat /.rhosts
linux1 root
linux1-priv root
linux2 root
linux2-priv root

linux1# cat /root/.rhosts
irix0 root
irix0-priv root
linux2 root
linux2-priv root
```

```
linux2# cat /root/.rhosts
irix0 root
irix0-priv root
linux1 root
linux1-priv root
```

Verifying the Private and Public Networks for Linux

For each private network on each Linux node in the pool, verify access with the ping command. Enter the following, where *nodeIPAddress* is the IP address of the node:

```
ping nodeIPAddress
```

For example:

```
[root@linux root]# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) from 128.162.240.141 : 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.310 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.122 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.127 ms
```

Also execute a ping on the public networks. If ping fails, follow these steps:

1. Verify that the network interface was configured up using `ifconfig`. For example:

```
[root@linux root]# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:81:8A:10:5C:34
          inet addr:10.0.0.10  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:19 Base address:0xef00 Memory:febfd000-febfd038
```

In the third output line above, UP indicates that the interface was configured up.

2. Verify that the cables are correctly seated.

Repeat this procedure on each node.

Client Software Installation Steps for Linux on IA32 Platforms

The CXFS software will be initially installed and configured by SGI personnel. This section provides an overview of those procedures. You can use the information in this section to verify the installation.

Installation Overview

Installing the CXFS client CD for Linux requires approximately 50–200 MB of space, depending upon the packages installed at your site.

To install the required software on a Linux node, SGI personnel will do the following:

1. Verify that the node is running a supported Linux distribution, according to the CXFS for Linux release notes. For example, use the following command to display the currently installed Red Hat system:

```
[root@linux root]# cat /etc/redhat-release
Red Hat Linux release 7.3 (Valhalla)
```

2. Insert and mount the XFS 1.3 CD-ROM.
3. Install the appropriate kernel RPM for the system, according to the information about upgrading the kernel in the operating system documentation.

For example, to install the kernel on an Intel Pentium III dual-processor machine, enter the following (the RPM version numbers may differ from the released product):

```
[root@linux cdrom]# rpm -ivh kernel-smp-2.4.20-13.7.sgi4.i686.rpm
Preparing...      ##### [100%]
 1:kernel-smp      ##### [100%]
```

4. Install the `xfs-modules` RPM:

```
[root@linux cdrom]# rpm -ivh xfs-modules-smp-1.3pre1-2.4.20_13.7.sgi4_sgi3.i686.rpm
Preparing...      ##### [100%]
 1:xfs-modules     ##### [100%]
```

5. Configure the new kernel into the bootloader according to the instructions in the operating system documentation.

6. Install the XFS filesystem packages (line break added here for readability):

```
[root@linux cdrom]# rpm -Uvh acl-2.2.7-0.i386.rpm attr-2.4.1-0.i386.rpm dmapi-2.0.6-0.i386.rpm \
xfsdump-2.2.8-0.i386.rpm xfsprogs-2.4.5-0.i386.rpm
Preparing...                               ##### [100%]
 1:acl                                     ##### [ 20%]
 2:attr                                   ##### [ 40%]
 3:dmapi                                  ##### [ 60%]
 4:xfsdump                                ##### [ 80%]
 5:xfsprogs                               ##### [100%]
```

7. Insert and mount the CXFS MultiOS CD-ROM.

8. Install the CXFS kernel modules that correspond to the kernel installed in step 3, and the user-space packages. The following is an example of installing CXFS kernel modules:

```
[root@linux cdrom]# rpm -Uvh cxfs-modules-smp-2.5-2.4.20_13.7.sgi4_sgi16.i686.rpm
Preparing...                               ##### [100%]
 1:cxfs-modules                            ##### [100%]
```

The following is an example of installing user-space packages (line break added for readability):

```
[root@linux cdrom]# rpm -Uvh cxfs_client-2.5-RH7.3_sgi6.i386.rpm \
xvm-cluster-cmds-2.5-RH7.3_sgi8.i386.rpm
Preparing...                               ##### [100%]
 1:cxfs_client                             ##### [ 50%]
 2:xvm-cluster-cmds                         ##### [100%]
```

9. Edit the `/etc/cluster/config/cxfs_client.options` file as necessary.
10. Verify that the CXFS license key has been installed in `/var/flexlm`. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.

For example:

```
[root@linux root]# /usr/cluster/bin/cxfslicense -d
CXFS license granted.
```

11. Reboot the system with the newly installed kernel:

```
[root@linux root]# reboot
```

Verifying the Linux Installation

Use the `uname -r` command to ensure the kernel installed in step 8 above is running. For example:

```
[root@linux root]# uname -r  
2.4.18-27.7.x.sgimp
```

To verify that the CXFS software has been installed properly, use the `rpm` command to query the packages.

For example, the following output indicates that the CXFS packages are installed properly:

```
[root@linux root]# rpm -q cxfs-modules cxfs_client xvm-cluster-cmds  
cxfs-modules-2.5-sgi  
cxfs_client-2.5-sgi  
xvm-cluster-cmds-2.5-sgi
```

Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File

I/O fencing is required to protect data integrity for Linux nodes.

On Linux, the `cxfs_client` software detects the world wide port names (WWPNs) of any supported host bus adapters (HBAs) in the system that are connected to a switch that is configured in the cluster database. These HBAs will then be available for fencing.

If automatic HBA WWPN detection fails according to messages in the `/var/log/cxfs_client` log file, then the `/etc/fencing.conf` file can be used to specify the WWPN(s). If used, this file must contain a simple list of WWPNs, one per line. To determine the WWPNs, see the documentation for the driver in use.

If you want to use the `/etc/fencing.conf` file, you must update it whenever the HBA configuration changes, including the replacement of an HBA.

Manual CXFS Startup/Shutdown for Linux on IA32 Platforms

The `/etc/init.d/cxfs_cluster` script will be invoked automatically during normal system startup and shutdown procedures. This script starts and stops the processes required to run CXFS.

To start up CXFS processes manually on your Linux node, enter the following:

```
[root@linux root]# /etc/init.d/cxfs_cluster start
Loading cxfs modules:                [ OK ]
Mounting devfs filesystems:         [ OK ]
Starting cxfs client:                [ OK ]
```

To stop CXFS processes manually, enter the following:

```
[root@linux root]# /etc/init.d/cxfs_cluster stop
Stopping cxfs client:                [ OK ]
```

To see the current status of the CXFS processes, use the `status` argument. For example, the following output shows that the service is running:

```
[root@linux root]# /etc/init.d/cxfs_cluster status
cxfs_client (pid 3226) is running...
```

For example, if the service is stopped:

```
[root@linux root]# /etc/init.d/cxfs_cluster status
cxfs_client is stopped
```

Software Maintenance: Modifying the CXFS Software on a Linux on IA32 Platforms

You can modify the CXFS client service (`/usr/cluster/bin/cxfs_client`) by placing options in the `/usr/cluster/bin/cxfs_client.options` file. The available options are documented in the `cxfs_client` man page.



Caution: Some of the options are intended to be used internally by SGI only for testing purposes and do not represent supported configurations. Consult your SGI service representative before making any changes.

The first line in the `cxfs_client.options` file must contain the options you want `cxfs_client` to process; you cannot include a comment as the first line.

To see if `cxfs_client` is using the options in `cxfs_client.options`, enter the following:

```
[root@linux root]# ps -ax | grep cxfs_client
3612 ?          S          0:00 /usr/cluster/bin/cxfs_client -i cxfs3-5
3841 pts/0      S          0:00 grep cxfs_client
```


SGI ProPack for Linux Platform

CXFS supports a client-only node running on SGI ProPack for Linux on SGI Altix servers. SGI ProPack for Linux is an overlay product that adds or enhances features in the supported Linux base distributions.

This chapter discusses the following:

- "CXFS on SGI ProPack for Linux"
- "FLEXlm License Verification for SGI ProPack for Linux ", page 66
- "Host Bus Adapter Installation and Configuration for SGI ProPack for Linux", page 66
- "Preinstallation Steps for SGI ProPack for Linux", page 67
- "Client Software Installation Steps for SGI ProPack for Linux", page 72
- "Postinstallation Steps for SGI ProPack for Linux", page 73
- "Manual CXFS Startup/Shutdown on SGI ProPack for Linux", page 74
- "Software Maintenance: Modifying the CXFS Software on an SGI ProPack for Linux System ", page 74

CXFS on SGI ProPack for Linux

This section contains the following information about CXFS on SGI ProPack:

- "Requirements Specific to SGI ProPack for Linux", page 64
- "CXFS Commands Installed on SGI ProPack for Linux", page 64
- "Log Files on SGI ProPack for Linux", page 64
- "Limitations and Considerations for SGI ProPack for Linux ", page 65
- "Maximum CXFS Filesystem and File Size on SGI ProPack for Linux", page 65
- "Access Control Lists and SGI ProPack for Linux", page 66

Requirements Specific to SGI ProPack for Linux

In addition to the items listed in "Requirements", page 6, using a SGI ProPack node to support CXFS requires the following:

- SGI ProPack 2.2 for Linux
- SGI Altix server
- At least one QLogic 2310 or QLogic 2342 Fibre Channel host bus adapter (HBA)

IRIX nodes do not permit nested mount points on CXFS filesystems; that is, you cannot mount an IRIX XFS or CXFS filesystem on top of an existing CXFS filesystem. Although it is possible to mount other filesystems on top of a SGI ProPack CXFS filesystem, this is not recommended.

CXFS Commands Installed on SGI ProPack for Linux

The following commands are shipped as part of the CXFS SGI ProPack package:

- `/usr/cluster/bin/cxfs_client` (the CXFS client service)
- `/usr/cluster/bin/cxfslicense`
- `/sbin/xvm`

These commands provide all of the services needed to include a SGI ProPack node in a CXFS cluster. The `rpm` output lists all software added; see "Installation Overview", page 57.

For more information, see the `cxfs_client` and `xvm` man pages.

Log Files on SGI ProPack for Linux

The `cxfs_client` command creates a `/var/log/cxfs_client` log file. (There is no `/var/cluster` log on SGI ProPack nodes.)

For information about the log files created on IRIX nodes, see the *CXFS Version 2 Software Installation and Administration Guide*.

Limitations and Considerations for SGI ProPack for Linux

CXFS for SGI ProPack has the following limitations and considerations:

- CXFS SGI ProPack nodes cannot view or edit user and group quotas because CXFS administration must be performed from an IRIX node. However, user and group quotas are enforced correctly by the meta-data server.
- To view or edit your quota information, you must log in to an IRIX cluster node and make any necessary changes. If you would like to provide a viewing command such as `repquota`, you could construct a SGI ProPack shell script similar to the following:

```
#!/bin/sh
#

# Where repquota lives on IRIX
repquota=/usr/etc/repquota

# The name of an IRIX node in the cluster
irixnode=cain

rsh $irixnode "$repquota $*"
exit
```

- The maximum block size supported is 16 KB, determined by the kernel page size. (XFS uses a default block size of 4 KB unless overridden by an administrator to a different blocksize value, for example 8 KB or 16 KB.)
- Due to SGI ProPack Kernel limitations, CXFS filesystems cannot be mounted with the `inode64` mount option.
- CXFS filesystems with XFS version 1 directory format cannot be mounted on SGI ProPack nodes.

Maximum CXFS Filesystem and File Size on SGI ProPack for Linux

The maximum size of an XFS filesystem under SGI ProPack for Linux is 2^{64} bytes.
The maximum size of an XFS file under SGI ProPack for Linux is 2^{63-1} bytes.

Access Control Lists and SGI ProPack for Linux

All CXFS files have UNIX mode bits (read, write, and execute) and optionally an access control list (ACL). For more information, see the `chmod` and `setfacl` man pages.

SGI ProPack 2.2 file utilities do not provide ACL support, so commands such as `ls` and `cp` will not show or preserve ACLs. However, the commands in the `acl` package will allow manipulation of the ACLs of files on CXFS filesystems.

FLEXlm License Verification for SGI ProPack for Linux

Use the `cxfslicense` command with the `-d` option to verify that the FLEXlm licenses have been installed properly.

If the CXFS license is properly installed, you will see the following:

```
[root@linux root]# /usr/cluster/bin/cxfslicense -d
CXFS license granted.
```

If you do not have the CXFS license properly installed, you will see the following error on the console and in the `cxfs_client` logs when trying to run CXFS:

```
May 12 14:40:17 cxfs_client: cis_main FATAL: cxfs_client failed the CXFS
license check. Use the cxfslicense command to diagnose the license
problem.
```

```
May 12 14:40:17 cxfs_client: FATAL: aborting on fatal error
```

Host Bus Adapter Installation and Configuration for SGI ProPack for Linux

The Fibre Channel host bus adapter (HBA) installation may be performed by you or by a qualified service representative for your hardware. See the SGI ProPack and SGI Altix documentation.

The QLogic drivers are supplied with the SGI ProPack `xscsi-driver` package. The module name is `qlfc-mod`.

Preinstallation Steps for SGI ProPack for Linux

When you install the CXFS software on the client-only node, you must modify certain system files. The network configuration is critical. Each node in the cluster must be able to communicate with every other node in the cluster by both logical name and IP address without going through any other network routing; proper name resolution is key. SGI recommends static routing.

This section provides an overview of the steps that you will perform on your SGI ProPack nodes prior to installing the CXFS software. It contains the following sections:

- "Hostname Resolution and Network Configuration Rules for SGI ProPack for Linux", page 67
- "Adding a Private Network for SGI ProPack for Linux Nodes", page 68
- "Verifying the Private and Public Networks for SGI ProPack for Linux", page 71

Hostname Resolution and Network Configuration Rules for SGI ProPack for Linux



Caution: It is critical that you understand these rules before attempting to configure a CXFS cluster.

The following hostname resolution rules and recommendations apply to client-only nodes:

- Hostnames cannot begin with an underscore (_) or include any whitespace characters.
- The private network IP addresses on a running node in the cluster cannot be changed while cluster services are active.
- You must be able to communicate directly between every node in the cluster (including client-only nodes) using IP addresses and logical names, without routing.
- A private network must be dedicated to be the heartbeat and control network. No other load is supported on this network.
- The heartbeat and control network must be connected to all nodes, and all nodes must be configured to use the same subnet for that network.

If you change hostname resolution settings in the `/etc/nsswitch.conf` file after you have defined the first IRIX node (which creates the cluster database), you must recreate the cluster database.

Adding a Private Network for SGI ProPack for Linux Nodes

The following procedure provides an overview of the steps required to add a private network to the SGI ProPack system.

Note: A private network is required for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site. For details about any of these steps, see the SGI ProPack documentation.

1. Edit the `/etc/hosts` file so that it contains entries for every node in the cluster and their private interfaces as well.

The `/etc/hosts` file has the following format, where *primary_hostname* can be the simple hostname or the fully qualified domain name:

```
IP_address    primary_hostname    aliases
```

You should be consistent when using fully qualified domain names in the `/etc/hosts` file. If you use fully qualified domain names on a particular node, then all of the nodes in the cluster should use the fully qualified name of that node when defining the IP/hostname information for that node in their `/etc/hosts` file.

The decision to use fully qualified domain names is usually a matter of how the clients (such as NFS) are going to resolve names for their client server programs, how their default resolution is done, and so on.

Even if you are using the domain name service (DNS) or the network information service (NIS), you must add every IP address and hostname for the nodes to `/etc/hosts` on all nodes. For example:

```
190.0.2.1 server1-company.com server1
190.0.2.3 stocks
190.0.3.1 priv-server1
190.0.2.2 server2-company.com server2
```

```
190.0.2.4 bonds
190.0.3.2 priv-server2
```

You should then add all of these IP addresses to `/etc/hosts` on the other nodes in the cluster.

For more information, see the `hosts` and `resolver` man pages.

Note: Exclusive use of NIS or DNS for IP address lookup for the nodes will reduce availability in situations where the NIS or DNS service becomes unreliable.

For more information, see "Hostname Resolution and Network Configuration Rules for Linux", page 52.

2. Edit the `/etc/nsswitch.conf` file so that local files are accessed before either NIS or DNS. That is, the `hosts` line in `/etc/nsswitch.conf` must list `files` first.

For example:

```
hosts:      files nis dns
```

(The order of `nis` and `dns` is not significant to CXFS, but `files` must be first.)

For example:

3. Configure your private interface according to the instructions in the Network Configuration section of your SGI ProPack distribution manual. To verify that the private interface is operational, issue the following command:

```
[root@linux root]# ifconfig -a

eth0      Link encap:Ethernet  HWaddr 00:50:81:A4:75:6A
          inet addr:192.168.1.1  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:13782788  errors:0  dropped:0  overruns:0  frame:0
          TX packets:60846  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0  txqueuelen:100
          RX bytes:826016878 (787.7 Mb)  TX bytes:5745933 (5.4 Mb)
          Interrupt:19  Base address:0xb880  Memory:fe0fe000-fe0fe038

eth1      Link encap:Ethernet  HWaddr 00:81:8A:10:5C:34
          inet addr:10.0.0.10  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
```

```
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
Interrupt:19 Base address:0xef00 Memory:febfd000-febfd038
```

```
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:162 errors:0 dropped:0 overruns:0 frame:0
TX packets:162 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:11692 (11.4 Kb) TX bytes:11692 (11.4 Kb)
```

This example shows that two ethernet interfaces, `eth0` and `eth1`, are present and running (as indicated by `UP` in the third line of each interface description).

If the second network does not appear, it may be that a network interface card must be installed in order to provide a second network, or it may be that the network is not yet initialized.

4. (*Optional*) Edit the `/root/.rhosts` file if you want to use the connectivity diagnostics provided with CXFS.

Ensure that the `/root/.rhosts` file on each SGI ProPack node allows all of the nodes in the cluster to have access to each other. The connectivity tests execute a `ping` command from the local node to all nodes and from all nodes to the local node. To execute `ping` on a remote node, CXFS uses `rsh` as user `root`.

For example, suppose you have a cluster with three nodes: `irix0`, `linux1`, and `linux2`. The `.rhosts` files could be as follows (the prompt denotes the node name):

```
irix0# cat /.rhosts
linux1 root
linux1-priv root
linux2 root
linux2-priv root

linux1# cat /root/.rhosts
irix0 root
irix0-priv root
linux2 root
```

```
linux2-priv root

linux2# cat /root/.rhosts
irix0 root
irix0-priv root
linux1 root
linux1-priv root
```

Verifying the Private and Public Networks for SGI ProPack for Linux

For each private network on each SGI ProPack node in the pool, verify access with the ping command. Enter the following, where *nodeIPaddress* is the IP address of the node:

```
ping nodeIPaddress
```

For example:

```
[root@linux root]# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) from 128.162.240.141 : 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.310 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.122 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.127 ms
```

Also execute a ping on the public networks. If ping fails, follow these steps:

1. Verify that the network interface was configured up using `ifconfig`. For example:

```
[root@linux root]# ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:81:8A:10:5C:34
          inet addr:10.0.0.10  Bcast:10.0.0.255  Mask:255.255.255.0
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
          Interrupt:19 Base address:0xef00 Memory:febfd000-febfd038
```

In the third output line above, UP indicates that the interface was configured up.

2. Verify that the cables are correctly seated.

Repeat this procedure on each node.

Client Software Installation Steps for SGI ProPack for Linux

The CXFS software will be initially installed and configured by SGI personnel. This section provides an overview of those procedures. You can use the information in this section to verify the installation.

Installation Overview

Installing the CXFS client CD for SGI ProPack requires approximately 50–200 MB of space, depending upon the packages installed at your site.

To install the required software on a SGI ProPack node, SGI personnel will do the following:

1. Verify that the node is running a supported SGI ProPack distribution, according to the SGI ProPack release notes. For example, use the following command to display the currently installed SGI ProPack system:

```
[root@linux root]# cat /etc/sgi-release
SGI ProPack 2.2 for Linux
```

2. Insert and mount the CXFS for SGI ProPack 2.2 CD-ROM.
3. Install the CXFS kernel modules and the user-space packages. The following is an example of installing CXFS kernel modules (the package names may differ on the released system):

```
[root@linux cdrom]# rpm -Uvh cxfs-modules-2.5-sgi220.ia64.rpm
Preparing...                               ##### [100%]
 1:cxfs-modules                             ##### [100%]
```

The following is an example of installing user-space packages (line break here for readability):

```
[root@linux cdrom]# rpm -Uvh cxfs_client-2.5-sgi220.ia64.rpm
xvm-cluster-cmds-2.5-sgi220.ia64.rpm
Preparing...                               ##### [100%]
 1:cxfs_client                             ##### [ 50%]
 2:xvm-cmds-cluster                         ##### [100%]
```

4. Edit the `/etc/cluster/config/cxfs_client.options` file as necessary.

5. Verify that the CXFS license key has been installed in `/var/flexlm`. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.

For example:

```
[root@linux root]# /usr/cluster/bin/cxfslicense -d
CXFS license granted.
```

Verifying the SGI ProPack for Linux Installation

To verify that the CXFS software has been installed properly, use the `rpm` command to query the packages.

For example, the following output indicates that the CXFS packages are installed properly (the version numbers on the released system may differ):

```
[root@linux root]# rpm -q cxfs-modules cxfs_client xvm-cluster-cmds
cxfs-modules-2.5-sgi220
cxfs_client-2.5-sgi220
xvm-cluster-cmds-2.5-sgi220
```

Postinstallation Steps for SGI ProPack for Linux

I/O fencing is required to protect data integrity for SGI ProPack nodes.

On SGI ProPack, the `cxfs_client` software detects the world wide port names (WWPNs) of any supported host bus adapters (HBAs) in the system that are connected to a switch that is configured in the cluster database. These HBAs will then be available for fencing.

If automatic HBA WWPN detection fails according to messages in the `/var/log/cxfs_client` log file, then the `/etc/fencing.conf` file can be used to specify the WWPN(s). If used, this file must contain a simple list of WWPNs, one per line. To determine the WWPNs, see the documentation for the driver in use.

If you want to use the `/etc/fencing.conf` file, you must update it whenever the HBA configuration changes, including the replacement of an HBA.

Manual CXFS Startup/Shutdown on SGI ProPack for Linux

The `/etc/init.d/cxfs_cluster` script will be invoked automatically during normal system startup and shutdown procedures. This script starts and stops the processes required to run CXFS.

To start up CXFS processes manually on your SGI ProPack node, enter the following:

```
[root@linux root]# /etc/init.d/cxfs_cluster start
Loading cxfs modules:                [ OK ]
Mounting devfs filesystems:         [ OK ]
Starting cxfs client:                [ OK ]
```

To stop CXFS processes manually, enter the following:

```
[root@linux root]# /etc/init.d/cxfs_cluster stop
Stopping cxfs client:                [ OK ]
```

To see the current status of the CXFS processes, use the `status` argument. For example, the following output shows that the service is running:

```
[root@linux root]# /etc/init.d/cxfs_cluster status
cxfs_client (pid 3226) is running...
```

For example, if the service is stopped:

```
[root@linux root]# /etc/init.d/cxfs_cluster status
cxfs_client is stopped
```

Software Maintenance: Modifying the CXFS Software on an SGI ProPack for Linux System

You can modify the CXFS client service (`/usr/cluster/bin/cxfs_client`) by placing options in the `/usr/cluster/bin/cxfs_client.options` file. The available options are documented in the `cxfs_client` man page.



Caution: Some of the options are intended to be used internally by SGI only for testing purposes and do not represent supported configurations. Consult your SGI service representative before making any changes.

The first line in the `cxfs_client.options` file must contain the options you want `cxfs_client` to process; you cannot include a comment as the first line.

To see if `cxfs_client` is using the options in `cxfs_client.options`, enter the following:

```
[root@linux root]# ps -ax | grep cxfs_client
3612 ?          S          0:00 /usr/cluster/bin/cxfs_client -i cxfs3-5
3841 pts/0      S          0:00 grep cxfs_client
```


Solaris Platform

CXFS supports a client-only node running the Solaris operating system.

This chapter contains the following sections:

- "CXFS on Solaris"
- "FLEXlm License Verification for Solaris", page 82
- "Host Bus Adapter Installation and Configuration for Solaris", page 83
- "Preinstallation Steps for Solaris", page 96
- "Client Software Installation Steps for Solaris", page 102
- "Postinstallation Steps for Solaris: Creating the I/O Fencing File", page 104
- "Manual CXFS Startup/Shutdown for Solaris", page 106
- "Software Maintenance for Solaris", page 107

CXFS on Solaris

This section contains the following information about CXFS on Solaris:

- "Requirements Specific to Solaris", page 78
- "CXFS Commands Installed on Solaris", page 79
- "Log Files on Solaris", page 79
- "Limitations and Considerations on Solaris", page 79
- "Maximum CXFS Filesystem and File Size on Solaris", page 81
- "Access Control Lists and Solaris", page 81

Requirements Specific to Solaris

In addition to the items listed in "Requirements", page 6, using a Solaris node to support CXFS requires the following:

- Solaris 8 or Solaris 9 operating system.
- One to four JNI FibreStar FCE-6460-N (PCI) 2-Gbit Fibre Channel host bus adapters (HBAs).

Note: 1-Gbit HBAs and Sbus HBAs are not supported.

- One or more of the following Sun Microsystems hardware platform series:
 - Sun Fire 280R
 - Sun Fire V480
 - Sun Fire V880
 - Sun Fire 4800/4810 (PCI slots only, cPCI is not supported)
 - Sun Fire 6800 (PCI slots only, cPCI is not supported)
 - Sun Fire 12K
 - Sun Fire 15K
 - Ultra Enterprise 250
 - Ultra Enterprise 450
 - Ultra Enterprise 4000
 - Ultra Enterprise 3000
 - Ultra Enterprise 5000
 - Ultra Enterprise 6000
 - Ultra Enterprise 10000

IRIX nodes do not permit nested mount points on CXFS filesystems; that is, you cannot mount an IRIX XFS or CXFS filesystem on top of an existing CXFS filesystem. Although it is possible to mount a UFS or NFS filesystem on top of a Solaris CXFS filesystem, this is not recommended.

CXFS Commands Installed on Solaris

The following commands are shipped as part of the CXFS Solaris package:

- `/usr/cxfs_cluster/bin/cxfs_client` (the CXFS client service)
- `/usr/cxfs_cluster/bin/cxfslicense`
- `/usr/cxfs_cluster/bin/xvm`
- `/usr/cxfs_cluster/bin/xvmprobe`

These commands provide all of the services needed to include a Solaris node in a CXFS cluster. The `pkgadd` output lists all software added; see "Solaris Installation Overview", page 103.

For more information, see the `cxfs_client`, `xvm`, and `xvmprobe` man pages.

Log Files on Solaris

The `cxfs_client` command creates a `/var/log/cxfs_client` log file. (There is no `/var/cluster` log on Solaris nodes.) This log file is not rotated or truncated.

For information about the log files created on IRIX nodes, see the *CXFS Version 2 Software Installation and Administration Guide*.

Limitations and Considerations on Solaris

CXFS for Solaris has the following limitations and considerations:

- For optimal performance, you should set the value of the Solaris system tunable parameter `maxphys` in the `/etc/system` file. Do the following:

1. Make a backup copy of the `/etc/system` file.

Note: Exercise extreme caution in changing `/etc/system` and always make a backup copy.

2. Change the value of `maxphys` to `0x800000` (hexadecimal) in the `/etc/system` file.
3. Reboot the Solaris node. This causes the change to take effect.

4. Verify that the new value for `maxphys` is in effect by running the following command:

```
solaris# echo "maxphys/X" | adb -k
physmem 1f03f
maxphys:
maxphys:      800000
```

- CXFS Solaris nodes cannot view or edit user and group quotas because CXFS administration must be performed from an IRIX node. However, user and group quotas are enforced correctly by the meta-data server.

To view or edit your quota information, you must log in to an IRIX cluster node and make any necessary changes. If you would like to provide a viewing command such as `repquota`, you could construct a Solaris shell script similar to the following:

```
#!/bin/sh
#

# Where repquota lives on IRIX
repquota=/usr/etc/repquota

# The name of an IRIX node in the cluster
irixnode=cain

rsh $irixnode "$repquota $*"
exit
```

- The minimum block size supported is 2 KB, determined by a maximum of 4 extents per 8-KB page. (XFS uses a default block size of 4 KB unless overridden by an administrator to a different blocksize value, for example 2 KB or 8 KB.) It is recommended that a filesystem block size of at least 8 KB is used with CXFS Solaris nodes to improve performance.
- All disk devices attached to JNI controllers must be for use only by CXFS disks; do not attach non-disk devices to any JNI controllers that are configured for CXFS use. This restriction is required because all disk devices on JNI controllers configured for CXFS make use of the whole disk volume, which must be conveyed to Solaris via modification in the JNI driver to the value returned by the `READ_CAPACITY SCSI` command.

Maximum CXFS Filesystem and File Size on Solaris

The maximum size of a CXFS filesystem on Solaris is 2^{64} bytes. The maximum size of a file is 2^{63-1} bytes. An attempt to write beyond this limit will result in an `EFBIG` (File too large) error and the process will be sent a `SIGXFSZ` (Filesize limit exceeded) signal.

Access Control Lists and Solaris

All CXFS files have UNIX mode bits (read, write, and execute) and optionally an access control list (ACL). For more information, see the `chmod` and `setfacl` man pages.

If you restore a CXFS file that had an ACL containing only owner-ACL entries (that is, owner/group/other/mask) from a Solaris node, upon restoration one of the following will happen:

- **When using `tar(1)`, `cpio(1)`, and Legato Networker:** The ACL will be lost because these tools behave "intelligently" by not calling `acl` to set an ACL if the file has only owner/group/other/mask entries. These tools will only set the file mode. However, this does not present a change in functionality because an access permissions check on the mode and the ACL containing only owner entries will give the same result.
- **When using other backup/restore utilities:** A mask will be added to the ACL if the application calls `acl` for every file.

A backup/restore utility that calls `acl` to set an ACL for every file will result in a file being restored with four ACL entries (that is, owner/group/other/mask), even though it may have originally had only three (that is, owner/group/other). This is due to a requirement in `getfacl` that it receive four ACL entries for the `GETACL` command to `acl`. (If fewer than four entries are returned, `getfacl` will report an error).

Note: Normally, Solaris filesystem ACLs can have up to 1024 entries for a file and a directory can have 1024 entries as well as an additional 1024 entries for the default ACL. However, CXFS filesystems on Solaris nodes in a multiOS cluster must maintain compatibility with the IRIX meta-data server. The CXFS filesystems on a Solaris node are limited to a maximum of 25 ACL entries for a file and a maximum total of 50 for a directory (that is, the directory ACL plus the default ACL).

When using the `ls` command to display access permissions for a file with an ACL, the mode reported for a CXFS file follows IRIX semantics instead of Solaris/UFS semantics.

On Solaris, a UFS file mode reports the group permission as the intersection of the `GROUP` and `MASK` entries in the ACL. If the `GROUP` entry is `r-x` and the `MASK` entry is `rw-`, the group permission will be reported as `r--`.

The IRIX model calls for reporting the ACL `MASK` for the group permission in the mode. Therefore, using the example above, the group permission will be reported as `rw-`. Although, it appears that the group has write permission, it does not and an attempt to write to the file will be rejected. You can obtain the real (that is, effective) group permission by using the Solaris `getfacl` command.

FLEXlm License Verification for Solaris

Use the `cxfslicense` command with the `-d` option to verify that the FLEXlm licenses have been installed properly.

If the CXFS license is properly installed, you will see the following:

```
solaris# /usr/cxfs_cluster/bin/cxfslicense -d
CXFS license granted.
```

If you do not have the CXFS license properly installed, you will see the following error on the console when trying to run CXFS:

```
Cluster services: CXFS not properly licensed for this host. Run
    '/usr/cxfs_cluster/bin/cxfslicense -d'
for detailed failure information. After fixing the
license, please run '/etc/init.d/cxfs_cluster restart'.
```

An error such as the following example will appear in the `SYSLOG` file:

```
Mar  4 12:58:05 6X:typhoon-q32 crsd[533]: <<CI> N crs 0> Crsd restarted.
Mar  4 12:58:05 6X:typhoon-q32 clconfd[537]: <<CI> N clconf 0>
Mar  4 12:58:05 5B:typhoon-q32 CLCONFD failed the CXFS license check. Use the
Mar  4 12:58:05 5B:typhoon-q32    '/usr/cxfs_cluster/bin/cxfslicense -d'
Mar  4 12:58:05 5B:typhoon-q32 command to diagnose the license problem.
```

Host Bus Adapter Installation and Configuration for Solaris

This section provides an overview of the JNI Fibre Channel host bus adapter (HBA) installation and verification for Solaris nodes:

- "Installing the JNI HBA"
- "Protecting Data Integrity", page 85
- "Installing and Running the EZ Fibre Configuration GUI", page 85
- "Verifying the JNI HBA Installation", page 94

These procedures may be performed by you or by a qualified Sun service representative. You must be logged in as `root` to perform the steps listed in this section.

Installing the JNI HBA

You can use one to four JNI HBAs for CXFS per Sun machine. (Other HBAs may be present that are not shared with the CXFS cluster.)

To install the JNI HBA, perform the following steps. Additional details are provided in various chapters/sections of the *Installation Guide, FCE-6460 and FCE2-6460 PCI-to-Fibre Channel Host Bus Adapters (Solaris, Windows NT/2000, Novell, AIX, HP-UX, Mac OS, Linux) JNI FibreStar*, as noted.

1. Install the JNI host bus adapter (HBA) into the Solaris system. Perform the steps in the following chapter:
 - "Hardware Installation"

Note: The JNI card **must** be installed in a 66MHz slot; if it is installed in another type of slot, CXFS will not work properly.

2. Bring the system back up using the steps listed in the following "Verifying" sections (the following represents the location of these sections in the manual):
 - "Unix Server DriverSuite"
 - "Solaris Driver"
 - "Verifying Hardware in OpenBoot PROM"

- “Verifying Hardware in Solaris”

You will be required to perform a Solaris `boot -r` after installing hardware.



Caution: If you do not see the expected results, do not attempt to continue. Instead, diagnose why the card is not being seen by the hardware.

3. Install the latest JNI HBA driver software (JNIC146x) and Storage Networking Industry Association (SNIA) application programming interface package (JNISnia), according to the instructions in the following “Installing” section:

- “Unix Server DriverSuite”
 - “Solaris Driver”
 - “Installing the Software”

You can retrieve the driver and SNIA package from the following JNI website:

<http://www.jni.com/Drivers>

- a. Under **Locate Driver by Product**, click on **FCE-6460**.
- b. Under the **Solaris** section, left click **JNIC146x.pkg** and save as the following pathname:

```
/var/tmp/JNIC146x.pkg
```

Verify that the driver attached correctly to the HBA and that the package installed correctly by following the verification steps at the end of the section. Do not proceed until the verification succeeds.

4. Set the HBA to fabric mode:

- a. In the `/kernel/drv/jnic146x.conf` file, change the following lines:

```
# FcLoopEnabled=1;  
# FcFabricEnabled=0;
```

Delete the # character at the beginning of each line to uncomment it and change the values so that loop is disabled (0) and fabric is enabled (1). When you are done, the lines will appear as follows:

```
FcLoopEnabled=0;  
FcFabricEnabled=1;
```

- b. Reboot the Solaris node by entering the following command:

```
solaris# init 6
```

Protecting Data Integrity

It is possible to put data integrity at risk if Fibre Channel cables are disconnected or fail. To protect against this issue, you must uncomment the `FailoverDelay` parameter in the `/kernel/drv/jnic146x.conf` file and change it to 0. It then should appear as follows:

```
FailoverDelay = 0;
```

This change ensures data integrity in the CXFS filesystem in the event when the Solaris client machine is unable to see the filesystems.

Installing and Running the EZ Fibre Configuration GUI

After you have verified the installation of the HBA and the driver's attachment to it, you are ready to install and run the JNI EZ Fibre program. This graphical user interface (GUI) will modify the driver's configuration file, `/kernel/drv/jnic146x.conf`, so that it lists the worldwide node name (WWNN) and worldwide port name (WWPN) for the devices on your Fibre Channel.

For general installation information, see *Quick Installation Guide, Solaris, AIX and Windows JNI EZ Fibre*.

Do the following:

1. Install the GUI and change to the appropriate directory:
 - a. Download the latest EZ Fibre GUI from the JNI website (EZ Fibre 2.2.1 or later):
 - i. Go to the following website:

```
http://www.jni.com/Drivers
```
 - ii. Under **Locate Driver by Product**, click on **FCE-6460**
 - iii. Under the **Solaris** section, left click **EZF_221.tar** or later and save as the following pathname (for example):

```
/var/tmp/EZF_221.tar
```

- b. Extract the saved file using the tar command. For example:

```
solaris# tar xvf /var/tmp/EZF_221.tar
```

- c. Change to the directory where the extracted GUI command is located:

```
solaris# cd /var/tmp/EZF_22
```

- d. Run the install.sh script:

```
solaris# ./install.sh
Checking for required and recommended patches...
checkpatches.sh: Note - The following OS vendor recommended patches are
not installed or have been superseded -- please consult the EZ-Fibre
read me:
 108434-01 108435-01
<press enter to continue>
```

```
InstallAnywhere is preparing to install...
Installer using temporary disk space in '/tmp' ($TMPDIR not set).
```

You should install the GUI into the default location. When you see **Congratulations!**, click on **Done**.

- 2. Change to the following directory and read any README files you find there:

```
solaris# cd /opt/jni/ezfibre/standalone
```

- 3. Invoke the EZ Fibre GUI from within the same directory by entering the following command:

```
solaris# ./ezf
```

Two windows will appear. The first, titled **Refresh SAN information**, will say **Discovering LUNs for HBA#0**.

After a short while, this window will be replaced by a larger window, as shown in the example in Figure 8-1. (The example screen snaps in this document may not exactly match the display you see.)

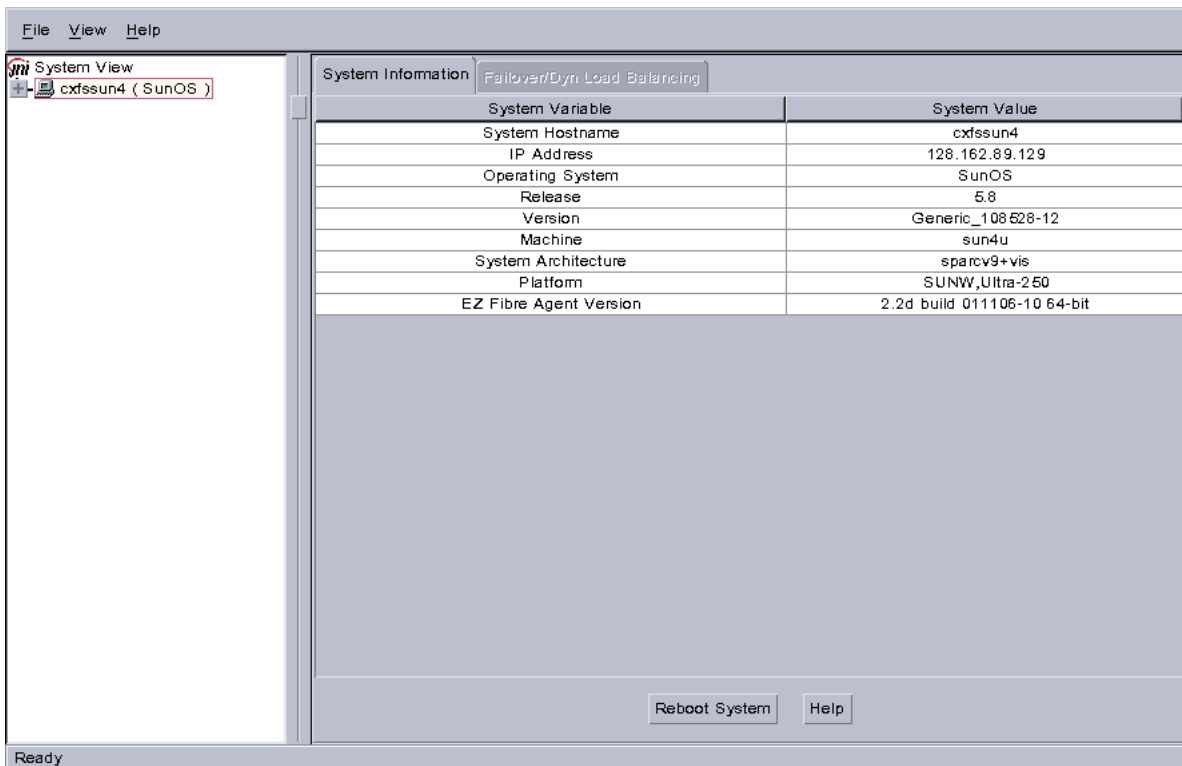


Figure 8-1 Example: Second Window: EZ Fibre Configuration Utility - Standalone

The left-hand pane of this window displays a listing of systems. Find the system you are configuring and click on the + sign next to it; this action expands the display so that it shows the installed JNI HBA on the system. Figure 8-2 highlights the + sign.

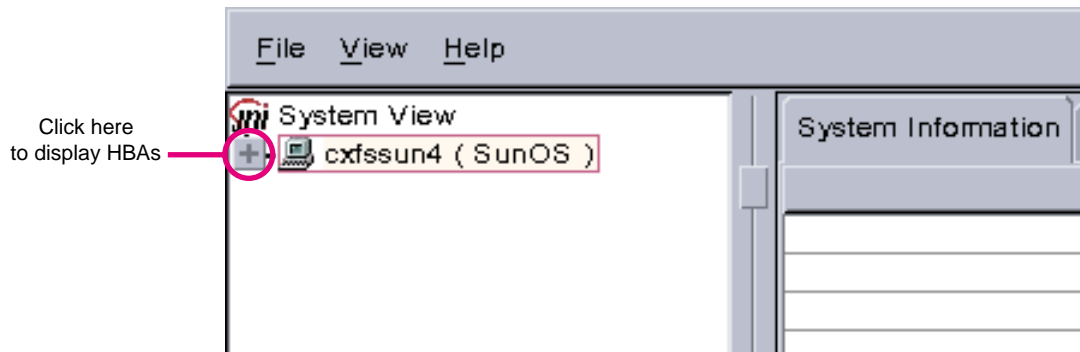


Figure 8-2 Location of icon (+) to Display the HBA

Figure 8-3 shows an example of the display after clicking on the + sign for *cxfsun4*, which shows the JNI HBA.

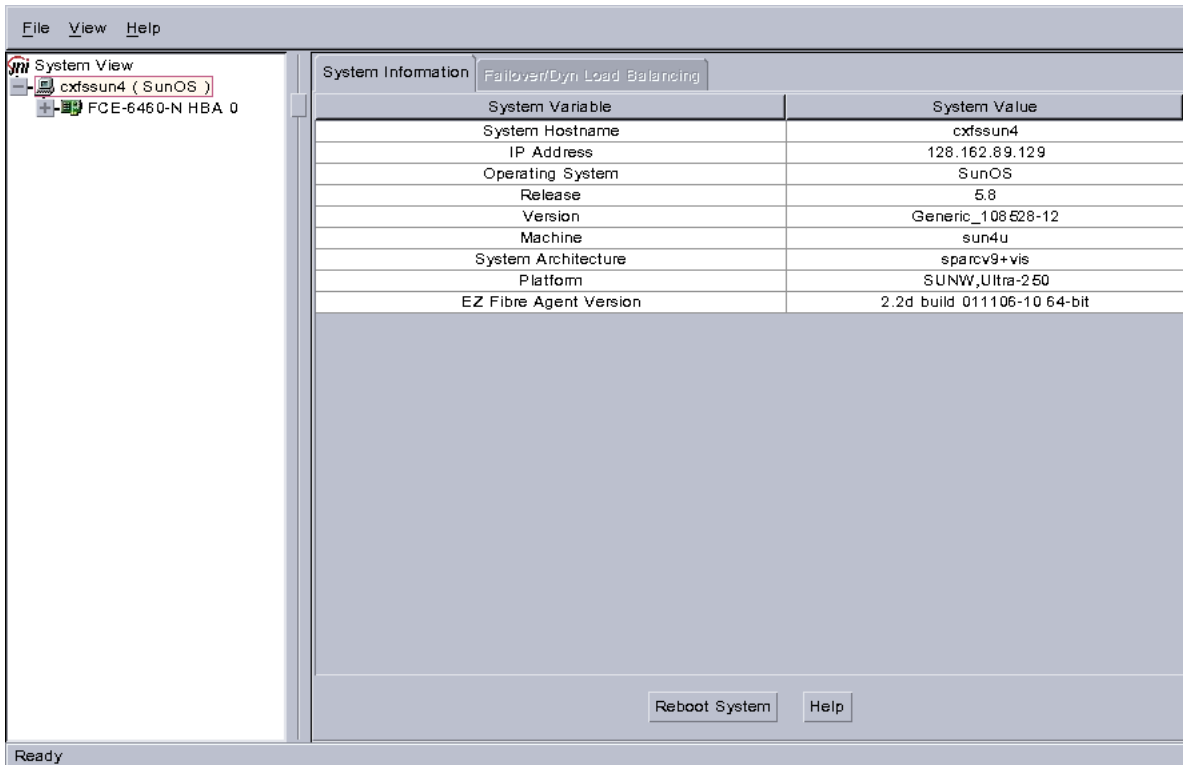


Figure 8-3 Example: After Clicking + to Display the HBA

4. Click on the icon to the right (not the + sign to the left). Figure 8-4 shows the icon.

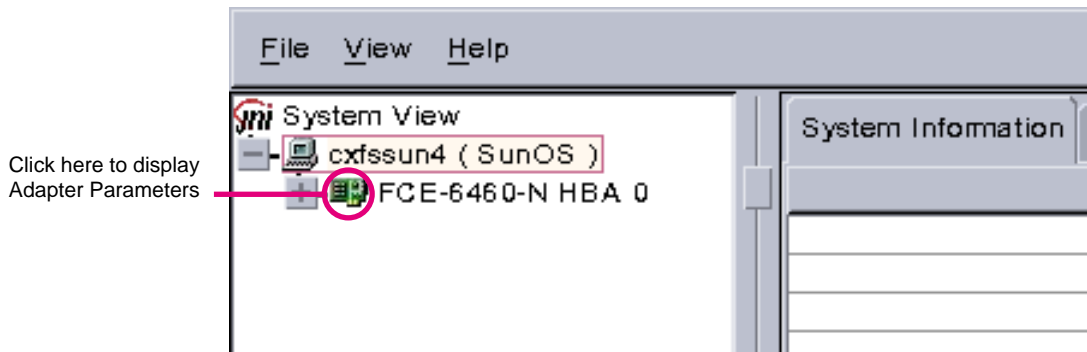


Figure 8-4 Location of the Icon to Display the Adapter Parameters

The right-hand pane will change to show **Adapter Parameters** for the selected HBA, as shown in Figure 8-5.

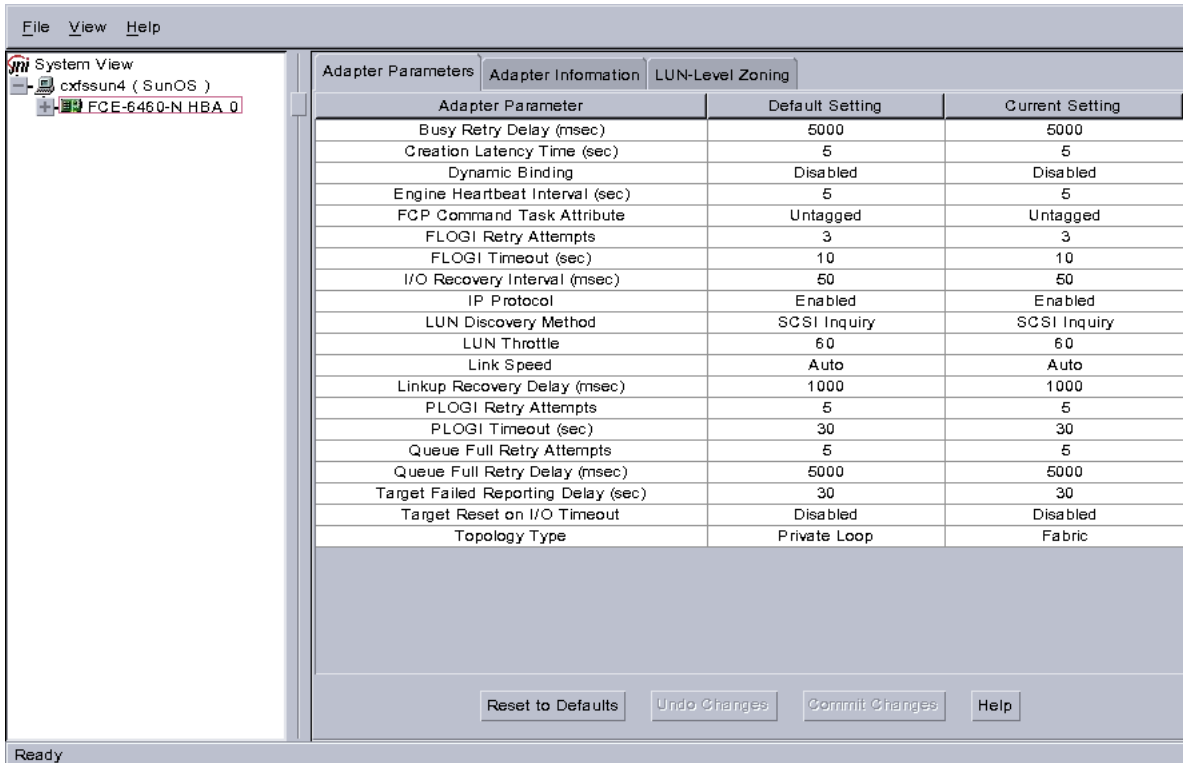


Figure 8-5 Example: After Clicking the HBA Icon to Show the Adapter Parameters

- a. Click on the **Adapter Information** tab to see the information in Figure 8-6.

The last two lines show the WWNN and WWPN of the JNI HBA. You will need the WWPN numbers when you configure the `/etc/fencing.conf` file, so take note of them now; for more information about the `/etc/fencing.conf` file, see "Postinstallation Steps for Solaris: Creating the I/O Fencing File", page 104.

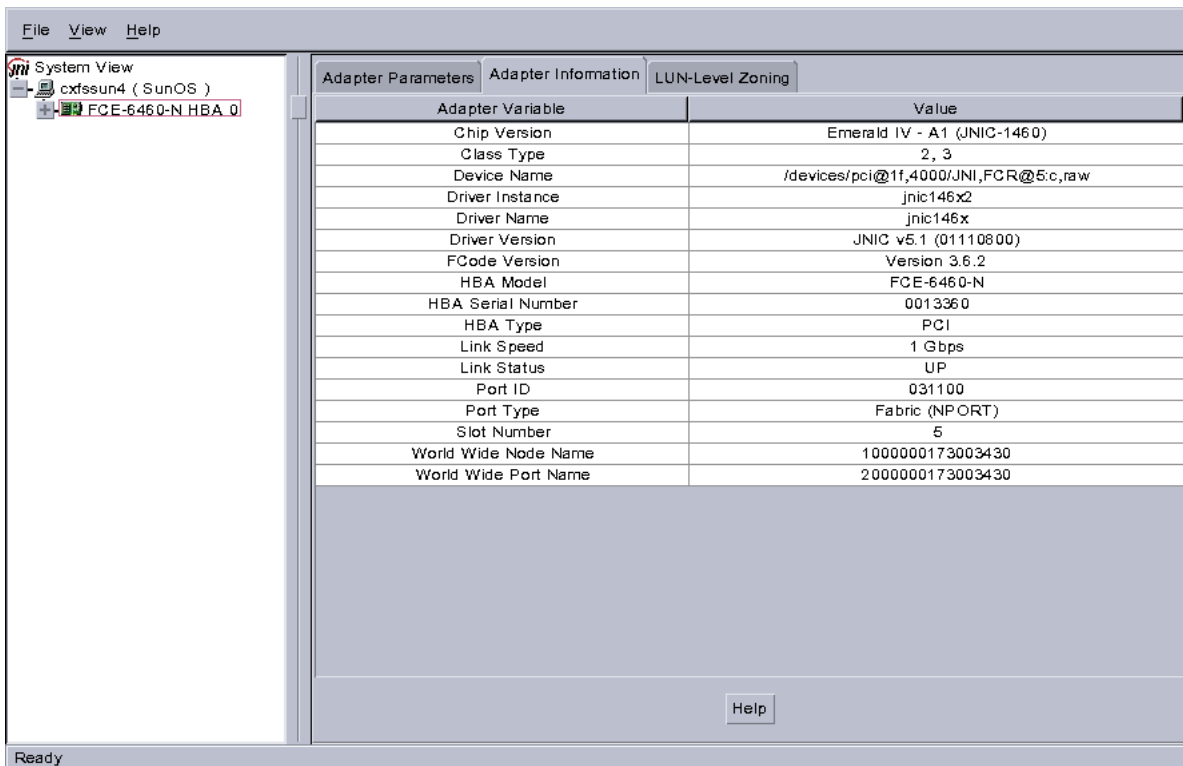


Figure 8-6 After Clicking the Adapter Information Tab

- b. Click on the **LUN-Level Zoning** tab in the left-hand pane to display a list of all the known devices on the selected HBA, as shown in Figure 8-7.

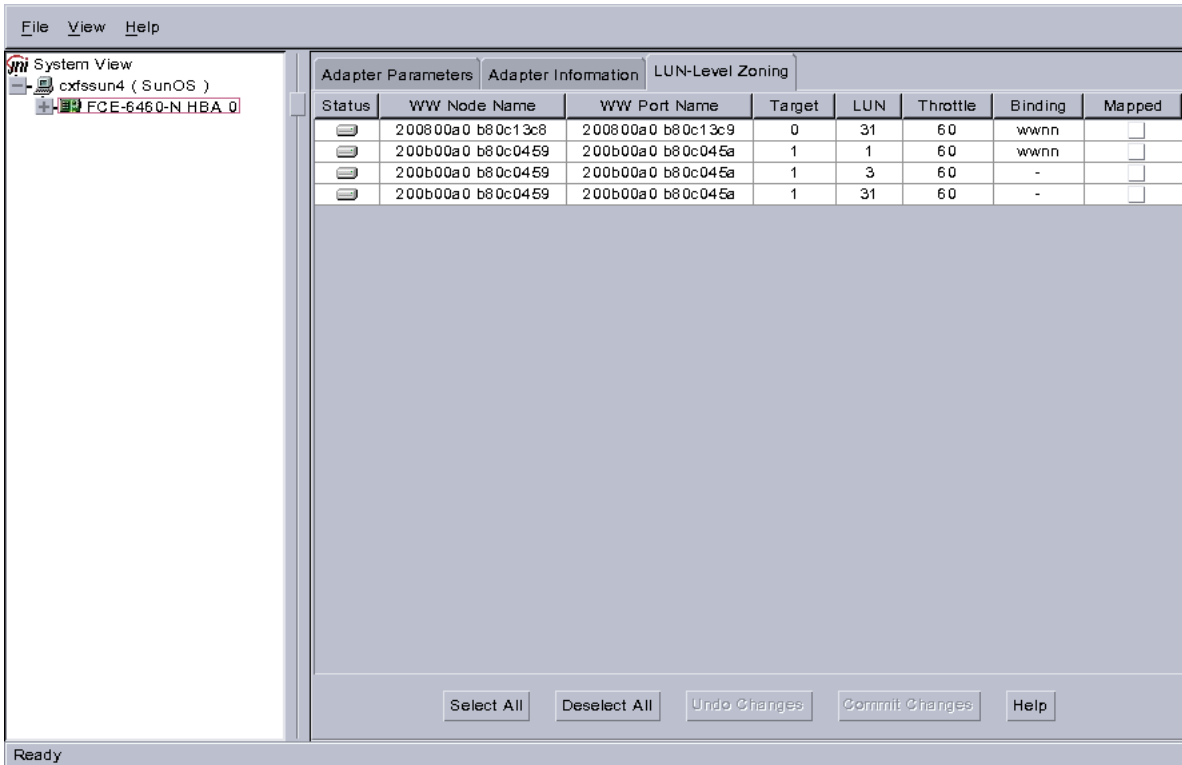


Figure 8-7 After Clicking on LUN-Level Zoning

5. Select the devices that should be accessed through the HBA.

For each device you want to access, click on the corresponding box in the **Mapped** column to make a check mark appear, as shown in Figure 8-8. After you have selected all the desired devices for the HBA, click on **Commit Changes**. The LUNs you map will depend upon your own site's needs.



Caution: In this example, LUN 31 is used for administration by the TP9400. This LUN must not be used for other purposes; do not map it or use it for XVM volumes.

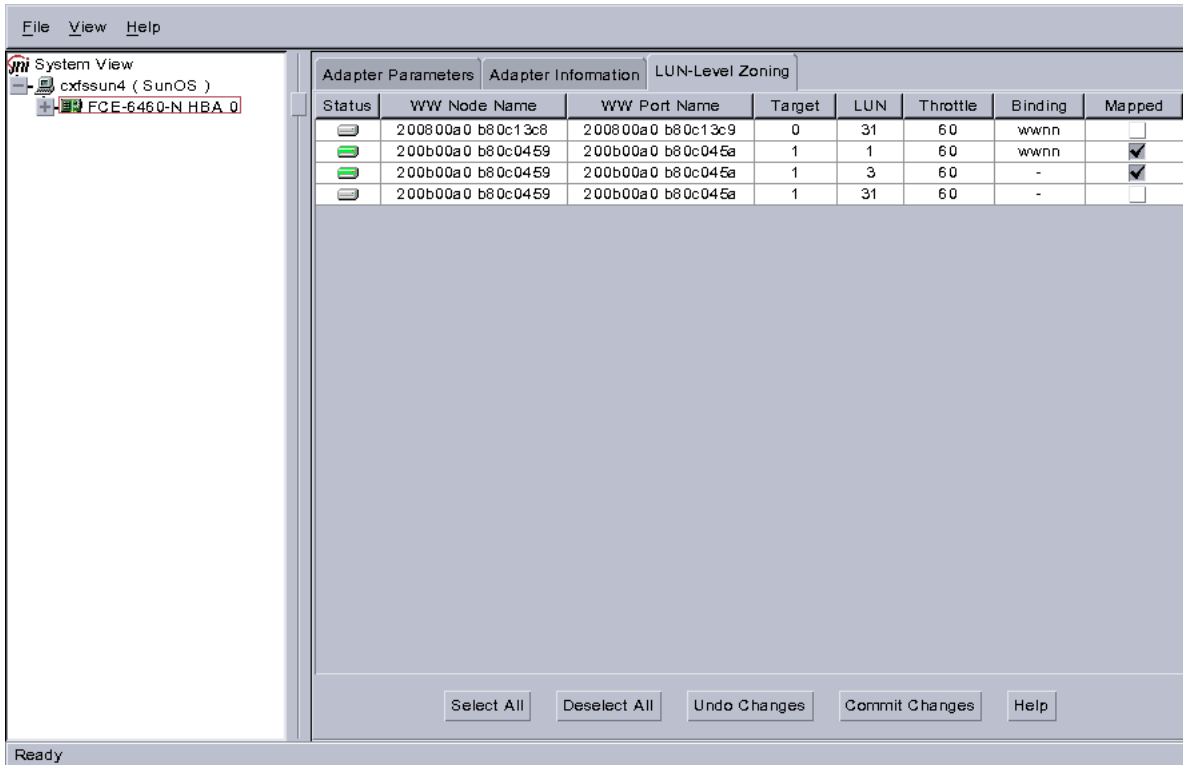


Figure 8-8 Example: After Mapping the LUNs and Committing the Changes

6. Reboot the system to make the changes take effect:

```
solaris# init 6
```

Verifying the JNI HBA Installation

After the system reboots, you should verify that the devices were correctly configured by running the Solaris `format` command. You should see a list of each device you selected.

For example:

```
solaris# format
Searching for disks... done

c4t1d1: configured with capacity of 133.99GB
c4t1d3: configured with capacity of 133.99GB
```

```
AVAILABLE DISK SELECTIONS:
  0. c0t0d0 <SUN9.0G cyl 4924 alt 2 hd 27 sec 133>
     /pci@1f,4000/scsi@3/sd@0,0
  1. c4t1d1 <SGI-TP9400-0401 cyl 65533 alt 2 hd 64 sec 67>
     /pci@1f,4000/JNI,FCR@5/sd@1,1
  2. c4t1d3 <GI-TP9400-0401 cyl 65533 alt 2 hd 64 sec 67>
     /pci@1f,4000/JNI,FCR@5/sd@1,3
Specify disk (enter its number):
```

In this example, disks 1 and 2 are being addressed by the JNI driver, as indicated by the presence of JNI,FCR in the pathname.

The system log and console display may display warning messages because the disks have IRIX labels on them. For example:

```
Mar  5 14:17:33 cxfssun4 scsi: WARNING: /pci@1f,4000/JNI,FCR@5/sd@1,1 (sd154):
Mar  5 14:17:33 cxfssun4      corrupt label - wrong magic number
Mar  5 14:17:33 cxfssun4 scsi:      Vendor 'SGI', product 'TP9400', 284203008 512 byte blocks
Mar  5 14:17:33 cxfssun4 scsi: WARNING: /pci@1f,4000/JNI,FCR@5/sd@1,3 (sd155):
Mar  5 14:17:33 cxfssun4      corrupt label - wrong magic number
Mar  5 14:17:33 cxfssun4 scsi:      Vendor 'SGI', product 'TP9400', 284203008 512 byte blocks
```

This situation will be corrected automatically by CXFS after it is installed.

Note: You should not be alarmed by the preceding messages, nor should you try to relabel the disks with the `format` command. At this point, you are only trying to achieve connectivity to the devices, and the content is not important.

If you are having trouble with the verification steps, see "Common Host Bus Adapter Problems", page 177.

Preinstallation Steps for Solaris

When you install the CXFS software on the client-only node, you must modify certain system files. **The network configuration is critical.** Each node in the cluster must be able to communicate with every other node in the cluster by both logical name and IP address without going through any other network routing; proper name resolution is key. SGI recommends static routing.

This section provides an overview of the steps that you or a qualified Sun service representative will perform on your Solaris nodes prior to installing the CXFS software. It contains the following sections:

- "Hostname Resolution and Network Configuration Rules for Solaris"
- "Adding a Private Network for Solaris Nodes", page 97
- "Verifying the Private and Public Networks for Solaris", page 102

Hostname Resolution and Network Configuration Rules for Solaris



Caution: It is critical that you understand these rules before attempting to configure a CXFS cluster.

The following hostname resolution rules and recommendations apply to client-only nodes:

- Hostnames cannot begin with an underscore (_) or include any whitespace characters.
- The private network IP addresses on a running node in the cluster cannot be changed while cluster services are active.
- You must be able to communicate directly between every node in the cluster (including client-only nodes) using IP addresses and logical names, without routing.
- A private network must be dedicated to be the heartbeat and control network. No other load is supported on this network.
- The heartbeat and control network must be connected to all nodes, and all nodes must be configured to use the same subnet for that network.

If you change hostname resolution settings in the `/etc/nsswitch.conf` file after you have defined the first IRIX node (which creates the cluster database), you must recreate the cluster database.

Adding a Private Network for Solaris Nodes

The following procedure provides an overview of the steps required to add a private network to the Solaris system.

Note: A private network is **required** for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site. For details about any of these steps, see the Solaris documentation.

1. If your system is already operational and on the network, skip to step 2.

If your Solaris system has **never** been set up, bring the system to single-user mode. For example, go to the PROM prompt and boot the Solaris node into single-user mode:

```
> boot -s
```

As a last resort, you can reach the PROM prompt by pressing the L1-A (or Stop-A) key sequence.

2. Edit the `/etc/inet/hosts` (or `/etc/hosts`) file so that it contains entries for every node in the cluster and their private interfaces as well.

The `/etc/inet/hosts` (or `/etc/hosts`) file has the following format, where *primary_hostname* can be the simple hostname or the fully qualified domain name:

```
IP_address    primary_hostname    aliases
```

You should be consistent when using fully qualified domain names in the `/etc/inet/hosts` (or `/etc/hosts`) file. If you use fully qualified domain names on a particular node, then all of the nodes in the cluster should use the fully qualified name of that node when defining the IP/hostname information for that node in their `/etc/inet/hosts` (or `/etc/hosts`) file.

The decision to use fully qualified domain names is usually a matter of how the clients (such as NFS) are going to resolve names for their client server programs, how their default resolution is done, and so on.

Even if you are using the domain name service (DNS) or the network information service (NIS), you must add every IP address and hostname for the nodes to `/etc/inet/hosts` (or `/etc/hosts`) on all nodes. For example:

```
190.0.2.1 server1-company.com server1
190.0.2.3 stocks
190.0.3.1 priv-server1
190.0.2.2 server2-company.com server2
190.0.2.4 bonds
190.0.3.2 priv-server2
```

You should then add all of these IP addresses to `/etc/inet/hosts` (or `/etc/hosts`) on the other nodes in the cluster.

For more information, see the `hosts`, `named`, and `nis` man pages.

Note: Exclusive use of NIS or DNS for IP address lookup for the nodes will reduce availability in situations where the NIS or DNS service becomes unreliable.

For more information, see "Hostname Resolution and Network Configuration Rules for Solaris", page 96.

3. Edit the `/etc/nsswitch.conf` file so that local files are accessed before either NIS or DNS. That is, the `hosts` line in `/etc/nsswitch.conf` must list files first.

For example:

```
hosts:      files nis dns
```

(The order of `nis` and `dns` is not significant to CXFS, but `files` must be first.)

4. Determine the name of the private interface by using the `ifconfig` command as follows:

```
solaris# ifconfig -a
```

If the second network does not appear, it may be that a network interface card must be installed in order to provide a second network, or it may be that the network is not yet initialized.

For example, on an Ultra Enterprise 250, the integrated Ethernet is hme0; this is the public network. The following `ifconfig` output shows that only the public interface exists:

```
solaris# 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 128.162.2.91 netmask ffffffff broadcast 128.162.2.255
    ether 8:0:20:d2:29:c5
```

If the second network does not appear, do the following:

- a. If you do not have the PCI card installed, install it. Refer to your PCI documentation for instructions.

If your card is already installed, skip to step b.

- b. Use the output from the `dmesg` command to determine the interface name for the private network; look for the network interface that immediately follows the public network; you may wish to search for `Found`. For example:

```
solaris# dmesg
Feb  6 09:38:36 ue250 last message repeated 42 times
Feb  6 11:38:40 ue250 pseudo: [ID 129642 kern.info] pseudo-device: devinfo0
Feb  6 11:38:40 ue250 genunix: [ID 936769 kern.info] devinfo0 is /pseudo/devinfo@0
Feb  6 11:38:41 ue250 hme: [ID 517527 kern.info] SUNW,hme0 : PCI IO 2.0 (Rev Id = c1) Found
Feb  6 11:38:41 ue250 genunix: [ID 936769 kern.info] hme0 is /pci@1f,4000/network@1,1
Feb  6 11:38:41 ue250 hme: [ID 517527 kern.info] SUNW,hme1 : PCI IO 2.0 (Rev Id = c1) Found
Feb  6 11:38:41 ue250 hme: [ID 517527 kern.info] SUNW,hme1 : Local Ethernet address = 8:0:20:cc:43:48
Feb  6 11:38:41 ue250 pcipsy: [ID 370704 kern.info] PCI-device: SUNW,hme@1,1, hme1
Feb  6 11:38:41 ue250 genunix: [ID 936769 kern.info] hme1 is /pci@1f,2000/SUNW,hme@1,1
```

The second network is hme1; this is the private network, and is displayed after hme0 in the `dmesg` output. In this example, hme1 is the value needed in step c and in step 5 below.

- c. Initialize the private network's interface by using the `ifconfig` command as follows, where *interface* is the value determined in step b:

```
ifconfig interface plumb
```

For example:

```
solaris# ifconfig hme1 plumb
```

After performing the `plumb`, the `hme1` interface will appear in the `ifconfig` output, although it will not contain the appropriate information (the correct information will be discovered after the system is rebooted later in step 8). For example, at this stage you would see the following:

```
solaris# 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 128.162.2.91 netmask ffffffff broadcast 128.162.2.255
    ether 8:0:20:d2:29:c5
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 0.0.0.0 netmask ff000000 broadcast 255.0.0.0
    ether 8:0:20:d2:29:c5
```

5. Create a file named `/etc/hostname.interface`, where *interface* is the value determined in step 4. This file must contain the name of the **private** network. For example:

```
solaris# cat /etc/hostname.hme1
cxfssun3-priv
```

Note: In this scenario, `/etc/hostname.hme0` must contain the same value as the `/etc/nodename` file. For example:

```
solaris# cat /etc/hostname.hme0
cxfssun3
solaris# cat /etc/nodename
cxfssun3
```

The Solaris `/etc/nodename` file is analogous to the IRIX `/etc/sys_id` file.

6. Edit the `/etc/netmasks` file to include the appropriate entries.
7. (*Optional*) Edit the `.rhosts` file if you want to use remote access or if you want to use the connectivity diagnostics provided with CXFS.

Ensure that the `.rhosts` file on each Solaris node allows all of the nodes in the cluster to have access to each other. The connectivity tests execute a `ping`

command from the local node to all nodes and from all nodes to the local node. To execute ping on a remote node, CXFS uses rsh as user root.

For example, suppose you have a cluster with three nodes: `irix0`, `sun1`, and `sun2`. The `/.rhosts` files could be as follows (the prompt denotes the node name):

```
irix0# cat /.rhosts
sun1 root
sun1-priv root
sun2 root
sun2-priv root
```

```
sun1# cat /.rhosts
irix0 root
irix0-priv root
sun2 root
sun2-priv root
```

```
sun2# cat /.rhosts
irix0 root
irix0-priv root
sun1 root
sun1-priv root
```

8. Reboot the Solaris system:

```
solaris# init 6
```

At this point, `ifconfig` will show the correct information for the private network.

For example:

```
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 128.162.2.91 netmask ffffffff broadcast 128.162.2.255
    ether 8:0:20:d2:29:c5
hme1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
    inet 10.1.1.36 netmask ffffffff broadcast 10.1.1.255
    ether 8:0:20:d2:29:c5
```

Verifying the Private and Public Networks for Solaris

For each private network on each Solaris node in the pool, verify access with the Solaris `ping` command. Enter the following, where *nodeIPAddress* is the IP address of the node:

```
solaris# /usr/sbin/ping -s -c 3 nodeIPAddress
```

For example:

```
solaris# /usr/sbin/ping -s -c 3 128.162.2.91
PING 128.162.2.91: 56 data bytes
64 bytes from cxfssun3.americas.sgi.com (128.162.2.91): icmp_seq=0. time=0. ms
64 bytes from cxfssun3.americas.sgi.com (128.162.2.91): icmp_seq=1. time=0. ms
64 bytes from cxfssun3.americas.sgi.com (128.162.2.91): icmp_seq=2. time=0. ms
64 bytes from cxfssun3.americas.sgi.com (128.162.2.91): icmp_seq=3. time=0. ms
```

Also execute a ping on the public networks. If ping fails, follow these steps:

1. Verify that the network interface was configured up using `ifconfig`; for example:

```
solaris# /usr/sbin/ifconfig eri0
eri0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
    inet 128.162.2.127 netmask ffffffff broadcast 128.162.2.255
    ether 0:3:ba:d:ad:77
```

In the first output line above, `UP` indicates that the interface was configured up.

2. Verify that the cables are correctly seated.

Repeat this procedure on each node.

Client Software Installation Steps for Solaris

The CXFS software will be initially installed and configured by SGI personnel. This section provides an overview of those procedures. You can use the information in this section to verify the installation.

Solaris Installation Overview

Installing the CXFS client CD for Solaris requires approximately 20 MB of space.

To install the required software on a Solaris node, SGI personnel will do the following:

1. Verify that the node has been upgraded to Solaris 8 (also known as SunOS 5.8) or Solaris 9 (also known as SunOS 5.9) according to the Solaris installation guide. Use the following command to display the currently installed system:

```
solaris# uname -r
```

This command should return a value of 5.8 or 5.9.

2. Insert the CXFS MultiOS CD-ROM.
3. Read the already inserted CD-ROM as follows:

- Solaris 8:

```
solaris# pkgadd -d /cdrom/cdrom01/solaris/SGIcxfs-sol8.pkg
```

- Solaris 9

```
solaris# pkgadd -d /cdrom/cdrom01/solaris/SGIcxfs-sol9.pkg
```

For example, installing `SGIcxfs-sol8.pkg` under Solaris 8 will display at least the following output, although the exact version numbers may differ:

```
solaris# pkgadd -d /cdrom/cdrom01/solaris/SGIcxfs-sol8.pkg
```

The following packages are available:

```
  1 SGIcxfs      SGI CXFS client software
                   (sparc) release 2.4
```

Select package(s) you wish to process (or `^all^` to process all packages). (default: all) [?,??,q]:

```
Processing package instance <SGIcxfs> from </cdrom/solaris/SGIcxfs-sol8.pkg>
```

```
. . .
```

4. Verify that the CXFS license key has been installed. See Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.

For example:

```
solaris# /usr/cxfs_cluster/bin/cxfslicense -d
CXFS license granted.
```

Verifying the Solaris Installation

To verify that the CXFS software has been installed properly, use the `pkginfo` command as follows:

```
pkginfo -l SGICxfs
```

For example, the following output indicates that the CXFS package installed properly:

```
% pkginfo -l SGICxfs
  PKGINST:  SGICxfs
    NAME:   SGI CXFS MultiOS client software
  CATEGORY: system
    ARCH:   sparc
  VERSION:  release 2.4
  BASEDIR:  /
  VENDOR:   Silicon Graphics Inc.
```

Postinstallation Steps for Solaris: Creating the I/O Fencing File

I/O fencing is required to protect data integrity for Solaris nodes.

On Solaris, the `cxfs_client` software detects the worldwide port names (WWPNs) of any supported host bus adapters (HBAs) in the system that are connected to a switch that is configured in the cluster database. These HBAs will then be available for fencing.

If automatic HBA WWPN detection fails according to the messages in the `/var/log/cxfs_client` log file, then the `/etc/fencing.conf` file can be used to specify the WWPN(s). If used, this file must contain a simple list of WWPNs, one per line. To determine the WWPNs, see the documentation for the driver in use.

If you want to use the `/etc/fencing.conf` file, you must update it whenever the HBA configuration changes, including the replacement of an HBA.

The `/etc/fencing.conf` file enumerates the worldwide port name for all of the JNI host bus adapters (HBA) that will be used to mount a CXFS filesystem. There

must be a line for the JNI HBA worldwide port name (WWPN) as a 64-bit hexadecimal number.

To determine the HBA WWPN, you must first set up the Brocade Fibre Channel switch and JNI HBA according to the directions in Chapter 3, "Brocade Fibre Channel Switch Verification", page 19, and "Host Bus Adapter Installation and Configuration for Solaris", page 83.

At this point, you might be able to determine the HBA WWPN by running the EZ Fibre Configuration GUI: see "Installing and Running the EZ Fibre Configuration GUI", page 85, and Figure 8-6, page 92. If so, and you are **completely certain** that you can determine the correct WWPN of the HBA (and **not** that of any of the SAN targets), you can enter this value in the `/etc/fencing.conf` file.

Note: The WWPN is that of the JNI HBA itself, **not** any of the devices that are visible to that HBA in the fabric.

If you are **not** completely certain which number you should use, do the following:

1. Follow the Fibre Channel cable on the back of the Solaris host to determine the port to which it is connected in the Brocade Fibre Channel switch. Ports are numbered beginning with 0. (For example, if there are 8 ports, they will be numbered 0 through 7.)
2. Use the `telnet` command to connect to the Brocade Fibre Channel switch and log in as user `admin` (the password is `password` by default).
3. Execute the `switchshow` command to display the switches and their WWPN numbers.

For example:

```
brocade04:admin> switchshow
switchName:      brocade04
switchType:      2.4
switchState:     Online
switchRole:      Principal
switchDomain:     6
switchId:        fffc06
switchWwn:       10:00:00:60:69:12:11:9e
switchBeacon:    OFF
port 0: sw Online      F-Port 20:00:00:01:73:00:2c:0b
port 1: cu Online      F-Port 21:00:00:e0:8b:02:36:49
```

```
port 2: cu Online F-Port 21:00:00:e0:8b:02:12:49
port 3: sw Online F-Port 20:00:00:01:73:00:2d:3e
port 4: cu Online F-Port 21:00:00:e0:8b:02:18:96
port 5: cu Online F-Port 21:00:00:e0:8b:00:90:8e
port 6: sw Online F-Port 20:00:00:01:73:00:3b:5f
port 7: sw Online F-Port 20:00:00:01:73:00:33:76
port 8: sw Online F-Port 21:00:00:e0:8b:01:d2:57
port 9: sw Online F-Port 21:00:00:e0:8b:01:0c:57
port 10: sw Online F-Port 20:08:00:a0:b8:0c:13:c9
port 11: sw Online F-Port 20:0a:00:a0:b8:0c:04:5a
port 12: sw Online F-Port 20:0c:00:a0:b8:0c:24:76
port 13: sw Online L-Port 1 public
port 14: sw No_Light
port 15: cu Online F-Port 21:00:00:e0:8b:00:42:d8
```

The WWPN is the hexadecimal string to the right of the port number. For example, the WWPN for port 0 is 2000000173002c0b (you must remove the colons from the WWPN reported in the `switchshow` output to produce the string to be used in the `/etc/fencing.conf` file).

4. Edit or create the `/etc/fencing.conf` file and add the WWPN for the port determined in step 1. (Comment lines begin with #.)

For example, if you determined that port 0 is the port connected to the Brocade Fibre Channel switch, your `/etc/fencing.conf` file should appear as follows:

```
# WWPN of the JNI HBA installed on this system
#
2000000173002c0b
```

5. After the Solaris node is added to the cluster (see Chapter 10, "Cluster Configuration", page 159), enable the fencing feature by using the CXFS GUI or `cmgr` command on an IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Manual CXFS Startup/Shutdown for Solaris

The `/etc/init.d/cxfs_cluster` script will be invoked automatically during normal system startup and shutdown procedures. This script starts and stops the processes required to run CXFS.

To start up CXFS processes manually on your Solaris node, enter the following:

```
solaris# /etc/init.d/cxfs_cluster start
```

To stop CXFS processes manually, enter the following:

```
solaris# /etc/init.d/cxfs_cluster stop
```

Software Maintenance for Solaris

This section contains the following:

- "Upgrading the CXFS Software on a Solaris System"
- "Modifying the CXFS Software on a Solaris System", page 107

Upgrading the CXFS Software on a Solaris System

Before upgrading CXFS software, ensure that no applications on the node are accessing files on a CXFS filesystem. You can then run the new CXFS software package, which will automatically upgrade all CXFS software.

Modifying the CXFS Software on a Solaris System

You can modify the CXFS client service (`/usr/cxfs_cluster/bin/cxfs_client`) by placing options in the `/usr/cxfs_cluster/bin/cxfs_client.options` file. The available options are documented in the `cxfs_client` man page.



Caution: Some of the options are intended to be used internally by SGI only for testing purposes and do not represent supported configurations. Consult your SGI service representative before making any changes.

The first line in the `cxfs_client.options` file must contain the options you want `cxfs_client` to process; you cannot include a comment as the first line.

To see if `cxfs_client` is using the options in `cxfs_client.options`, enter the following:

```
solaris# ps -ef | grep cxfs
```


Windows NT and Windows 2000 Platforms

CXFS supports a client-only node running the Windows NT or Windows 2000 operating system. The information in this chapter applies to both Windows NT and Windows 2000 nodes unless otherwise noted. This chapter contains the following sections:

- "CXFS on Windows"
- "Host Bus Adapter Installation for Windows", page 129
- "Preinstallation Steps for Windows", page 133
- "Client Software Installation Steps for Windows", page 138
- "Postinstallation Steps for Windows", page 145
- "Manual CXFS Startup/Shutdown for Windows", page 154
- "Software Maintenance for Windows", page 154

CXFS on Windows

This section contains the following information about CXFS on Windows:

- "Requirements Specific to Windows", page 110
- "CXFS Commands Installed on Windows", page 111
- "Windows Log Files and Cluster Status", page 111
- "Functional Limitations Specific to Windows", page 112
- "Maximum CXFS Filesystem and File Size on Windows", page 114
- "Performance Considerations on a CXFS Windows Node", page 114
- "Access Controls on a Windows Node", page 115

Requirements Specific to Windows

In addition to the items listed in "Requirements", page 6, using a Windows node to support CXFS requires the insertion of a Windows host with at least the following:

- An Intel Pentium or compatible processor.
- A minimum of 128 MB of RAM (more will improve performance).
- A minimum of 10 MB of free disk space.
- A QLogic 2200, QLogic 2310, or QLogic 2342 host bus adapter (HBA).
- The following QLogic software from the <http://www.qlogic.com> website:
 - QLogic driver and BIOS:
 - QLogic 2200 driver version 8.1.5.12 or later and BIOS 1.76 or later.
 - QLogic 2310/2342 driver version 8.2.0.10 or later and BIOS 1.33 or later.

Note: If your configuration has dual HBAs, you should keep the HBA firmware and HBA driver at the same version on both cards.

- QLogic SANSurfer 2.0.15 or later. You must install both the SANblade NT Agent and the SANblade Manager.

You should install the documentation associated with the software. See the SANblade Manager `README` for the default password. Follow the QLogic instructions to install the driver, the SANblade NT Agent, and the SANblade Manager software. If you do not have the correct QLogic BIOS version installed, see the procedure in "Upgrading the QLogic BIOS", page 134.

- If two QLogic HBAs are installed, you should also install the QLdirect Filter (8.1.3 or later) in order to facilitate HBA failover and load balancing. If two different model HBAs are installed, you must install drivers for both models.

Note: If the primary HBA path is at fault during the Windows boot up (for example, if the Fibre Channel cable is disconnected), no failover to the secondary HBA path will occur. This is a limitation of the QLogic driver.

- Windows NT 4.0 Service Pack 6, or Windows 2000 Service Pack 2 or later.

CXFS Commands Installed on Windows

A single CXFS service and a single CXFS filesystem driver are installed as part of the Windows installation. The service and the CXFS filesystem driver can be configured to run automatically when the first user logs into the node.

The command `C:\Program Files\CXFS\cxfslicense` is installed to assist with license validation; see "Configuring the FLEXlm License for Windows", page 146.

The command `C:\Program Files\CXFS\cxfs_info` is installed to display in a human-readable format the current state of the node in the cluster; see "Windows Log Files and Cluster Status", and "Verifying the Cluster", page 165.

Windows Log Files and Cluster Status

The Windows node will log important events in the system event log. You can view these events by selecting the following:

- For Windows NT:

Start
 > **Programs**
 > **Administrative Tools**
 > **Event Viewer**

- For Windows 2000:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Event Viewer**

For information about the log files created on IRIX nodes, see the *CXFS Version 2 Software Installation and Administration Guide*. The CXFS Client service will also log important information to the following file:

`C:\Program Files\CXFS\log\cxfs_client.log`

You may also wish to keep the **CXFS** window open to check the cluster status. To open this informational window on any Windows system, select the following:

```
Start
  > Programs
    > CXFS
      > CXFS Info
```

Functional Limitations Specific to Windows

There are a number of limitations in the CXFS software that are unique to the Windows platform. These limitations are described from both a UNIX and a Windows perspective in the following sections.

UNIX Perspective of CXFS on a Windows Node

This section describes the differences and limitations of a CXFS filesystem on a Windows node from a UNIX perspective:

- Windows nodes can support multiple CXFS filesystems mounted under a single drive letter. Only one CXFS drive letter may be configured on a Windows node.

The top-level file structure under the CXFS drive letter consists of an in-memory directory structure that mimics the mount points on the IRIX server. The CXFS software creates these directories before mounting the CXFS filesystems. This file structure supports only creating and deleting directories; there is no support for creating and deleting regular files, renaming directories, and so on. Attempts to perform unsupported actions will generally result in an invalid parameter error. You can perform normal filesystem operations on files and directories beneath the mount points.

- A Windows node can support regular files, directories, and links. However, it does not support other XFS file types.
- Symbolic links cannot be distinguished from normal files or directories on a Windows node. Opening a symbolic link will open the target of the link, or will report `file not found` if it is a dangling link.
- You can move, rename, or delete a symbolic link; however, you cannot copy a symbolic link. Copying a valid symbolic link will result in copying the file or directory that the link refers to, rather than the normal UNIX behavior that copies the link itself.

- CXFS Windows nodes cannot use the `edquota(1M)` or `repquota(1M)` commands to view or edit user and group quotas because CXFS administration must be performed on an IRIX node. However, user and group quotas will be enforced correctly, independent of the node in the CXFS cluster where files are created and deleted.

Windows Perspective of CXFS on a Windows Node

This section describes the differences and limitations of a CXFS filesystem on a Windows node in comparison to other Windows filesystems from a Windows perspective:

- CXFS filesystems should not be shared to another Windows host from a CXFS Windows node because there is no support for the opportunistic locking that is used by Windows to guarantee data coherency. If this is required, it is recommended that an IRIX server share the filesystem via Samba.
- Avoid using duplicate filenames in the same directory that vary only in case. CXFS is case sensitive, but some Windows applications may not maintain the case of all filenames, which may result in unexpected behavior.
- CXFS software does not export 8.3 alternative filenames. Older Windows applications that only support 8.3 filenames may be unable to open files with longer filenames.
- Avoid using completely uppercase 8.3 filenames. If you use completely uppercase 8.3 filenames, some applications (including Windows Explorer) may incorrectly assume that only 8.3 filenames are supported by the filesystem and will not preserve case.
- Take care when using Disk Manager and other similar Microsoft and third-party disk tools under Windows NT. These tools assume particular disk formats (FAT or NTFS) and may even write to the super block of a Fibre Channel disk and write a Windows signature on the disk. This will corrupt the XVM volume information on the disk. A CXFS filter driver prevents other tools from writing to the disks under Windows 2000.



Caution: Writing a Windows signature to the disk used for CXFS will corrupt the volume and filesystem structure on that disk, potentially resulting in the loss of data on the disk.

- Install the CXFS software components onto a NTFS partition rather than a FAT partition. The security of the following files cannot be guaranteed if these files are installed onto a FAT filesystem:

```
C:\Program Files\CXFS\passwd  
C:\Program Files\CXFS\group  
C:\Program Files\CXFS\fencing.conf
```

- There is no recycle bin; deleted files are permanently deleted.
- There is no automatic notification of directory changes performed by other nodes in the cluster. Applications (such as Windows Explorer) will not automatically update their display if another node adds or removes files from the directory currently displayed.
- A CXFS filesystem cannot be used as the boot partition of a Windows host.
- The volume properties window for the CXFS drive letter will display the total capacity of all mounted filesystems and the largest free space on any one of those filesystems.

Forced Unmount on a Windows Node

SGI recommends that you enable the forced unmount feature on CXFS filesystems. See "Recommendations", page 8, and "Forced Unmount of CXFS Filesystems", page 167.

A forced unmount causes all processes that have open files on the specified filesystem to be unconditionally killed and therefore permit the filesystem to be unmounted without delay.

Maximum CXFS Filesystem and File Size on Windows

The maximum size of a CXFS filesystem on Windows is 2^{64} bytes. The maximum size of a file is 2^{63-1} bytes. An attempt to write beyond this limit will result in an Invalid argument or a File too large error.

Performance Considerations on a CXFS Windows Node

The following are performance considerations on a CXFS Windows node, in addition to the limitations described in "Performance Considerations", page 5:

- If you open the Windows Explorer **Properties** window on a directory, it will attempt to traverse the filesystem in order to count the number and size of all subdirectories and files; this action is the equivalent of running the UNIX `du` command. This can be an expensive operation, especially if performed on directories between the drive letter and the mount points, because it will traverse all mounted filesystems.
- Virus scanners, Microsoft Find Fast, and similar tools that traverse a filesystem are very expensive on a CXFS filesystem. Such tools should be configured so that they do not automatically traverse the CXFS drive letter.
- The mapping from Windows user and group names to UNIX identifiers occurs as the CXFS software starts up. In a Windows domain environment, this process can take a number of seconds per user for usernames that do not have accounts within the domain. If you are using a `passwd` file for user identification and the file contains a number of unknown users on the Windows host, you should remove users who do not have accounts on the Windows nodes from the `passwd` file that is installed on the Windows nodes.

This issue has less impact on Windows nodes in a workgroup than on those in a domain because the usernames can be quickly resolved on the node itself, rather than across the network to the domain controller.

- With 1-GB fabric to a single RAID controller, it is possible for one 32-bit 33-MHz QLogic card to reach the bandwidth limitations of the fabric, and therefore there will be no benefit from load balancing two HBAs in the same PCI bus. This can be avoided by using 2-GB fabric and/or multiple RAID controllers.
- For load balancing of two HBAs to be truly beneficial, the host must have at least one of the following three attributes:
 - A 64-bit PCI bus
 - A 66-MHz PCI bus
 - Multiple PCI buses

Access Controls on a Windows Node

The XFS filesystem used by CXFS implements and enforces UNIX mode bits and POSIX access control lists (ACLs), which are quite different from Windows file attributes and access control lists. The CXFS software attempts to map Windows access controls to the UNIX access controls for display and manipulation, but there

are a number of features that are not supported (or may result in unexpected behavior) that are described here.

User Identification on a Windows Node

The CXFS software supports several user identification mechanisms, which are described in "User Identification Mapping Methods", page 117. Windows user and group names that match entries in the configured user list will be mapped to those user IDs (UIDs) and group IDs (GIDs).

The following additional mappings are automatically applied:

- **User Administrator** is mapped to `root`
- **Group Administrators** is mapped to `sys`

A user's default UNIX GID is the default GID in the `passwd` listing for the user and is not based on a Windows group mapped to a UNIX group name.

You can display the users and groups that have been successfully mapped by running the following in a Windows command shell:

```
C:\Program Files\CXFS\cxfs_info -Dug
```

For example, suppose a CXFS Windows node was configured with the following `passwd` and `group` files:

```
C:\> type C:\Program Files\CXFS\passwd
root::0:0:Super-User:/root:/bin/tcsh
guest::998:998:Guest Account:/usr/people/guest:/bin/csh
fred::1040:402:Fred Costello:/users/fred:/bin/tcsh
diane::1052:402:Diane Green:/users/diane:/bin/tcsh
```

```
C:\> type C:\Program Files\CXFS\group
sys::0:root,bin,sys,adm
root::0:root
guest:*:998:
video::402:fred,diane
audio::403:fred
```

The following output indicates that there are five nodes configured for the cluster, with four currently in the cluster, three filesystems mounted, a number of recognized users (Administrator, fred, diane, and guest) and three groups (Administrators, video, and audio). The default group for user guest is also

named guest. (In this example, the guest group did not map to any known Windows group.)

```

cxfs_client status [timestamp May 02 12:46:17 / generation 42]

CXFS client:
  state: stable (2), cms: up, xvm: up, fs: up
Cluster:
  cluster250 (1) - enabled
Local:
  cxfs1 (5) - enabled
Nodes:
  cxfs1      enabled up    5
  cxfs2      enabled DOWN  1
  cxfs250    enabled up    0
  cxfs3      enabled up    2
  cxfs4      enabled up    4
Filesystems:
  cxfs      enabled mounted      cxfs      /mnt/cxfs
Usermaps:
  diane          ( 1052) : video          ( 402)
  fred           ( 1040) : video          ( 402)
  guest          (  998) :              ( 998)
  Administrator  (    0) : Administrators (    0)
Groups:
  audio          (  403)
  Administrators (    0)
  video          (  402)

```

User Identification Mapping Methods

User identification can be performed by one of the following methods:

- **files:** /etc/passwd and /etc/group files from the meta-data server copied onto the clients. If you select this method, you must install the /etc/passwd and /etc/group files immediately after installing the CXFS software, as described in "Performing User Configuration", page 149.
- **ldap_actedir:** Windows Active Directory server with Services for UNIX installed, which uses lightweight directory access protocol (LDAP).

- **ldap_generic:** Generic LDAP lookup for UNIX users and groups.

You must select one of these as the primary mapping method during installation, but you can change the method at a later time, as described in "Modifying the CXFS for Windows Software", page 155.

Optionally, you can select a secondary mapping method that will be applied to users that are not covered by the first method. If you choose a primary and a secondary mapping method, one of them must be **files**.

For example, in Figure 9-2, page 141, the user has selected **ldap_generic** as the primary method and **files** as the secondary method. A user mapping will be created for all suitable **ldap_generic** users and this mapping will be extended with any additional users found in the secondary method (**files**). The primary method will be used to resolve any duplicate entries.

Suppose the primary method (**ldap_generic**) has users for UIDs 1, 2 and 3, and the secondary method (**files**) has users for UIDs 2 and 4. The username for UIDs 1, 2 and 3 will be determined by the **ldap_generic** method and the username for UID 4 will be determined by the **files** method. If the LDAP lookup failed (such as if the LDAP server was down), a user mapping for UIDs 2 and 4 would be generated using the **files** method.

The default behavior is to use the **files** method to map Windows usernames to UNIX UIDs and GIDs, with no secondary method selected.

Note: LDAP options are not supported and will not be displayed under Windows NT 4.0.

The **ldap_actedir** method configures the CXFS Windows software to communicate with the Active Directory for the CXFS node's domain. With the Windows Services for UNIX (SFU) extensions, the Active Directory User Manager lets you define UNIX identifiers for each user and export these identifiers as an LDAP database.

This configuration requires a domain controller that is installed with the following:

- Microsoft Windows 2000 Server or Windows 2000 Advanced Server
- Windows 2000 Active Directory
- Microsoft Windows Services for UNIX (SFU) Version 2 or 3, including the NFS Server option

Note: The domain controller does not have to be a CXFS node.

The `ldap_generic` method configures the CXFS software to communicate with an LDAP database that maps user names and group names to UNIX identifiers.

Regardless of the method used, the consistent mapping of usernames is a requirement to ensure consistent behavior on all CXFS nodes. Most platforms can be configured to use an LDAP database for user identification.

Enforcing Access to Files and Directories

Access controls are enforced on the CXFS meta-data server by using the mapped UID and GID of the user attempting to access the file. Therefore, a user can expect the same access on a Windows node as any other node in the cluster when mounting a given filesystem. Access is determined using the file's ACL, if one is defined, otherwise by using the file's mode bits.

ACLs that are set on any files or directories are also enforced as they would be on any IRIX node. The presentation of ACLs is customized to the interfaces of Windows Explorer, so the enforcement of the ACL may vary from an NTFS ACL that is presented in the same way. A new file will inherit the parent directory default ACL, if one is defined.

The user `Administrator` has read and write access to all files on a CXFS filesystem, in the same way that `root` has super user privileges on a UNIX node.

The following example is a directory listing on the meta-data server:

```
irix# ls -l .
drwxr-x---  2 fred   video           6 Nov 20 13:33 dir1
-rw-r----- 1 fred   audio           0 Nov 20 12:59 file1
-rw-rw-r--  1 fred   video           0 Nov 20 12:59 file2
```

Users will have the following access to the contents of this directory:

- `file1` will be readable and writable to user `fred` and `Administrator` on a CXFS Windows node. It can also be read by other users in group `audio`. No other users, including `diane` and `guest`, will be able to access this file.
- `file2` will be readable by all users, and writable by user `fred`, `diane` (because she is in group `video`), and `Administrator`.

- `dir1` will be readable, writable, and searchable by user `fred` and `Administrator`. It will be readable and searchable by other users in group `video`, and not accessible by all other users.

Viewing and Changing File Attributes with Windows Explorer

File permissions may be viewed and manipulated in two different ways when using Windows Explorer:

- By displaying the list of attributes in a detailed directory listing; this is the most limited approach
- By selecting properties on a file

The only file attribute that is supported by CXFS is the read-only attribute, other attributes will not be set by CXFS and changes to those attributes will be ignored.

If the user is not permitted to write to the file, the read-only attribute will be set. The owner of the file may change this attribute and modify the mode bits. Other users, including the user `Administrator`, will receive an error message if they attempt to change this attribute.

Marking a file read-only will remove the write bit from the user, group, and other mode bits on the file. Unsetting the read-only attribute will make the file writable by the owner only.

For example, selecting file properties on `file1` using Windows Explorer on a CXFS Windows node will display the read-only attribute unset if logged in as `Administrator` or `fred`, and it will be set for `diane` and `guest`.

Only user `fred` will be able to change the attribute on these files, which will change the files under UNIX to the following:

```
-r--r----- 1 fred  audio          0 Nov 20 12:59 file1
-r--r--r--  1 fred  video          0 Nov 20 12:59 file2
```

If `fred` then unset these flags, only he could write to both files:

```
-rw-r----- 1 fred  audio          0 Nov 20 12:59 file1
-rw-r--r--  1 fred  video          0 Nov 20 12:59 file2
```


Viewing and Changing File Permissions with Windows Explorer

By selecting the **Security** tab in the **File Properties** window of a file, a user may view and change a file's permissions with a high level of granularity. However, the Windows Explorer interface is significantly different between Windows NT and Windows 2000.

Windows Explorer will list the permissions of the file's owner, the file's group, and the Everyone group, which represents the mode bits for other users. Not all Windows permission flags are supported.

Windows NT

Under Windows NT, `file1` will be displayed with the following permissions:

<code>audio</code>	<code>Special Access(R)</code>
<code>Everyone</code>	<code>No Access(None)</code>
<code>fred (Fred Costello)</code>	<code>Special Access(RWPO)</code>

Likewise, `file2` is displayed as follows:

<code>fred (Fred Costello)</code>	<code>Special Access(RWPO)</code>
<code>Everyone</code>	<code>Special Access(R)</code>
<code>video</code>	<code>Special Access(R)</code>

Windows NT uses the term `Special Access` because the combination permission flags are not a standard Windows NT set, such as `Read`, `Write`, or `Full Control`. Selecting the group `video` for `file2` and then **Special Access...** will raise a dialog with the Windows NT permission flags

CXFS maps the UNIX read mode bit to the Windows NT read (R) flag, write mode bit to the write (W) flag, and execute mode bit to the execute (X) flag. The Windows NT delete (D) flag is not set and is ignored because under UNIX this depends on the permissions of the parent directory. The change permissions (P) and the take ownership (O) flags are only set for the owner of the file or the user `Administrator`, and will be ignored if set on other users or groups.

Opening properties on `dir1` will display the following permissions:

<code>Everyone</code>	<code>Special Access(None)*(Not Stated)</code>
<code>fred (Fred Costello)</code>	<code>Special Access(RWXPO)*(Not Stated)</code>
<code>video</code>	<code>Special Access(RX)*(Not Stated)</code>

The asterisk and the second set of brackets containing `Not Stated` indicates that there is no default ACL that will be inherited when creating files or subdirectories in that directory.

Windows 2000

Under Windows 2000, the permissions on `file1` are displayed as follows:

```
audio (cxfs1\audio)           Allow: Read           Deny:
Everyone                     Allow:                Deny: All
Fred Costello (cxfs1\fred)   Allow: Read, Write   Deny:
```

Using the **Advanced** button, `file1` is displayed as follows:

```
Deny  Everyone                Full Control
Allow Fred Costello (cxfs1\fred) Special
Allow audio (cxfs1\audio)     Read
```

User `Everyone` represents all users that are not explicitly listed. In this case, all other users have `Deny` and `Full Control`, which indicates that they are denied all access.

User `fred` is listed as having `Special` access because the permission flags in the next example do not exactly match the standard Windows permissions for read and write access to a file. Select `Fred Costello` and then click **View/Edit** to display the permission flags listed in Table 9-1. (The table displays the permissions in the order in which they appear in the **View/Edit** window). You can choose to allow or deny each flag, but some flags will be ignored as described in Table 9-1.

Table 9-1 Permission Flags that May Be Edited

Permission	Description
Traverse Folder / Execute File	Used to display and change the execute mode bit on the file or directory
List Folder / Read Data	Used to display and change the read mode bit on the file or directory
Read Attributes	Set if the read mode bit is set; changing this flag has no effect
Read Extended Attributes	Set if the read mode bit is set; changing this flag has no effect

Permission	Description
Create Files / Write Data	Used to display and change the write mode bit on the file or directory
Create Folders / Append Data	Set if the write mode bit is set; changing this flag has no effect
Write Attributes	Set if the write mode bit is set; changing this flag has no effect
Write Extended Attributes	Set if the write mode bit is set; changing this flag has no effect
Delete Subfolders and Files	Set for directories if you have write and execute permission on the directory; changing this flag has no effect
Delete	Never set (because delete depends on the parent directory permissions); changing the flag has no effect
Read Permissions	Always set; changing the flag has no effect
Change Permissions	Always set for the owner of the file and the user Administrator; changing this flag has no effect
Take Ownership	Always set for the owner of the file and the user Administrator; changing this flag has no effect

The permissions for file2 are displayed as follows:

Everyone	Allow: Read	Deny:
video (cxfs1\video)	Allow: Read, Write	Deny:
Fred Costello (cxfs1\fred)	Allow: Read, Write	Deny:

The permissions for dir1 under Windows 2000 are displayed as follows:

Fred Costello (cxfs1\fred)	Allow:	Deny:
Video (cxfs1\video)	Allow:	Deny:

Note: In this example, the permission flags for directories do not match any of the standard permission sets, therefore no allow or deny flags are set.

In general, you will need to click the **Advanced** button under Windows 2000 to see the actual permissions of directories. For example:

Deny	Everyone	Full Control	This folder only
Allow	Fred Costello	Special	This folder only
Allow	audio	Read & Execute	This folder only

The `dir1` directory does not have a default ACL, so none of these permissions are inherited, as indicated by the `This folder only` tag, when a new subdirectory or file is created.

Viewing and Changing File Access Control Lists (ACLs)

If the file or directory has an ACL, the list may include other users and groups, and the `CXFS ACL Mask` group that represents the IRIX ACL mask. See the `chacl(1)` man page for an explanation of IRIX ACLs and the mask bits. The effective permissions of all entries except the owner and the `Everyone` group will be the intersection of the listed permissions for that user or group and the mask permissions. Therefore, changing the `CXFS ACL Mask` permissions will set the maximum permissions that other listed users and groups may have. Their access may be further constrained in the specific entries for those users and groups.

By default, files and directories do not have an ACL, only mode bits, but an ACL will be created if changes to the permissions require an ACL to be defined. For example, granting or denying permissions to another user or group will force an ACL to be created. Once an ACL has been created for a file, the file will continue to have an ACL even if the permissions are reduced back to only the owner or group of the file. The `chacl(1)` command under IRIX can be used to remove an ACL from a file.

Windows NT

For example, `fred` grants `diane` read access to `file1` by adding user `diane` using the file properties dialogs. The access list under Windows NT will then appear as follows:

```
fred (Fred Costello)  Special Access(RWPO)
audio                Special Access(R)
Everyone             Special Access(None)
diane (Diane Green)  Read(RX)
```

The Windows NT read permission set includes execute, so to change this to read-only you must select **Special Access ...** and deselect **Execute**. Pressing **OK** will cause a dialog to be raised claiming that you have denied everyone access to the file. Under NTFS, this ACL will deny access to everyone including the other listed users and groups, because the deny access on user `Everyone` will be applied before all other ACL entries. You can safely ignore this warning under CXFS (because ordering is not

important to a UNIX ACL) and click **Yes** to continue. This will cause an ACL to be added to the file:

```
> ls -ld
-rw-r-----+ 1 fred      audio          0 Nov 20 12:59 file1
[u:diane:r--,g::r--,u::rw-,o::---,m::r--]
```

Opening the file permission dialog again on file1 will display the following:

audio	Special Access(R)
CXFS ACL Mask	Special Access(R)
diane (Diane Green)	Special Access(R)
Everyone	Special Access(None)
fred (Fred Costello)	Special Access(RWPO)

Windows 2000

For example, fred grants diane read access to file1 by adding user diane using the file properties dialogs, and then deselecting Read & Execute so that only Read is selected. The access list under Windows 2000 now appears as follows:

audio (cxfs1\audio)	Allow: Read	Deny:
Diane Green (cxfs1\diane)	Allow: Read	Deny:
Everyone	Allow:	Deny: All
Fred Costello (cxfs1\fred)	Allow: Read, Write	Deny:

After clicking **OK**, the properties for file1 will also include the CXFS ACL Mask displayed as follows:

audio (cxfs1\audio)	Allow: Read	Deny:
CXFS ACL Mask (cxfs1\CXFS...)	Allow: Read	Deny:
Diane Green (cxfs1\diane)	Allow: Read	Deny:
Everyone	Allow:	Deny: All
Fred Costello (cxfs1\fred)	Allow: Read, Write	Deny:

You must select entries in **either** the Allow or the Deny column, but not in both columns at the same time, as the CXFS software will merge these entries, which may result in a different set of permissions than was intended. If all entries are deselected, such as when **All Clear** is selected, the CXFS software will generate an ACL that denies access to all users.

Effective Access

The effective access of user `diane` and group `audio` is read-only. Granting write access to user `diane` as in the following example does not give `diane` write access because the mask remains read-only. However, because user `fred` is the owner of the file, the mask does not apply to his access to `file1`.

Windows NT

For example:

<code>audio</code>	<code>Special Access(R)</code>
<code>CXFS ACL Mask</code>	<code>Special Access(R)</code>
<code>diane (Diane Green)</code>	<code>Special Access(RW)</code>
<code>fred (Fred Costello)</code>	<code>Special Access(RWPO)</code>

Windows 2000

For example:

<code>audio (cxfs1\audio)</code>	<code>Allow: Read</code>	<code>Deny:</code>
<code>CXFS ACL Mask (cxfs1\CXFS...)</code>	<code>Allow: Read</code>	<code>Deny:</code>
<code>Diane Green (cxfs1\diane)</code>	<code>Allow: Read, Write</code>	<code>Deny:</code>
<code>Fred Costello (cxfs1\fred)</code>	<code>Allow: Read, Write</code>	<code>Deny:</code>

Restrictions with file ACLs on Window nodes

If the users and groups listed in a file's permissions (whether mode bits and/or ACL entries) cannot be mapped to users and groups on the Windows node, attempts to display the file permissions in a file properties window will fail with an unknown user or group error. This prevents the display of an incomplete view, which could be misleading.

Both the owner of the file and the user `Administrator` may change the permissions of a file or directory using Windows Explorer. All other users will get a `permission denied` error message.

Note: A user must use a node that is **not** running Windows to change the ownership of a file because a Windows user takes ownership of a file with Windows Explorer, rather than the owner giving ownership to another user (which is supported by the UNIX access controls).

Inheritance and Default ACLs on a Windows node

When a new file or directory is created, normally the mode bits are set using a mask of 022. Therefore, a new file has a mode of 644 and a new directory of 755. This mask is defined in the registry of the CXFS driver and may be configured to other values (typically 000 or 002):

```
HKEY_LOCAL_MACHINE->SYSTEM->CurrentControlSet->Services->CXFS->Parameters->DefaultUmask
```

Therefore, creating a file on the IRIX meta-data server results in a mode of 644 for a mask of 022:

```
irix% ls -lda .
drwxr-xr-x   3 fred      video           41 Nov 21 18:01 ./

irix% umask
22

irix% touch file3
irix% ls -l file3
-rw-r--r--   1 fred      video           0 Nov 21 18:23 file3
```

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

Creating a file in Windows explorer on a Windows node will have the same result.

An IRIX directory ACL may include a default ACL that is inherited by new files and directories, instead of applying the `umask`. Default ACLs are displayed in the Windows Explorer file permission window if they have been set on a directory. Unlike a Windows inheritable ACL on an NTFS filesystem, an IRIX default ACL applies to both new files and subdirectories, there is no support for an inheritable ACL for new files and another ACL for new subdirectories.

The following example applies an ACL and a default ACL to `dir1` and then creates a file and a directory in `dir1`:

```
irix% chacl -b "u::rwx,g::r-x,u:diane:r-x,o::---,m::r-x" \
           "u::rwx,g::r-x,u:diane:rwx,o::---,m::rwx" dir1
irix% touch dir1/newfile
irix% mkdir dir1/newdir
irix% ls -D dir1
newdir [u::rwx,g::r-x,u:diane:rwx,o::---,m::r-x/
       u::rwx,g::r-x,u:diane:rwx,o::---,m::rwx]
newfile [u::rw-,g::r-x,u:diane:rwx,o::---,m::r--]
```

The permissions for dir1 will be as follows:

- Windows NT:

CXFS ACL Mask	Special Access(RX)*(RWX)
diane (Diane Green)	Special Access(RX)*(RWX)
Everyone	Special Access(None)*(None)
fred (Fred Costello)	Special Access(RWXPO)(RWXPO)
video	Special Access(RX)*(RX)

- Windows 2000:

CXFS ACL Mask (cxfs1\CXFS...)	Allow:	Deny:
Diane Green (cxfs1\diane)	Allow:	Deny:
Everyone	Allow:	Deny: All
Fred Costello (cxfs1\fred)	Allow: Read & Exec, List, Read, Write	Deny:
Video (cxfs1\video)	Allow: Read & Exec, List, Read	Deny:

After clicking on **Advanced**, the permissions displayed are as follows;

Deny	Everyone	Full Control	This folder, subfolders and files
Allow	Fred Costello	Special	This folder, subfolders and files
Allow	video	Read & Execute	This folder, subfolders and files
Allow	Diane Green	Read, Write & Exec	Subfolders and files
Allow	CXFS ACL Mask	Read, Write & Exec	Subfolders and files
Allow	Diane Green	Read & Exec	This folder only
Allow	CXFS ACL Mask	Read & Exec	This folder only

If an ACL entry is the same in the default ACL, a single entry is generated for the This folder, subfolders and files entry. Any entries that are different will have both Subfolders and files and This folder only entries.

Adding the first inheritable entry to a directory will cause CXFS to generate any missing ACL entries like the owner, group, and other users. The mode bits for these entries will be generated from the umask.

The process for modifying default ACLs in Windows NT differs from Windows 2000, because the Windows Explorer file permissions window in Windows NT is limited in its support of ACLs. As with some NTFS ACLs, some CXFS ACLs will cause a dialog to be displayed by Windows Explorer under Windows NT stating that this ACL can only be manipulated by a machine running Windows NT 5.0 (that is, Windows 2000). In these cases, changes to the ACL should be performed on non-Windows NT nodes.

For example, if the directory default ACL includes a user or group entry that is not included in the main ACL for that directory, in this case user `diane`, the Windows NT will be unable to display the ACL:

```
irix% ls -D dir1
dir1[u::rwx,g::r-x,o::---,m::r-x/u::rwx,g::r-x,u:diane:r-x,o::---,m::r-w]
```

Adding an entry in the non-default ACL for user `diane` will remove this limitation.

Adding different `Subfolders Only` and `Files Only` entries under Windows 2000 will result in only the first entry being used because an IRIX ACL cannot differentiate between the two.

To remove a default ACL entry under Windows NT requires that all you unset all permission flags rather than selecting `None Specified`. (The latter may result in a new entry being generated.)

Host Bus Adapter Installation for Windows

The QLogic Fibre Channel host bus adapter (HBA) should be installed according to the QLogic hardware and driver installation instructions.

Information regarding large logical unit (LUN) support under Windows can be found in the QLogic documentation and also in Microsoft's support database:

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q310072>

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q245637>

Note: LUN 0 must be present in order for Windows NT to see the other LUNs. See the QLogic driver `README` and release notes for further details.

This section discusses the following:

- "Confirming the QLogic HBA Installation", page 130
- "Support for More than 8 LUNs under Windows NT ", page 130
- "Configuring Two HBAs for Failover Operation", page 131

Confirming the QLogic HBA Installation

To confirm that the QLogic HBA and driver are correctly installed, select the following to display all of the logical units (LUNs) attached to the Fibre Channel switch:

```
Start
  > Settings
      > Control Panel
          > SCSI Adapters
              > QLogic QLA2200
                  > Bus 0
```

Support for More than 8 LUNs under Windows NT

The Qlogic HBA will require further configuration to support more than 8 LUNs under Windows NT. This is performed by using the SANsurfer software that is distributed with the QLogic hardware or that can be downloaded from the following QLogic website:

http://www.qlogic.com/support/drivers_software.asp

The installation instructions for SANsurfer are also located on the QLogic website.

Upon successful installation of the HBA hardware and software, start the SANsurfer manager application.

Connect to the local host and change the LUNs per target to 0 by selecting the following:

```
Select NVRAM section
  > Advanced NVRAM Settings
      > LUNS per Target
```

Save this configuration and then reboot when prompted.

To confirm that the QLogic HBA and driver are correctly installed after the reboot, check the SCSI registry by selecting the following:

```
Start
  > Run
      > regedit
```

Select the following key:

```
HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\Scsi\Scsi Port x\Scsi Bus 0\Target Id 0\Logical Unit Id x-xxx
```

You should see all of the LUNs available in the RAID device, where `Scsi Port x` represents the `QL2xxx` driver.

If you have fewer than 72 LUNs, you should also see the HBA and LUNs in the SCSI adapters list by selecting the following:

```
Start
  > Settings
    > Control Panel
      > SCSI Adapters
        > QLogic QLA2xxx
          > Bus 0
```

The limitation of the SCSI adapters list with a large number of LUNs is further described in the following:

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q190834>

If you are unable to see the SCSI devices in the registry, check the following, as described in "Common Host Bus Adapter Problems", page 177:

- The HBA is properly seated in the host
- Cables are connected correctly
- The node is not fenced

Configuring Two HBAs for Failover Operation

This procedure assumes that the CXFS driver is already installed and working properly with one host bus adapter (HBA).

Note: QLogic only supports failover and load balancing of two or more HBAs when all the HBAs have Fibre Channel connections to the LUNs on startup. If the connection to one of the HBAs is not present upon boot, this feature may not function correctly.

To configure two HBAs for failover operation under Windows, do the following:

1. Install the QLdirect driver v8.01.03 (NT4) by following all the default settings for the installation and verify that the CXFS client still operates normally.
2. Disable fencing for this node. You can do this using the CXFS GUI or the `cmgr(1M)` command.
3. Determine the world wide port name (WWPN) of the current adapter:
 - a. Install SANSurfer Qlogic SANblade NT Agent v2.0.15.
 - b. Install SANSurfer Qlogic SANblade Manager v2.0.15.
 - c. Run SANSurfer to determine the WWPN.
 - d. Record the WWPN on paper.
4. Shut down Windows.
5. Install the second HBA and start Windows.
6. If the second HBA is a different model from the original one, install its mini port driver (for example, `ql2300.sys`).
7. Start the Qlogic SANblade Manager and verify that two HBAs are detected. Verify that both of them mirror the same devices and logical units (LUNs). Notice that both HBAs have the same world wide node name (WWNN) but different WWPNs. The original HBA can be recognized by its WWPN recorded in step 3.
8. Make the NVRAM settings (or at least LUNs per target) of the new HBA adapter the same as the original one.
9. Configure the HBA port (click on **Configure**).

Note: Ignore the following message, which appears when HBA/LAN configuration is done for the first time (line breaks added here for readability):

```
An invalid device and LUN configuration has been detected. Auto
configure run automatically.
```

Click on **OK** to continue.

The HBA0 devices are automatically set to be visible for Windows application (notice the open eye) and HBA1 devices are set to be invisible (notice the closed eye).

10. Select the first device in the table, right click, and then select **Configure LUN(s)**. In the new window, select the following:

Tools

- > **Load Balance**
- > **All LUNs**

This will statically distribute the LANs traffic load associated with this device between the two HBAs.

Repeat this step for each of the other HBA devices.

11. Click on **Apply** to save the new configuration.
12. Add the new HBA WWPN to the `fencing.conf` file and update the switch port information using the CXFS GUI or the `cmgr` command. Enable fencing.
13. Reboot Windows.

For more information about using the CXFS GUI or the `cmgr` command to perform these tasks, see *CXFS Version 2 Software Installation and Administration Guide*.

Preinstallation Steps for Windows

When you install the CXFS software on the client-only node, you must modify certain system files. **The network configuration is critical.** Each node in the cluster must be able to communicate with every other node in the cluster by both logical name and IP address without going through any other network routing; proper name resolution is key. SGI recommends static routing.

This section provides an overview of the steps that you or a qualified Windows service representative will perform on your Windows nodes prior to installing the CXFS software. It contains the following:

- "Upgrading the QLogic BIOS", page 134
- "Hostname Resolution and Network Configuration Rules for Windows", page 135
- "Adding a Private Network for Windows Nodes", page 135

- "Adding a Private Network for Windows 2000 Nodes", page 136
- "Verifying the Private and Public Networks for Windows", page 137

Upgrading the QLogic BIOS

Note: If CXFS is already installed and running, stop the CXFS Client service and set it to manual, as described in "Manual CXFS Startup/Shutdown for Windows", page 154, and then restart the machine. You can then perform the following procedure.

If you need to upgrade the QLogic BIOS, do the following:

1. Run the QLogic SANsurfer software and connect to the machine:

```
Start
  > Programs
    > QLogic Management Suite
      > SANsurfer
        > Connect
```

If you are unable to connect to the machine, you may not have installed the QLogic NT Agent software, which is another option in the SANsurfer software installation.

2. Enable the BIOS on the HBA by selecting the following:

```
Adapter 2xxx
  > NVRAM Settings
    > Enable Host Adaptor BIOS
```

3. Update the BIOS by selecting the following:

```
Adapter 2xxx
  > Utilities
    > Update Flash
```

Select the BIOS update file.

4. Mark the CXFS Client service to automatically start.
5. Reboot the machine.

Hostname Resolution and Network Configuration Rules for Windows



Caution: It is critical that you understand these rules before attempting to configure a CXFS cluster.

The following hostname resolution rules and recommendations apply to Windows nodes:

- Hostnames cannot begin with an underscore (_) or include any whitespace characters.
- The private network IP addresses on a running node in the cluster cannot be changed while cluster services are active.
- You must be able to communicate directly between every node in the cluster (including client-only nodes) using IP addresses and logical names, without routing.
- A private network must be dedicated to be the heartbeat and control network. No other load is supported on this network.
- The heartbeat and control network must be connected to all nodes, and all nodes must be configured to use the same subnet for that network.

Adding a Private Network for Windows Nodes

The steps to add a private network are platform-specific.

Adding a Private Network for Windows NT Nodes

The following procedure provides an overview of the steps required to add a private network to the Windows NT node.

Note: A private network is **required** for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site.

1. Install the second network adapter in the Windows node as per the network adapter vendor instructions. In some cases you must remove all network setups, restart, and then add network services to each network adapter from scratch.

2. Ensure that the Windows NT node recognizes two network adapters in the system. Select the following:

Start
 > **Settings**
 > **Control Panel**
 > **Network**
 > **Adapters**

3. Specify the private network settings (IP address, subnet mask, default gateway) on one of the network adapters. Select the following:

Start
 > **Settings**
 > **Control Panel**
 > **Network**
 > **Protocols**
 > **TCP/IP Protocol**
 > **Properties**
 > **IP Address**
 > **Adapter**

The private network IP address must be a fixed address and cannot be configured by DHCP.

4. Specify DNS settings as required in the **DNS** tab.
5. Select the software components to be installed. All components should be installed.
6. Skip the **WINS Address tab** (a WINS server is not required).
7. Ensure that **IP Forwarding** is *not* selected in the **Routing** tab.
8. In the **IP Address** tab, ensure that the other network adapter is configured with a different IP address and a different subnet; this second network is the public network for all other network traffic.

The public network interface can be configured with DHCP.

Adding a Private Network for Windows 2000 Nodes

The following procedure provides an overview of the steps required to add a private network to the Windows 2000 node.

Note: A private network is **required** for use with CXFS. Only the private network is used by CXFS for heartbeat/control messages.

You may skip some steps, depending upon the starting conditions at your site.

1. Install the second network adapter in the Windows node as per the network adapter vendor instructions. In some cases you must remove all network setups, restart, and then add network services to each network adapter from scratch.
2. Ensure that the Windows 2000 node recognizes two network adapters in the system. Select the following:

Start

> **Settings**

> **Network and Dial-up Connections**

3. Specify the private network settings (IP address, subnet mask, default gateway) on one of the network adapters. Select the following:

Start

> **Settings**

> **Network and Dial-up Connections**

Then right-mouse click on **Properties** and select the following:

Internet Protocol (TCP/IP)

> **Properties**

The private network IP address must be a fixed address and cannot be configured by DHCP.

4. Specify the static IP address and DNS server.

Verifying the Private and Public Networks for Windows

You can confirm that the previous procedures to add private networks were performed correctly by using the `ipconfig` command in a DOS command shell. In

the following example, the 10 network is the private network and the 192.168.63 network is the public network on a Windows system:

```
> ipconfig /all
Windows NT IP Configuration
    Host Name . . . . . : cxfs1
    DNS Servers . . . . . :
    Node Type . . . . . : Hybrid
    NetBIOS Scope ID. . . . . :
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No
    NetBIOS Resolution Uses DNS : No

Ethernet adapter E190x1:
    Description . . . . . : 3Com EtherLink PCI
    Physical Address. . . . . : 00-01-03-46-2E-09
    DHCP Enabled. . . . . : No
    IP Address. . . . . : 10.0.0.201
    Subnet Mask . . . . . : 255.0.0.0
    Default Gateway . . . . . : 10.0.0.255

Ethernet adapter E190x2:
    Description . . . . . : 3Com EtherLink PCI
    Physical Address. . . . . : 00-B0-D0-31-22-7C
    DHCP Enabled. . . . . : No
    IP Address. . . . . : 192.168.63.201
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.63.254
    Primary WINS Server . . . . . : 192.168.63.254
```

Client Software Installation Steps for Windows

The CXFS software will be initially installed and configured by SGI personnel. This section provides an overview of those procedures. You can use the information in this section to verify the installation.

Note: This procedure assumes that the CXFS software is installed under the default path C:\Program Files\CXFS. If a different path is selected, then that path should be used in its place in the following instructions.

To install the CXFS client software on a Windows node, do the following:

1. Log onto the Windows node as `Administrator`.
2. Verify that the node has been updated to the correct service pack:
 - Windows NT 4.0 Service Pack 6:
 - Start**
 - > **Programs**
 - > **Administrative Tools**
 - > **Windows NT Diagnostics**
 - > **Version**
 - Windows 2000 Service Pack 2 or later:
 - Start**
 - > **Programs**
 - > **Accessories**
 - > **System Tools**
 - > **System Information**
3. Insert the CXFS for Windows CD-ROM into the Windows host. Normally the setup program will automatically run, otherwise run `winnt/setup.exe` from the CD-ROM.
4. Acknowledge the software license agreement when prompted. The release notes will be displayed automatically.
5. Install the CXFS software, as shown in Figure 9-1. If the software is to be installed in a nondefault directory, click on **Browse** to select another directory. Click on **Next** when finished.

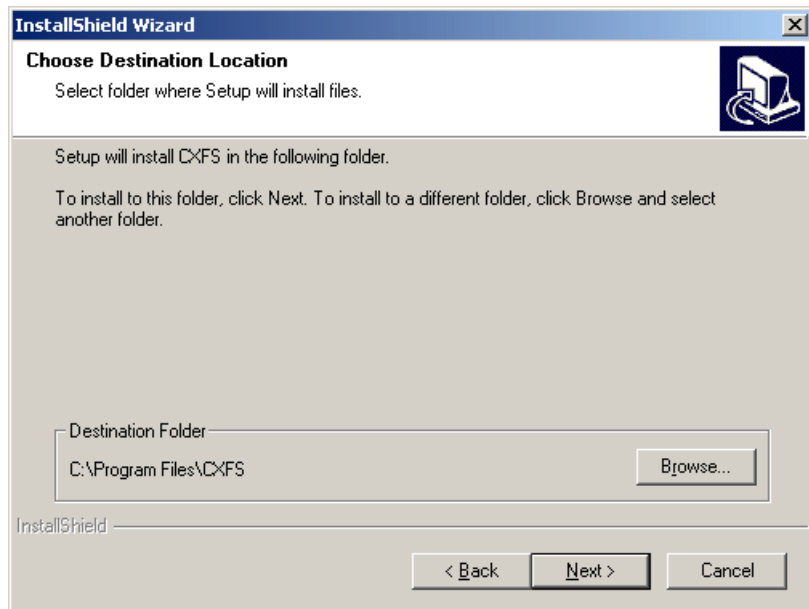


Figure 9-1 Select All Software Components

6. Enter details for the following fields as shown in Figure 9-2 and click on **Next** when finished:
 - **Drive letter for CXFS volumes to be mounted on:** specify the **drive letter** under which all CXFS filesystems will be mounted. You cannot select a drive letter that is currently in use.
 - **User ID mapping lookup sequence:** choose the appropriate primary and (optionally) secondary method. See "User Identification Mapping Methods", page 117.
 - **Location of fencing, UNIX /etc/passwd and /etc/group files:** specify the path where the configuration files will be installed and accessed by the CXFS software if required. The default is the same location as the software under C:\Program Files\CXFS.
 - **IP address of the heartbeat network adapter:** specify the IP address of the private network adapter on the Windows node.

- **Additional arguments:** enter arguments that may be passed to the CXFS Client service. For most configurations, this can be left empty. See "Modifying the CXFS Software on a Solaris System", page 107.

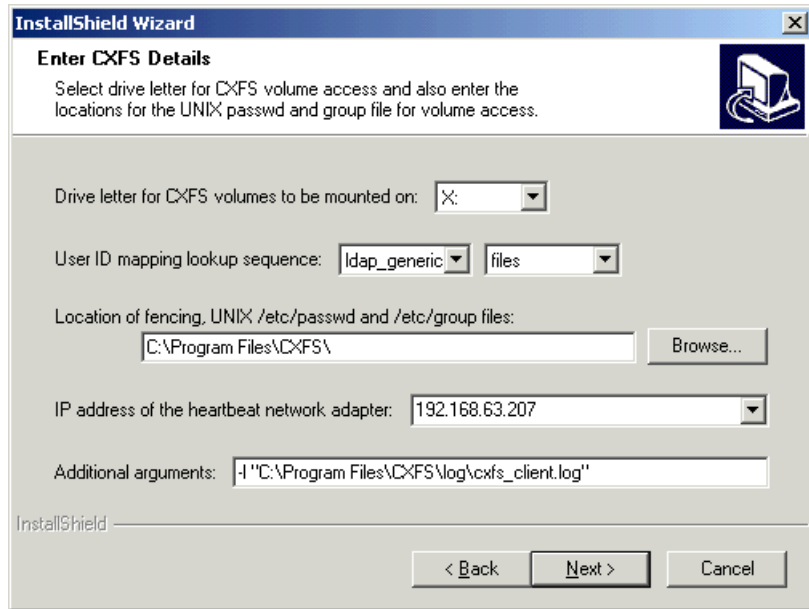


Figure 9-2 Enter CXFS Details

7. If you select **ldap_activatedir** as the user ID mapping method, the dialog in in Figure 9-3 is displayed after you click **Next**.

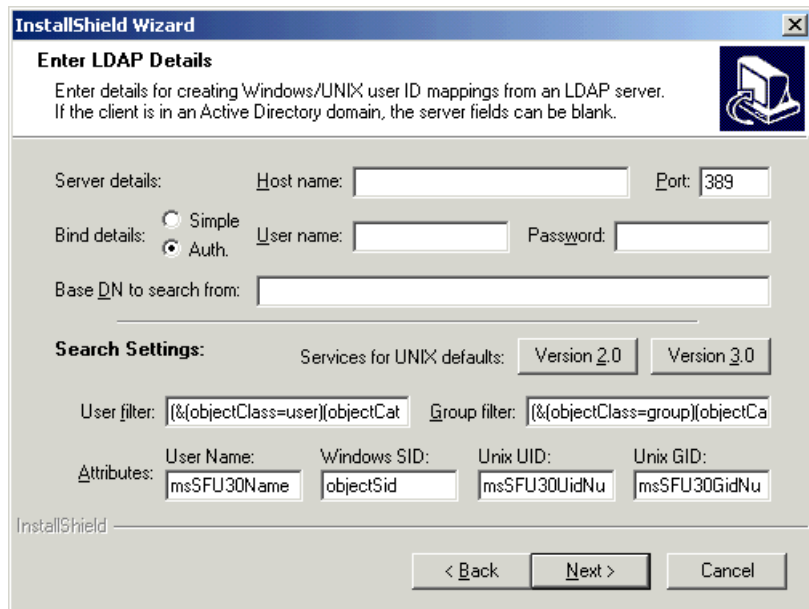


Figure 9-3 Active Directory Details

If you have a standard Active Directory configuration with Windows Services for UNIX (SFU), you need only to select the version of SFU and ensure that **Authenticated** binding is selected; doing so will then define the correct Active Directory defaults. The other server details can normally remain blank.

8. If you select **ldap_generic** as the user ID mapping method, the dialog in Figure 9-4 is displayed after you click **Next**. You must provide entries for the **Host name** and the **Base DN to search from** fields. For a standard OpenLDAP server, you can select a simple anonymous bind (default settings with the **User name** and **Password** fields left blank) and select the standard search settings by clicking **Posix**.

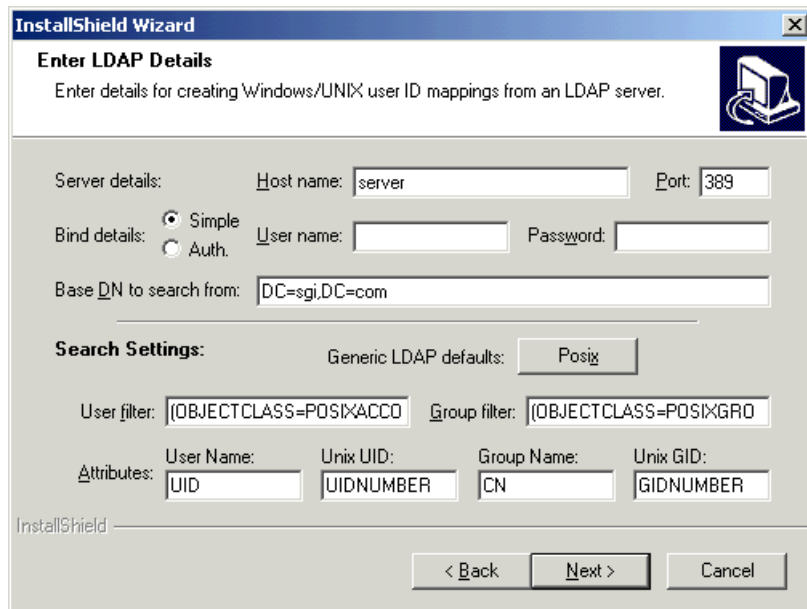


Figure 9-4 Generic LDAP Details

- Review the settings, as shown in Figure 9-5. If they appear as you intended, click on **Next**. If you need to make corrections, click on **Back**.

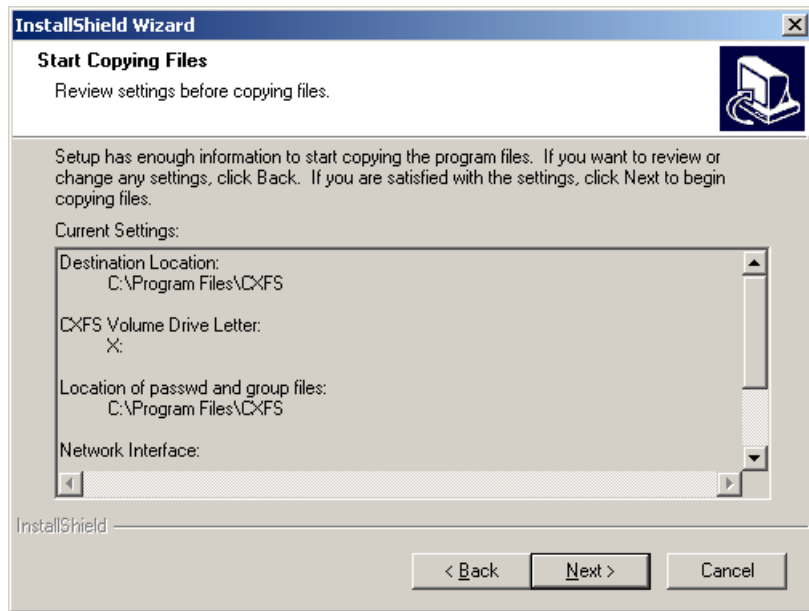


Figure 9-5 Review the Settings

After you click on **Next**, the CXFS software will be installed.

10. You will be given the option to start the driver at system start-up or now, as show in Figure 9-6.

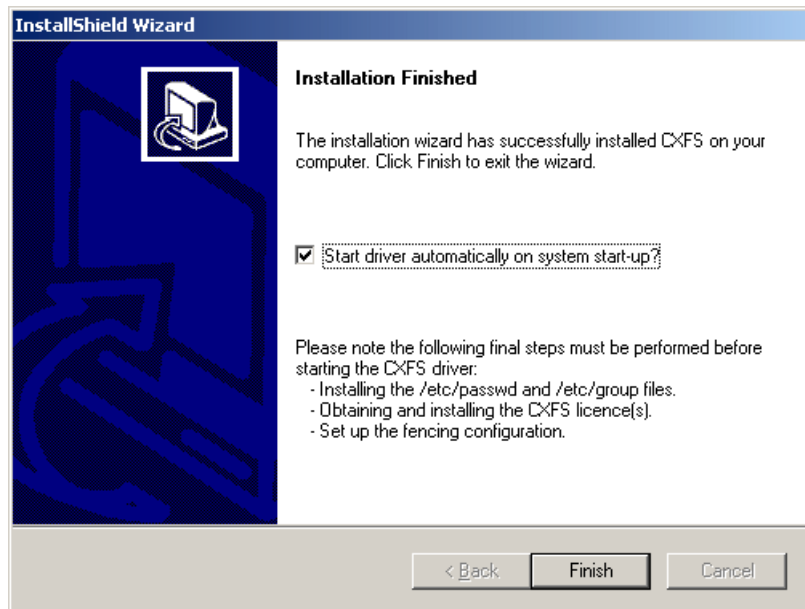


Figure 9-6 Start CXFS Driver

Because there are some important postinstallation steps, do not start the CXFS driver now. Click **Start driver automatically on system start-up** and click on **Finish**.

Postinstallation Steps for Windows

This section discusses the configuration steps that you should perform after installing CXFS software but before restarting a Windows node.

The following postinstallation steps are required to ensure the correct operation of the CXFS software:

- "Configuring the FLEXlm License for Windows"
- "Creating the Windows I/O Fencing File", page 147
- "Performing User Configuration", page 149

- "Checking Permissions on the Password, Group, and Fencing Files", page 150
- "Creating a New Hardware Profile", page 151

Configuring the FLEXlm License for Windows

Note: Windows NT licenses cannot be used under Windows 2000 and vice versa. If you are upgrading a Windows NT node to Windows 2000, you must obtain a new license.

You must configure a FLEXlm license before you restart the Windows node by following these steps:

1. Add the mandatory CXFS license and the optional XVM license to the following file:

```
C:\Program Files\CXFS\lib\license.dat
```

For more information, see Chapter 4, "Obtaining CXFS and XVM FLEXlm Licenses", page 25.

2. Validate these licenses by running the `cxfslicense` command in a DOS command shell.

Create a DOS command shell with the following sequence:

- Windows NT:

```
Start
  > Programs
    > Command Prompt
```

- Windows 2000:

Start
 > Programs
 > Accessories
 > Command Prompt

To run `cxfslicense`, enter the following command:

```
C:\Program Files\CXFS\cxfslicense.exe
```

If a valid license has been correctly specified, the following will be displayed:

```
Found valid license for feature CXFS_NT version 2.000  
The CPU count specified in the license is OK.
```

If the Windows node has the optional XVM mirroring license, you will also see the following:

```
Found valid license for feature XVM_NT version 3.000  
The CPU count specified in the license is OK.
```

Note: Licenses for Windows 2000 have the feature names `CXFS_W2K` and `XVM_W2K`.

Creating the Windows I/O Fencing File

I/O fencing is required to protect data integrity for Windows nodes.

On Windows, the `cxfs_client` software detects the world wide port names (WWPNs) of any supported host bus adapters (HBAs) in the system that are connected to a switch that is configured in the cluster database. These HBAs will then be available for fencing.

If automatic HBA WWPN detection fails according to the CXFS Client service log located at `C:\Program Files\CXFS\cxfs_client.log`, then the file `C:\Program Files\CXFS\fencing.conf` can be used to specify the WWPN(s). If used, this file must contain a simple list of WWPNs, one per line. To determine the WWPNs, see the documentation for the driver in use.

If you use the `/etc/fencing.conf` file, you must update it whenever the HBA configuration changes, including the replacement of an HBA.

The fencing file enumerates the worldwide port name for all of the QLogic host bus adapters (HBA) that will be used to mount a CXFS filesystem. The path to the I/O fencing file is as follows:

```
C:\Program Files\CXFS\fencing.conf
```

The I/O fencing file must contain a line for the QLogic HBA worldwide port name (WWPN) as a 64-bit hexadecimal number. You must update the fencing file whenever the QLogic HBA configuration changes, including the replacement of a QLogic HBA.

To determine the HBA WWPN, do the following:

1. Set up the Brocade Fibre Channel switch and QLogic HBA according to the directions in Chapter 3, "Brocade Fibre Channel Switch Verification", page 19, and "Host Bus Adapter Installation for Windows", page 129.
2. Follow the Fibre Channel cable on the back of the Windows host to determine the port to which it is connected in the Brocade Fibre Channel switch. Ports are numbered beginning with 0. (For example, if there are 8 ports, they will be numbered 0 through 7.)
3. Use the `telnet(1)` command to connect to the Brocade Fibre Channel switch and log in as user `admin` (the password is `password` by default).
4. Execute the `switchshow` command to display the switches and their WWPN numbers. For example:

```
brocade04:admin> switchshow
switchName:      brocade04
switchType:      2.4
switchState:     Online
switchRole:      Principal
switchDomain:    6
switchId:        fffc06
switchWwn:       10:00:00:60:69:12:11:9e
switchBeacon:    OFF
port  0: sw  Online      F-Port  20:00:00:01:73:00:2c:0b
port  1: cu  Online      F-Port  21:00:00:e0:8b:02:36:49
port  2: cu  Online      F-Port  21:00:00:e0:8b:02:12:49
port  3: sw  Online      F-Port  20:00:00:01:73:00:2d:3e
port  4: cu  Online      F-Port  21:00:00:e0:8b:02:18:96
port  5: cu  Online      F-Port  21:00:00:e0:8b:00:90:8e
port  6: sw  Online      F-Port  20:00:00:01:73:00:3b:5f
port  7: sw  Online      F-Port  20:00:00:01:73:00:33:76
```

```

port 8: sw Online      F-Port  21:00:00:e0:8b:01:d2:57
port 9: sw Online      F-Port  21:00:00:e0:8b:01:0c:57
port 10: sw Online     F-Port  20:08:00:a0:b8:0c:13:c9
port 11: sw Online     F-Port  20:0a:00:a0:b8:0c:04:5a
port 12: sw Online     F-Port  20:0c:00:a0:b8:0c:24:76
port 13: sw Online     L-Port  1 public
port 14: sw No_Light
port 15: cu Online     F-Port  21:00:00:e0:8b:00:42:d8

```

The WWPN is the hexadecimal string to the right of the port number. For example, the WWPN for port 0 is 2000000173002c0b (you must remove the colons from the WWPN reported in the `switchshow` output to produce the string to be used in the `C:\Program Files\CXFS\fencing.conf` file).

5. Edit the following file to add the WWPN for the port determined in step 2 (comment lines begin with #):

```
C:\Program Files\CXFS\fencing.conf
```

For example, if you determined that port 0 is the port connected to the Brocade Fibre Channel switch, the fencing file should appear as follows:

```

C\> type C:\Program Files\CXFS\fencing.conf
#
# WWPN of the JNI HBA installed on this system
#
2000000173002c0b

```

6. After the Windows node is added to the cluster (see Chapter 10, "Cluster Configuration", page 159), enable the fencing feature by using the CXFS GUI or `cmgr` command on an IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Performing User Configuration

If the user mapping is not correctly configured, all filesystem operations will be as user `nobody`.

If you selected the **passwd and group files** user ID mapping method, you must install the `passwd` and `group` files. The default `passwd` and `group` files that are installed are invalid files containing comments; these invalid files will cause the CXFS Client to generate warnings in its log file and users may not be correctly configured. You must remove the comments in these files when you install the `passwd` and `group` files.

After installing the CXFS software onto the Windows node, but before the CXFS node is restarted, you must install the `/etc/passwd` and `/etc/group` files from an IRIX node to the location on the Windows node specified during installation, which defaults to the following:

- `/etc/passwd` as `C:\Program Files\CXFS\passwd`
- `/etc/group` as `C:\Program Files\CXFS\group`

If you selected the **Active Directory** method, you must specify the UNIX identifiers for all users of the CXFS node. On the domain controller, run the following to specify the UNIX UID and GID of a given user:

```
Start
  > Program Files
      > Administrative Tools
          > Active Directory Users and Computers
              > Users
```

Select a user and then select:

```
Properties
  > UNIX Attributes
```

The CXFS software will check for changes to the LDAP database every 30 seconds.

After the CXFS software has started, you can use the `cxfs_info` command to confirm the user configuration, regardless of the user ID mapping method chosen. See "User Identification on a Windows Node", page 116.

Checking Permissions on the Password, Group, and Fencing Files

The permissions on the `fencing.conf`, `passwd`, and `group` files must restrict access so that only the system administrator can modify these files. This can be done by right-clicking on the file names in Windows Explorer and selecting the following:

- Windows NT:

Properties
 > **Security**
 > **Permissions**

- Windows 2000:

Properties
 > **Security**

Verify that the permissions are Read for Everyone and Full Control for Administrators.



Caution: Failure to set permissions on the `passwd` and `group` files would allow users to change their UID/GUI at will and even gain superuser access to the files on the CXFS filesystem. Failure to set permissions on the `fencing.conf` file would allow users to misconfigure and even disable I/O fencing, which could result in data corruption on the CXFS filesystems.

Creating a New Hardware Profile

It is strongly recommended that you create a new hardware profile and that you disable the CXFS software in the current hardware profile, in order to have a backup profile available. If the CXFS software causes the host to crash on startup, you can easily switch back to the original hardware profile and successfully return to the configuration before the CXFS software was installed.

To create a new hardware profile, right-click the **My Computer** icon and select the following:

- Windows NT:

Properties
 > **Hardware Profiles**
 > **Copy**

- Windows 2000:

Properties
 > **Hardware**
 > **Hardware Profiles**
 > **Copy**

This action copies the current hardware profile, most likely called the **Original Configuration (Windows NT)** or **Profile 1 (Windows 2000)**. You should call this new profile **CXFS Configuration** to distinguish it from other profiles. You can make the **CXFS Configuration** the default profile chosen on startup by selecting the up arrow button and moving the **CXFS Configuration** profile to the top of the list.

To remove the CXFS driver from the current hardware profile, which should be the original profile, select the following:

- Windows NT:

Start
 > **Settings**
 > **Control Panel**
 > **Devices**
 > **CXFS**
 > **HW Profile**
 > **Disable**

- Windows 2000:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Computer Management**
 > **System Tools**
 > **Device Manager**

To show non-plug-and-play devices, select the following:

View
 > **Show hidden devices**

To show the CXFS driver, select the following:

Non-Plug and Play Devices
 > **CXFS**
 > **Properties**
 > **Device Usage**
 > **Do not use this device in the current hardware profile**

You should also disable the CXFS Client service for the current profile by selecting the following:

- Windows NT:

Start
 > **Settings**
 > **Control Panel**
 > **Services**
 > **CXFS Client**
 > **HW Profile**
 > **Disable**

- Windows 2000:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Services**
 > **CXFS Client**
 > **Properties**
 > **Log On**
 > **Disable**

When the Windows host boots, you may choose **CXFS Configuration** to automatically start CXFS or choose the previous profile (most likely **Original Configuration**) to start without CXFS.

Manual CXFS Startup/Shutdown for Windows

The CXFS processes are automatically started when a Windows node is restarted. This behavior may be altered by changing the configuration of the CXFS filesystem driver and the CXFS Client service.

By default, the driver is configured to start manually and the Client service is configured to start automatically. Because the CXFS Client service depends on the CXFS filesystem driver, the driver will be started by the service.

It is recommended that the CXFS driver configuration remains manual.

You can change the CXFS Client service configuration to start manually, so that CXFS does not automatically start, by selecting the following:

- Windows NT:

Start
 > **Settings**
 > **Control Panel**
 > **Services**

- Windows 2000:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Services**

Change **CXFS Client** to manual rather than automatic. CXFS can then be started and stopped manually by the Administrator using the same selection sequence.

Software Maintenance for Windows

This section contains the following:

- "Modifying the CXFS for Windows Software", page 155
- "Upgrading the CXFS Software on a Windows System", page 156
- "Removing the CXFS Software from a Windows System", page 157

- "Downgrading the CXFS Software on a Windows System", page 157

Modifying the CXFS for Windows Software

To change the location of the software and other configuration settings that were requested in "Client Software Installation Steps for Windows", page 138, perform the following steps:

1. Select the following:

```
Start
  > Settings
    > Control Panel
      > Add/Remove Programs
        > CXFS
          > Add/Remove
            > Modify
```

Figure 9-7, page 156, shows the screen that lets you modify the software.

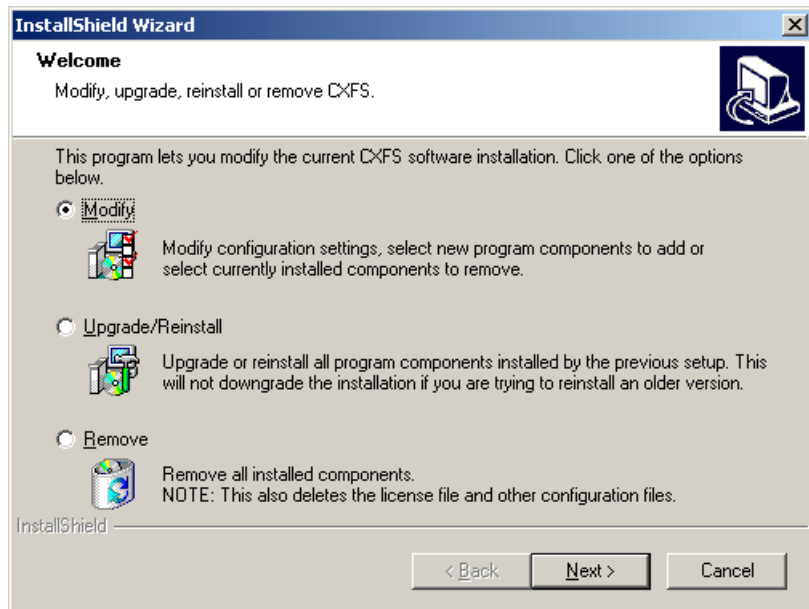


Figure 9-7 Modify the CXFS for Windows

2. Make the necessary configuration changes.

You can display the list of possible command line arguments supported by the CXFS Client service by running the service from a DOS command shell as follows:

```
> C:\Winnt\system32\cxfs_client.exe -h
```

3. Restart the Windows node, which causes the changes to take effect.

Upgrading the CXFS Software on a Windows System

To upgrade the CXFS for Windows software, perform the following steps:

1. Insert the CD-ROM containing the upgraded software to run the setup program. If the setup program does not automatically start, run `winnt/setup.exe` from the CD-ROM.
2. Select **Upgrade/Reinstall** and follow the prompts as discussed in "Client Software Installation Steps for Windows", page 138.

Figure 9-7, page 156, shows the screen that lets you modify the software.

3. Restart the Windows node. The upgraded software will not activate until the Windows node is restarted.

Removing the CXFS Software from a Windows System

To remove the CXFS for Windows software, first ensure that no applications on this host are accessing files on a CXFS filesystem. Then, select the following sequence to remove all installed files and registry entries:

```
Start
  > Settings
    > Control Panel
      > Add/Remove Programs
        > CXFS
          > Add/Remove
            > Remove
```

Figure 9-7, page 156, shows the screen that lets you remove the software.

Note: The `passwd`, `group`, and `fencing.conf` files will be removed.

You should then restart the Windows node. This will cause the changes to take effect.

Downgrading the CXFS Software on a Windows System

To downgrade the CXFS software, follow the instructions to remove the software in "Removing the CXFS Software from a Windows System", page 157, and then install the older version of the software as directed in "Client Software Installation Steps for Windows", page 138.

Note: The removal process may remove the configuration and license files. You should back up these files before removing the CXFS software so that you can easily restore them after installing the downgrade.

Cluster Configuration

This chapter provides an overview of the procedures to add the client-only nodes to an established IRIX cluster. It assumes that you already have an IRIX cluster installed and running with mounted filesystems. These procedures will be performed by you or by SGI service personnel.

All CXFS administrative tasks other than restarting the Windows node must be performed on an IRIX node, using either the CXFS GUI (invoked by the `cxfsmgr` command) or the `cmgr` command. The GUI also provides a guided configuration for defining a cluster.

This section discusses the following tasks in cluster configuration:

- "Defining the Client-Only Nodes"
- "Adding the Client-Only Nodes to the Cluster", page 161
- "Defining the Switch for I/O Fencing", page 162
- "Starting CXFS Services on the Client-Only Nodes", page 164
- "Verifying LUN Masking", page 164
- "Mounting Filesystems on the Client-Only Nodes", page 164
- "Restarting the Windows Node", page 165
- "Verifying the Cluster", page 165
- "Forced Unmount of CXFS Filesystems", page 167

For detailed configuration instructions, see the *CXFS Version 2 Software Installation and Administration Guide*.

Defining the Client-Only Nodes

To add a client-only node to a CXFS cluster, you must define it as a node in the pool. You can do this on an IRIX node using the CXFS GUI or `cmgr` command.

Do the following to determine the value for the hostname field in the GUI:

- AIX: use the value displayed by `/usr/bin/hostname`, which must match the node's primary hostname in the `/etc/hosts` file; that is, the first field after the node's IP address in `/etc/hosts`. This field can be either the hostname or the fully qualified name.
- Linux on IA32 platforms and SGI ProPack: use the value displayed by `/bin/hostname`.
- Solaris: use the value displayed by `/etc/nodename`, which must match the node's primary hostname in the `/etc/inet/hosts` (or `/etc/hosts`) file; that is, the first field after the node's IP address in `/etc/inet/hosts` (or `/etc/hosts`). This field can be either the hostname or the fully qualified domain name.

- Windows NT:

Start

```
> Settings
  > Control Panel
    > Network
      > Identification
        > Computer Name
```

- Windows 2000:

Start

```
> Settings
  > Network and Dial-up Connections
    > Advanced
      > Network Identification
```

When you specify that a node is running an operating system other than IRIX or Linux, the node will automatically be defined as a client-only node and you cannot change it. (These nodes cannot be potential meta-data servers and are not counted when calculating the CXFS kernel membership quorum.) For client-only nodes, you must specify a unique node ID.

For example, the following shows the entries used to define a Solaris node named `solaris1` using the `cmgr` command in prompting mode:

```
# /usr/cluster/bin/cmgr -p
Welcome to SGI Cluster Manager Command-Line Interface

cmgr> define node solaris1
```


Enter commands, you may enter "done" or "cancel" at any time to exit

```

Hostname[optional] ?
Is this a FailSafe node <true|false> ? false
Is this a CXFS node <true|false> ? true
Operating System <IRIX|Solaris|Windows> ? solaris
Node ID ? 7
Do you wish to define failure hierarchy[y/n]:y
Hierarchy option 0 <System|Fence|Shutdown>[optional] ? fence
Hierarchy option 1 <System|Fence|Shutdown>[optional] ? shutdown
Hierarchy option 2 <System|Fence|Shutdown>[optional] ?
Number of Network Interfaces ? (1)
NIC 1 - IP Address ? 163.154.18.172
NIC 1 - Heartbeat HB (use network for heartbeats) <true|false> ? true
NIC 1 - (use network for control messages)
NIC 1 - (use network for control messages) true|false> ? true
NIC 1 - Priority <1,2,...> ? 1

```

For details about these commands, see the “Define a Node” sections of the GUI or `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Adding the Client-Only Nodes to the Cluster

After you define all of the client-only nodes, you must add them to the cluster using either the CXFS GUI or the `cmgr` command on an IRIX node.

For example, if you have already defined a cluster named `cxfscluster` and want to add the Solaris nodes `solaris1` and `solaris2`, you could use the following `cmgr` command:

```

cmgr> modify cluster cxfscluster

cxfscluster ? add node solaris1
cxfscluster ? add node solaris2
cxfscluster ? done

```

For details, see the “Modify a Cluster” sections of the GUI or `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Depending upon your filesystem configuration, you may also need to add the node to the list of clients that have access to the volume. See "Mounting Filesystems on the Client-Only Nodes", page 164.

Defining the Switch for I/O Fencing

You are required to use I/O fencing on client-only nodes in order to protect data integrity. I/O fencing requires a Brocade Fibre Channel switch. To define the switch for the cluster database, use either the CXFS GUI or the `cmgr` command on an IRIX node.

For example:

```
cmgr> define switch ptg-brocade username admin password password
```

Note: The masking feature applies only to IRIX nodes. (IRIX nodes automatically discover the available HBAs, and the masking feature is used to restrict the HBAs that are eligible for fencing.) For nodes running other supported operating systems, the fencing file determines the ports that may be fenced. For more information, see the following:

- "Postinstallation Steps for AIX: Creating the I/O Fencing File", page 43
 - "Postinstallation Steps for Linux on IA32 Platforms: Creating the I/O Fencing File", page 59
 - "Postinstallation Steps for Solaris: Creating the I/O Fencing File", page 104
 - "Creating the Windows I/O Fencing File", page 147
-

After you have defined the switch, you must ensure that all of the Brocade ports that are connected to the cluster nodes are enabled. To determine port status, enter the following on a CXFS administration node:

```
irix# hafence -v
```

If there are disabled ports that are connected to cluster nodes, you must enable them. Log into the switch as user `admin` and use the following command:

```
switch# portEnable portnumber
```

You must then update the switch port information using the GUI or `cmgr`.

For example, suppose that you have a cluster with port 0 connected to the node `blue`, port 1 connected to the node `green`, and port 5 connected to the node `yellow`,

all of which are defined in cluster `colors`. The following output shows that the status of port 0 and port 1 is `disabled` and that the host is `UNKNOWN` (as opposed to port 5, which has a status of `enabled` and a host of `yellow`). Ports 2, 3, 4, 6, and 7 are not connected to nodes in the cluster and therefore their status does not matter.

```
irix# hafence -v
Switch[0] "ptg-brocade" has 8 ports
Port 0 type=FABRIC status=disabled hba=0000000000000000 on host UNKNOWN
Port 1 type=FABRIC status=disabled hba=0000000000000000 on host UNKNOWN
Port 2 type=FABRIC status=enabled hba=210000e08b05fecf on host UNKNOWN
Port 3 type=FABRIC status=enabled hba=210000e08b01fec5 on host UNKNOWN
Port 4 type=FABRIC status=enabled hba=210000e08b01fec3 on host UNKNOWN
Port 5 type=FABRIC status=enabled hba=210000e08b019ef0 on host yellow
Port 6 type=FABRIC status=enabled hba=210000e08b0113ce on host UNKNOWN
Port 7 type=FABRIC status=enabled hba=210000e08b027795 on host UNKNOWN
```

In this case, you could need to enable ports 0 and 1.

Logged in to the switch:

```
switch# portEnable 0
switch# portEnable 1
```

Logged in to a CXFS administration node:

```
irix# hafence -v
Switch[0] "ptg-brocade" has 8 ports
Port 0 type=FABRIC status=disabled hba=210000e08b0103b8 on host UNKNOWN
Port 1 type=FABRIC status=disabled hba=210000e08b0102c6 on host UNKNOWN
Port 2 type=FABRIC status=enabled hba=210000e08b05fecf on host UNKNOWN
Port 3 type=FABRIC status=enabled hba=210000e08b01fec5 on host UNKNOWN
Port 4 type=FABRIC status=enabled hba=210000e08b01fec3 on host UNKNOWN
Port 5 type=FABRIC status=enabled hba=210000e08b019ef0 on host yellow
Port 6 type=FABRIC status=enabled hba=210000e08b0113ce on host UNKNOWN
Port 7 type=FABRIC status=enabled hba=210000e08b027795 on host UNKNOWN
```

```
irix# cmgr -c admin fence update
```

```
irix# hafence -v
Switch[0] "ptg-brocade" has 8 ports
Port 0 type=FABRIC status=disabled hba=210000e08b0103b8 on host blue
Port 1 type=FABRIC status=disabled hba=210000e08b0102c6 on host green
Port 2 type=FABRIC status=enabled hba=210000e08b05fecf on host UNKNOWN
Port 3 type=FABRIC status=enabled hba=210000e08b01fec5 on host UNKNOWN
```

```
Port 4 type=FABRIC status=enabled hba=210000e08b01fec3 on host UNKNOWN
Port 5 type=FABRIC status=enabled hba=210000e08b019ef0 on host yellow
Port 6 type=FABRIC status=enabled hba=210000e08b0113ce on host UNKNOWN
Port 7 type=FABRIC status=enabled hba=210000e08b027795 on host UNKNOWN
```

For details, see the “Define a Switch” and “Update Switch Port Information” sections of the GUI or `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Starting CXFS Services on the Client-Only Nodes

After adding the client-only nodes to the cluster, you must start CXFS services on them. You can do this using either the CXFS GUI or the `cmgr` command on an IRIX node.

For example:

```
cmgr> start cx_services on node solaris1 for cluster cxfscluster
cmgr> start cx_services on node solaris2 for cluster cxfscluster
```

For details, see the “Start CXFS Services” sections of the GUI or `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Verifying LUN Masking

You should verify that the HBA has LUN masking configured such that the LUNs are visible to all the machines in the cluster after you connect the HBA to the Brocade Fibre Channel switch and before configuring the filesystems with XVM. For more information, see the RAID documentation.

Mounting Filesystems on the Client-Only Nodes

If you have specified that the filesystems are to be automatically mounted on any newly added nodes, then you do not need to specifically mount the filesystems on the new client-only nodes that you added to the cluster.

Otherwise, you can mount the filesystems on the new client-only nodes by unmounting the currently active filesystems, enabling the mount on the required

nodes, and then performing the actual mount. You can do this using the GUI or the `cmgr` command on an IRIX node.

For example, to mount the `fs1` filesystem on all nodes in the cluster except `solaris2`, you could use the following commands:

```
cmgr> admin cxfs_unmount cxfs_filesystem fs1 in cluster cxfscluster
cmgr> modify cxfs_filesystem fs1 in cluster cxfscluster

cxfs_filesystem fs1 ? set dflt_local_status to enabled
cxfs_filesystem fs1 ? add disabled_node solaris2
cxfs_filesystem fs1 ? done
```

Note: SGI recommends that you enable the *forced unmount* feature for CXFS filesystems, which is turned off by default; see "Recommendations", page 8, and "Forced Unmount of CXFS Filesystems", page 167.

For details, see the "Define a Filesystem" and "Mount a Filesystem" sections of the GUI or the `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Restarting the Windows Node

After completing the steps in "Postinstallation Steps for Windows", page 145, and this chapter, you should restart the Windows node. This will automatically start the driver and the Client service.

When you log into the node after restarting it, Windows Explorer will list the CXFS drive letter, which will contain the CXFS filesystems configured for this node.

Verifying the Cluster

To verify that the client-only nodes have been properly added to the cluster and that filesystems have been mounted, use the view area of the CXFS GUI, the `clconf_info` command, and the `cluster_status` command on an IRIX node.

For example:

```
irix# /var/cluster/cmgr-scripts/cluster_status

+ Cluster=cxfscluster  FailSafe=Not Configured CXFS=ACTIVE          15:15:33
  Nodes =   cxfs6     cxfs7     cxfs8     solaris1     solaris2
FailSafe =
  CXFS =      UP      UP      UP      UP      UP

CXFS      DevName      MountPoint      MetaServer      Status
fs1       /dev/cxvm/fs1    /fs1            cxfs7           UP
fs2       /dev/cxvm/fs2    /fs2            cxfs6           UP
```

On client-only nodes, the `cxfs_info` command serves a similar purpose. The command path is as follows:

- AIX, Linux on IA32 platforms, SGI ProPack, and Solaris:
/usr/cxfs_cluster/bin/cxfs_info
- Windows: C:\program files\CXFS\cxfs_info.exe

On AIX, Linux on IA32 platforms, SGI ProPack, and Solaris nodes, you can use the `-e` option to wait for events, which keeps the command running until you kill the process and the `-c` option to clear the screen between updates.

On Windows nodes, these options are enabled by default, and the window will stay up until you close it. To disable these options on Windows and get the standard UNIX behavior, use the `-D` option.

For example, on a Solaris node:

```
solaris# /usr/cxfs_cluster/bin/cxfs_info

cxfs_client status [timestamp Jul 19 13:30:22 / generation 21604]

Cluster:
  zagato (1) - enabled
Local:
  thump (2) - enabled, state: stable, cms: up, xvm: up, fs: up
Nodes:
  leesa      enabled  up      0
  thump      enabled  up      2
  thunderbox enabled  up      1
Filesystems:
```

bigstripe0	enabled	mounted	bigstripe0	/mnt/bigstripe0
concat0	enabled	mounted	concat0	/mnt/concat0
mirror0	enabled	mounted	mirror0	/mnt/mirror0
r0lun0s0	enabled	mounted	r0lun0s0	/mnt/cxfs0
r0lun0s1	enabled	mounted	r0lun0s1	/mnt/cxfs1
r0lun0s2	enabled	mounted	r0lun0s2	/mnt/cxfs2
stripe0	enabled	mounted	stripe0	/mnt/stripe0

Forced Unmount of CXFS Filesystems

Normally, an unmount operation will fail if any process has an open file on the filesystem. However, a *forced unmount* allows the unmount to proceed regardless of whether the filesystem is still in use. To enable forced unmount, define or modify the filesystem to unmount with force and then unmount the filesystem, using the following `cmgr` commands:

```
define cxfs_filesystem logical_filesystem_name [in cluster clustername]
    set force to true

modify cxfs_filesystem logical_filesystem_name [in cluster clustername]
    set force to true

admin cxfs_unmount cxfs_filesystem filesystemname [on node nodename] [in cluster clustername]
```

For example, the following set of commands modifies the `fs1` filesystem to allow forced unmount, then unmounts the filesystem on all nodes in the `cxfscluster` cluster:

```
cmgr> modify cxfs_filesystem fs1 in cluster cxfscluster
Enter commands, when finished enter either "done" or "cancel"cmgr>

cxfs_filesystem fs1 ? set force to true
cxfs_filesystem fs1 ? done
Successfully defined cxfs_filesystem fs1

cmgr> admin cxfs_unmount cxfs_filesystem fs1 in cluster cxfscluster
```

For details, see the “CXFS Filesystems Tasks with the GUI” sections of the GUI or the `cmgr` reference chapters in the *CXFS Version 2 Software Installation and Administration Guide*.

Troubleshooting

This chapter contains the following:

- "Identifying Problems on AIX Nodes"
- "Identifying Problems on Linux on IA32 Nodes", page 170
- "Identifying Problems on SGI ProPack Nodes", page 171
- "Identifying Problems on Solaris Nodes", page 171
- "Identifying Problems on Windows Nodes", page 172
- "Common Problems and Solutions", page 177
- "Reporting Problems", page 180

Identifying Problems on AIX Nodes

This section will help you identify problems with AIX client-only nodes.

Is the AIX Node in the Cluster?

To determine if the node is in the cluster, use the `cluster_status` command on an IRIX node, connect to the CXFS GUI on an IRIX node, or use the `cxfs_info` command on the AIX node. See "Verifying the Cluster", page 165.

Are there Error Messages for the AIX Node?

Look at the `/var/tmp/cxfs_client` log to see if there are any error or warning messages. These include any messages containing the words `ERROR` or `Warning`. Specific cases where error messages will occur include the following:

- The fencing file was not found, therefore the fencing configuration will not be updated on the server. For example:

```
cxfs_client: cis_get_hba_wwns warning: fencing configuration file "/etc/fencing.conf" not found
```

- A filesystem mount has failed and will be retried. For example:

```
cxfs_client: op_failed ERROR : Mount failed for aixdisk0s0
```

Identifying Problems on Linux on IA32 Nodes

This section will help you identify problems with Linux client-only nodes.

Is the Linux Node in the Cluster?

To determine if the node is in the cluster, use the `cluster_status` command on an IRIX node, connect to the CXFS GUI on an IRIX node, or use the `cxfs_info` command on the Linux node. See "Verifying the Cluster", page 165.

Are there Error Messages for the Linux Node?

Look at the `/var/log/cxfs_client` log to see if there are any error or warning messages. These include any messages containing the words `ERROR` or `Warning`. Specific cases in which error messages will occur include the following:

- The fencing file was not found, therefore the fencing configuration will not be updated on the server. For example:

```
cxfs_client: cis_get_hba_wwns warning: fencing configuration file "fencing.conf" not found
```

- A filesystem mount has failed and will be retried. For example:

```
cxfs_client:op_failed ERROR: Mount failed for concat0
```

For more information about these files, see "Log Files on Solaris", page 79. Also see the log files on the IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Identifying Problems on SGI ProPack Nodes

This section will help you identify problems with SGI ProPack client-only nodes.

Is the SGI ProPack Node in the Cluster?

To determine if the node is in the cluster, use the `cluster_status` command on an IRIX node, connect to the CXFS GUI on an IRIX node, or use the `cxfs_info` command on the SGI ProPack node. See "Verifying the Cluster", page 165.

Are there Error Messages for the SGI ProPack Node?

Look at the `/var/log/cxfs_client` log to see if there are any error or warning messages. These include any messages containing the words `ERROR` or `Warning`. Specific cases in which error messages will occur include the following:

- The fencing file was not found, therefore the fencing configuration will not be updated on the server. For example:

```
cxfs_client: cis_get_hba_wwns warning: fencing configuration file "fencing.conf" not found
```

- A filesystem mount has failed and will be retried. For example:

```
cxfs_client:op_failed ERROR: Mount failed for concat0
```

For more information about these files, see "Log Files on Solaris", page 79. Also see the log files on the IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Identifying Problems on Solaris Nodes

This section will help you identify problems with Solaris client-only nodes.

Is the Solaris Node in the Cluster?

To determine if the node is in the cluster, use the `cluster_status` command on an IRIX node, connect to the CXFS GUI on an IRIX node, or use the `cxfs_info` command on the Solaris node. See "Verifying the Cluster", page 165.

Are there Error Messages for the Solaris Node?

Look at the `/var/log/cxfs_client` log to see if there are any error or warning messages. These include any messages containing the words `ERROR` or `Warning`. Specific cases in which error messages will occur include the following:

- The fencing file was not found, therefore the fencing configuration will not be updated on the server. For example:

```
cxfs_client: cis_get_hba_wwns warning: fencing configuration file "fencing.conf" not found
```

- A filesystem mount has failed and will be retried. For example:

```
cxfs_client:op_failed ERROR: Mount failed for concat0
```

For more information about these files, see "Log Files on Solaris", page 79. Also see the log files on the IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

Identifying Problems on Windows Nodes

The following sections will help you identify problems with Windows client-only nodes.

Is the CXFS Software Running Correctly on the Windows Node?

The methods used to verify that the CXFS software is running correctly varies by Windows platform.

Windows NT CXFS Software Verification

To verify that the CXFS software is running correctly on a Windows NT node, do the following:

- Verify that the CXFS driver has started by selecting the following:

```
Start
  > Settings
    > Control Panel
      > Devices
```

- Verify that the CXFS Client service has started by selecting the following:

Start
 > **Settings**
 > **Control Panel**
 > **Services**

Windows 2000 CXFS Software Verification

To verify that the CXFS software is running correctly on a Windows 2000 node, do the following:

- Verify that the CXFS driver has started by selecting the following:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Computer Management**
 > **System Tools**
 > **Device Manager**

To show non-plug-and-play devices, select the following:

View
 > **Show hidden devices**

To show the CXFS driver, select the following:

Non-Plug and Play Devices
 > **CXFS**
 > **Properties**

- Verify that the CXFS Client service has started by selecting the following:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Services**

Is the Windows Node in the Cluster?

To determine if the Windows node is in the cluster, use the `cluster_status` command or the CXFS GUI on an IRIX node, and the `cxfs_info` command on the Windows node. See "Verifying the Cluster", page 165.

Are There Error Messages for the Windows Node?

Look in the following file to see if there are any error or warning messages:

`C:\Program Files\CXFS\log\cxfs_client.log`

You can also view the **System Event** log by selecting the following:

- Windows NT:

Start
 > **Programs**
 > **Administrative Tools**
 > **Event Viewer**

- Windows 2000:

Start
 > **Settings**
 > **Control Panel**
 > **Administrative Tools**
 > **Event Viewer**

Windows Error Message Explanations

Following are typical Windows error messages and their meanings:

`cis_get_hba_wwns warning: fencing configuration file "fencing.conf" not found`

The fencing file `fencing.conf` was not found, therefore the fencing configuration will not be updated on the server.

`op_failed ERROR: Mount failed for concat0`

A filesystem mount has failed and will be retried.

`cis_generate_userid_map warning: could not open passwd file`

The `passwd` file could not be found.

`cis_generate_userid_map warning: could not open group file`

The `group` file could not be found.

Even with `passwd` and `group` warnings above, filesystem mounts should proceed; however, all users will be given `nobody` credentials and will be unable to view or modify files on the CXFS filesystems. For more information about these files, see "Log Files on Solaris", page 79, and "Windows Log Files and Cluster Status", page 111. Also see the log files on the IRIX node; for more information, see the *CXFS Version 2 Software Installation and Administration Guide*.

`could not get location of passwd/group files`
`could not retrieving fencing configuration file name from registry`
`error retrieving passwd filename`
`error retrieving group filename`
`error retrieving fencing filename`

The registry entries for the location of the `passwd`, `group`, or `fencing.conf` files may be missing, or the path provided on the command line to the CXFS Client service is badly formed. Reset these values by modifying the current installation as described in "Modifying the CXFS for Windows Software", page 155.

`could not open passwd file`
`could not open group file`
`fencing configuration file not found`

Check that the `passwd`, `group` and `fencing.conf` files are in the configured location and are accessible as described in "Checking Permissions on the Password, Group, and Fencing Files", page 150.

Unix user is something other than a user on the NT domain/workgroup
Unix group is something other than a group on the NT domain/workgroup

This warning indicates that a username or groupname is not a valid user or group on the Windows node, which may be confusing when examining file permissions.

no valid users configured in passwd file

No users in the passwd file could be matched to users on the Windows node. All users will be treated as user nobody for the purpose of all access control checks.

no valid groups configured in group file

No groups in the group file could be matched to groups on the Windows node. Attempts to display file permissions will most likely fail with the message Unknown Group Errors.

cis_driver_init() failed: could not open handle to driver
cis_driver_init() failed: could not close handle to CXFS driver

The CXFS driver may not have successfully started. Check the system event log for errors.

unable to create mount point
Configured drive letter may already be in use

Check that the configured drive letter is not already in use by a physical or mapped drive.

unable to join multicast group on interface
unable to create multicast socket
unable to allocate interface list
unable query interfaces
failed to configure any interfaces
unable to create multicast socket
unable to bind socket

Check the network configuration of the Windows node, ensuring that the private network is working and the Windows node can at least reach the meta-data server by using the ping command from a command shell.

Common Problems and Solutions

This section contains the following common problems and their solutions:

- "Incorrect Configuration"
- "Determining If a Client-Only Node Is Fenced", page 177
- "Common Host Bus Adapter Problems", page 177
- "Common Linux on IA32 Problems", page 178
- "Common Windows Problems", page 179

Incorrect Configuration

To avoid having trouble with the CXFS client-only node, ensure you have the correct configuration. See "Requirements", page 6.

Determining If a Client-Only Node Is Fenced

To determine if a client-only node is fenced, log in to IRIX and use the `hafence(1M)` command. For more details, see the *CXFS Version 2 Software Installation and Administration Guide*.

Common Host Bus Adapter Problems

Consult the following checklist to help you identify the problem with a host bus adapter (HBA):

- Is the HBA firmly seated in its PCI slot?
- Are all cables undamaged and connected?
- Is power applied to all devices?
- Do the link lights illuminate on all units?
- Is the problem confined to just one unit? If so, check the cabling between the switch and the unit; if no units are being shown, suspect cabling from the HBA.
- Is the Brocade switch properly licensed?

- For a Solaris node, did you enable fabric mode? See step 4, in "Installing the JNI HBA", page 83.
- For Windows node, check the QLogic management tool event and alarm logs. Select the following:

```
Start
  > Programs
    > QLogic Management Suite
      > SANsurfer
```

For more information:

- See *Installation Guide, FCE-6460 and FCE2-6460 PCI-to-Fibre Channel Host Bus Adapters (Solaris, Windows NT/2000, Novell, AIX, HP-UX, Mac OS, Linux) JNI FibreStar*
- QLogic documentation:
 - *Hardware Installation Guide for the QLA2xxx Board Family*
 - *Software Installation Guide for the QLA2xxx Board Family*

Also see the QLogic website at:

<http://www.qlogic.com>

- Contact your SGI service representative

Common Linux on IA32 Problems

The kernels provided for the Linux client on supported IA32 platforms have `devfs` (the Device File System) enabled. This can cause problems with locating system devices in some circumstances.

You should replace references to filesystem labels (`LABEL=`) from the `/etc/lilo.conf` and `/etc/fstab` files with actual device names.

See the `devfs` FAQ at the following location:

<http://www.atnf.csiro.au/people/rgooch/linux/docs/devfs.html>

Common Windows Problems

This section contains the following common Windows problems:

- "Filesystems Are Not Displayed on a Windows Node", page 179
- "Large Log Files on Windows", page 179
- "Windows Failure on Restart", page 180
- "Memory Configuration of the Windows Node", page 180

Filesystems Are Not Displayed on a Windows Node

If the CXFS drive letter is visible in Windows Explorer but no filesystems are mounted, do the following:

- Run `C:\Program Files\CXFS\cxfs_info` to ensure that the filesystems have been configured for this node.
- Verify the filesystems that should be mounted by using the `cmgr(1M)` command on an IRIX node. For more information, see "Mounting Filesystems on the Client-Only Nodes", page 164.
- Ensure that the CXFS meta-data server is up and that the Windows node is in the cluster membership; see "Verifying the Cluster", page 165.
- Check that the CXFS Client service has started. See "Is the CXFS Software Running Correctly on the Windows Node?", page 172, and "Manual CXFS Startup/Shutdown for Windows", page 154.
- Check the following file for warnings and errors regarding licenses or mounting filesystems:

`C:\Program Files\CXFS\log\cxfs_client.log`
- Check the cluster configuration to ensure that this node is configured to mount one or more filesystems.

Large Log Files on Windows

The CXFS Client service creates the following log file:

`C:\Program Files\CXFS\log\cxfs_client.log`

This log file may become quite large over a period of time if the verbosity level is increased. The service does not perform any automatic log rotation, so the service must be stopped in order to move or truncate this file, then restarted. See "Manual CXFS Startup/Shutdown for Windows", page 154, on how to stop and start the CXFS Client Service.

Windows Failure on Restart

If the CXFS Windows node fails to start and terminates in a blue screen, restart your computer, and select the backup hardware profile (with CXFS disabled). Alternatively, pressing **L** at the **Hardware Profile** menu will select the last configuration that was successfully started and shut down. If the node has only one hardware profile, press the spacebar after selecting the boot partition to get to the **Hardware Profile** menu.

Memory Configuration of the Windows Node

A Windows problem may affect Windows CXFS nodes performing large asynchronous I/O operations. If the Windows node crashes with a `NO_MORE_SYSTEM_PTES` message, the work-around described in the following link should be considered (line break added here for readability):

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/winxppro/reskit/prmd_stp_fztl.asp

Reporting Problems

When reporting a problem with a client-only node, it is important to retain the appropriate information; having access to this information will greatly assist SGI in the process of diagnosing and fixing problems. The methods used to collect required information for problem reports are platform-specific:

- "Reporting AIX Problems", page 181
- "Reporting Linux on IA32 Problems", page 182
- "Reporting Linux on IA32 Problems", page 182
- "Reporting Solaris Problems", page 183
- "Reporting Windows Problems", page 184

Reporting AIX Problems

When reporting a problem about a CXFS AIX node to SGI, you should retain the following information:

- Information about the AIX node system dump and system configuration:

```
aix# snap -a -o /dev/rmt0
```

- Console log:

```
aix# alog -o -t console
```

- Current syslog file
- The /var/tmp/cxfs_client CXFS log file
- Moduler debugger output from the kdb command:
 - For panics or generated dumps, use the following commands and save the output:

```
aix# kdb /var/adm/ras/vmcore.xx[/unix]
(0)> stat
```

- For dumps from hangs:

```
aix# kdb /var/adm/ras/vmcore.xx[/unix]
(0)> th* (to find the slot value of the working process or thread)
(0)> sw slot_value
(0)> stat
```

- A list of the installed CXFS packages. Use the `lslpp` command as follows:

```
aix# lslpp -l SGIcxfs-aix5L
```

- The version information of the operating system. Use the following `oslevel` commands:

```
aix# oslevel -r
aix# oslevel -g | grep bos.64bit
```

- A list of the loaded AIX kernel extensions. Use the `genkex` command.
- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

If any of these AIX tools are not currently installed on your AIX node, you should install them.

Reporting Linux on IA32 Problems

When reporting a problem about a CXFS 32-bit Linux node to SGI, you should retain the following information:

- The kernel you are running:

```
[root@linux root]# uname -a
```

- The CXFS packages you are running:

```
[root@linux root]# rpm -q cxfs_client cxfs-modules xvm-cluster-cmds
```

- The processor type of your machine:

```
[root@linux root]# cat /proc/cpuinfo
```

- The hardware installed on your machine:

```
[root@linux root]# /sbin/lspci
```

- Modules that are loaded on your machine:

```
[root@linux root]# /sbin/lsmmod
```

- The `/var/log/cxfs_client` log file
- Any messages that appeared in the system logs immediately before the system exhibited the problem.
- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

Reporting SGI ProPack Problems

When reporting a problem about a CXFS SGI ProPack for Linux node, you should retain the following information:

- The kernel you are running:

```
[root@linux root]# uname -a
```

- The CXFS packages you are running:

```
[root@linux root]# rpm -q cxfs_client cxfs-modules xvm-cluster-cmds
```

- The number of CPUs in your machine:

```
[root@linux root]# cat /proc/cpuinfo
```

- The hardware installed on your machine:

```
[root@linux root]# /sbin/lspci
```

- Modules that are loaded on your machine:

```
[root@linux root]# /sbin/lsmmod
```

- The `/var/log/cxfs_client` log file
- Any messages that appeared in the system logs immediately before the system exhibited the problem.
- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

Reporting Solaris Problems

When reporting a problem about a CXFS Solaris node to SGI, you should retain the following information:

- If there is a system panic, retain the system core file in `/var/crash/hostname` on a Solaris node.
- Output from the `crash` utility.
- `mdb(1M)` modular debugger output:
 - For panics or generated dumps, use the following commands and save the output:
 - `$c` (or `$C`)
 - `$r`
 - `$<msgbuf`

- For dumps from hangs:
 - `$<threadlist`
 - `$c` (or `$C`)
 - `$r`
 - `$<msgbuf`
- A list of the installed CXFS packages. Use the `pkginfo(1)` command as follows:

```
# pkginfo -l SGIcxf
```
- A list of the Solaris patches that have been installed. Use the `showrev` command. The `showrev` command without options prints a summary and the `-p` option lists the revision information about patches.
- A list of the loaded Solaris kernel modules and versions. Use the `modinfo` command.
- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

If any of the above Solaris tools are not currently installed on your Solaris system, you should install them.

Reporting Windows Problems

To report problems about a Windows node, you should retain platform-specific information and save crash dumps.

Retain Windows NT Information

When reporting a problem about a CXFS Windows NT node to SGI, you should retain the following information:

- The version and memory of the current diagnostics. Select the following:

Start

> **Programs**
> **Administrative Tools**
> **Windows NT Diagnostics**

Record the version displayed in the **Version** tab and the total physical memory in the **Memory** tab.

- The build date and firmware versions. Using Windows Explorer, open the following directory:

C:\Winnt\system32\drivers

Then do the following:

- Right click on `cxfs.sys` and select the following:

Properties
> **Version**

Record the values of **BuildDate** and **Product Version**.

- Right click on `ql2200.sys` and select the following:

Properties
> **Version**

Record the values of **Firmware** and **Product Version**.

- The contents of the following file:

C:\Program Files\CXFS\log\cxfs_client.log

Compress this file with `winzip` if it is large.

- The contents of the crash dump if one was generated, Compress this file with `winzip`. For more information, see "Save Crash Dumps for Windows", page 187.

- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

Retain Windows 2000 Information

When reporting a problem about a CXFS Windows 2000 node to SGI, you should retain the following information:

- The configuration of the machine. Select the following:

```
Start
  > Programs
    > Accessories
      > System Tools
        > System Information
          > Action
            > Save As System Information File
```

This will create a file that describes all of the installed hardware and configured drivers on the machine.

Alternatively, you could dump information about each item in the hardware tree to a text file by using the following selection:

```
Action
  > Save As Text File
```

However, you must repeat this action for each item.

- The build date and firmware versions. Using Windows Explorer, open the following directory:

```
C:\Winnt\system32\drivers
```

Then do the following:

- Right click on `cxfs.sys` and select the following:

```
Properties
  > Version
```

Record the values of **BuildDate** and **Product Version**.

- Right click on `ql2200.sys` and select the following:

Properties
> Version

Record the values of **Firmware** and **Product Version**.

- The contents of the following file:
`C:\Program Files\CXFS\log\cxfs_client.log`
 Compress this file with `winzip` if it is large.
- The contents of the crash dump if one was generated, Compress this file with `winzip`. For more information, see "Save Crash Dumps for Windows", page 187.
- Output about the cluster obtained from the IRIX `cxfsdump` utility run on an IRIX node. This utility is available from your SGI service representative.

Save Crash Dumps for Windows

If you are having problems, you should configure the Windows node to save crash dumps to a filesystem that is not a CXFS filesystem. To do this, click the right mouse button on the **My Computer** icon and select the following:

- Windows NT:

Properties
> Startup
> Shutdown
> Write debugging information to

- Windows 2000:

Properties
> Advanced
> Startup and Recovery
> Write debugging information to

Enter a path on a filesystem other than a CXFS filesystem. This change will take affect only after the node is restarted.

Glossary

active meta-data server

A server-capable administration node chosen from the list of potential meta-data servers. There can be multiple active meta-data servers, one for each file system.

client

See *CXFS client*.

cluster

A cluster is the set of systems (nodes) configured to work together as a single computing resource. A cluster is identified by a simple name and a cluster ID. A cluster running multiple operating systems is known as a multiOS cluster.

There is only one cluster that may be formed from a given pool of nodes.

Disks or logical units (LUNs) are assigned to clusters by recording the name of the cluster on the disk (or LUN). Thus, if any disk is accessible (via a Fibre Channel connection) from machines in multiple clusters, then those clusters must have unique names. When members of a cluster send messages to each other, they identify their cluster via the cluster ID. Thus, if two clusters will be sharing the same network for communications, then they must have unique cluster IDs. In the case of multiOS clusters, both the names and IDs must be unique if the clusters share a network.

Because of the above restrictions on cluster names and cluster IDs, and because cluster names and cluster IDs cannot be changed once the cluster is created (without deleting the cluster and recreating it), SGI advises that you choose unique names and cluster IDs for each of the clusters within your organization.

cluster database

Contains configuration information about nodes, filesystems, and the cluster. The database is managed by the `fs2d` daemon and is stored on IRIX nodes in the pool of nodes that are running the `fs2d` daemon.

cluster ID

A unique number within your network in the range 1 through 128. The cluster ID is used by the IRIX kernel to make sure that it does not accept cluster information from

any other cluster that may be on the network. The kernel does not use the database for communication, so it requires the cluster ID in order to verify cluster communications. This information in the kernel cannot be changed after it has been initialized; therefore, you must not change a cluster ID after the cluster has been defined. Clusters that share a network must have unique names and IDs.

cluster node

A node that is defined as part of the cluster.

control messages

Messages that cluster software sends between the cluster nodes to request operations on or distribute information about cluster nodes. Control messages and heartbeat messages are sent through a node's network interfaces that have been attached to a control network.

A node's control networks should not be set to accept control messages if the node is not a dedicated CXFS node. Otherwise, end users who run other jobs on the machine can have their jobs killed unexpectedly when CXFS resets the node.

control network

The network that connects nodes through their network interfaces (typically Ethernet) such that CXFS can send heartbeat messages and control messages through the network to the attached nodes. CXFS uses the highest priority network interface on the control network; it uses a network interface with lower priority when all higher-priority network interfaces on the control network fail.

CXFS client

A node that is part of the cluster and is a potential meta-data server, but is currently not acting as the active meta-data server. See also *IRIX node* and *CXFS client-only node*.

CXFS client-only node

A node that is part of the cluster but is not a potential meta-data server. AIX, Linux, Solaris, and Windows nodes are always client-only nodes. See also *IRIX node*.

CXFS database

See *cluster database*.

CXFS membership

The group of CXFS nodes that can share filesystems in the cluster, which may be a subset of the nodes defined in a cluster. During the boot process, a node applies for CXFS membership. Once accepted, the node can share the filesystems of the cluster. (Also known as *kernel-space membership*.)

database

See *cluster database*.

GUI

Graphical user interface.

heartbeat messages

Messages that cluster software sends between the nodes that indicate a node is operational. Heartbeat messages and control messages are sent through the node's network interfaces that have been attached to a control network.

I/O fencing

The failure action that isolates a problem node so that it cannot access I/O devices, and therefore cannot corrupt data in the shared CXFS filesystem. I/O fencing can be applied to any node in the cluster (CXFS clients and meta-data servers). The rest of the cluster can begin immediate recovery.

IRIX node

A CXFS node that is running the IRIX operating system. An IRIX node can be a potential meta-data server if it is configured as such and is server-capable, or it can be a CXFS client.

membership

See *CXFS membership*.

meta data

Information that describes a file, such as the file's name, size, location, and permissions.

meta-data server

The IRIX node that coordinates updating of meta data on behalf of all nodes in a cluster. There can be multiple potential meta-data servers, but only one is chosen to be the active meta-data server for any one filesystem. See also *active meta-data server* and *potential meta-data server*.

multiOS cluster

A cluster that is running multiple operating systems, such as IRIX and Solaris.

node

A node is an operating system (OS) image, usually an individual computer. (This use of the term node does not have the same meaning as a node in an SGI Origin 3000 or SGI 2000 system.)

A given node can be a member of only one pool (and therefore) only one cluster.

A node can run the IRIX operating system or another operating system, such as Solaris, as defined in the *CXFS Version 2 Software Installation and Administration Guide*.

node membership

The list of nodes that are active (have CXFS membership) in a cluster.

pool

The pool is the set of nodes from which a particular cluster may be formed. Only one cluster may be configured from a given pool, and it need not contain all of the available nodes. (Other pools may exist, but each is disjoint from the other. They share no node or cluster definitions.)

A pool is formed when you connect to a given node and define that node in the cluster database using the CXFS GUI or `cmgr` command. You can then add other nodes to the pool by defining them while still connected to the first node, or to any other node that is already in the pool. (If you were to connect to another node and then define it, you would be creating a second pool).

potential meta-data server

A server-capable administration node that is listed in the meta-data server list when defining a filesystem; there can be multiple potential meta-data servers, but only one node in the list will be chosen as the active meta-data server for one filesystem.

recovery

The process by which the meta-data server moves from one node to another due to an interruption in services on the first node.

relocation

The process by which the meta-data server moves from one node to another due to an administrative action; other services on the first node are not interrupted.

SAN

Storage area network, a high-speed, scalable network of servers and storage devices that provides storage resource consolidation, enhanced data access/availability, and centralized storage management.

standby node

A server-capable administration node that is configured as a potential meta-data server for a given filesystem, but does not currently run any applications that will use that filesystem.

tree view

The portion of the CXFS GUI window that displays components graphically.

quorum

The number of nodes required to form a cluster.

Index

8-port switch, 20
16-port switch, 20
100baseT TCP/IP network, 6
4774 and 4884 units, 17

A

acledit, 34
aclget, 34
aclput, 34
ACLs
 AIX, 34, 35
 Linux, 50
 SGI ProPack, 66
 Solaris, 81
Active Directory user ID mapping method, 141
adapter parameters, 90, 93
add
 client-only node to the cluster, 161
 package, 103
 private network
 Windows 2000, 137
 Windows NT, 135
adjacent releases not supported in multiOS, 6
admin cxfs_unmount, 165
administrative tasks, 4
AIX
 client software installation, 41
 commands installed by CXFS, 32
 FLEXlm license verification, 35
 HBA installation, 36
 limitations, 33
 log files, 33
 manual CXFS startup/shutdown, 45
 minimum block size, 34
 modify the CXFS software, 46

operating system version, 31
postinstallation steps, 44
preinstallation steps, 36
requirements, 31
software
 maintenance, 46
 upgrades, 46
 space requirements, 41
AIX hardware, 32
AMD CPUs, 48

B

bandwidth, 2, 5
BIOS version, 110
block size, 50, 65
 AIX, 34
 Solaris, 80
boot command, 84
boot Solaris in single-user mode, 97
Brocade
 license, 3
 switch, 7, 19
buffered I/O and CXFS, 5
build date for Windows NT, 185, 186

C

\$c or \$C, 183
C:\Program Files\CXFS directory, 138
C:\Program Files\CXFS\log\cxfs_client.log
 file, 174, 179
C:\Winnt\system32\drivers directory, 185, 186
cables and JNI HBA, 177
client software installation

- AIX, 41
 - Linux, 57
 - SGI ProPack, 72
 - Solaris, 102
 - Windows, 138
 - client-only node configuration
 - add to the cluster, 161
 - define the node, 159
 - define the switch, 162
 - modify the cluster, 161
 - mount filesystems, 164
 - permit fencing, 159
 - start CXFS services, 164
 - verify the cluster, 166
 - cluster
 - configuration, 159
 - size, 8
 - verification, 166
 - cluster_status command, 166
 - cmgr command, 159
 - command shell for DOS, 146
 - commands installed
 - AIX, 32
 - Linux, 49
 - SGI ProPack, 64
 - Solaris, 49, 64, 79
 - Windows, 111
 - common problems
 - HBA, 177
 - Windows, 179
 - concepts, 2
 - configuration overview, 11–14
 - configuration problems, 177
 - configure
 - cluster, 159
 - network interfaces, 135
 - Windows system files, 133
 - contacting SGI with problems
 - Linux, 182
 - SGI ProPack, 182
 - Solaris, 183
 - Windows 2000, 186
 - Windows NT, 185
 - controller firmware, 17, 18
 - core files, 183
 - CPU types for Linux, 48
 - cpuinfo, 182, 183
 - crash dumps
 - Solaris, 183
 - Windows, 187
 - crash utility and gathering output, 183
 - crontab, 8
 - CXFS
 - GUI and cmgr, 159
 - license, 3
 - meta-data server, 3
 - software removal on Windows, 157
 - startup/shutdown
 - Windows, 154
 - CXFS Client service
 - command line arguments, 141
 - CXFS startup/shutdown
 - Windows, 154
 - cxfs_client command, 79
 - cxfs_cluster command, 45, 60, 74, 106
 - cxfsdump, 184, 186, 187, 182, 183
 - cxfslicense command, 25, 35, 43, 51, 58, 66, 73, 79, 82, 104
- D**
- define a client-only node, 159
 - defragmentation software and CXFS, 8
 - dflt_local_status, 165
 - DHCP, 136
 - direct-access I/O, 2
 - disk device verification for Solaris, 95
 - display LUNs for QLogic HBA, 130
 - distributed applications and CXFS, 6
 - dmesg command, 99
 - DNS, 98
 - Linux, 54

- SGI ProPack, 69
- Solaris, 98
- DOS command shell, 146
- driver software for JNI HBA, 84
- dumps and output to gather, 183, 184

E

- entitlement ID, 25
- Entitlement Sheet, 6
- error messages
 - Linux, 170, 171
 - Solaris, 172
 - Windows, 174
- /etc/fencing.conf file, 59, 73, 92, 104, 147
 - AIX, 44
 - Linux, 59
 - SGI ProPack, 73
 - Solaris, 104
- /etc/hostname.<interface> file, 100
- /etc/hosts file, 160
 - Linux, 53
 - SGI ProPack, 68
 - Solaris, 97
- /etc/inet/hosts file, 160
 - Solaris, 97
- /etc/init.d/cxfs_cluster command, 45, 60, 74, 106
- /etc/netmasks file, 100
- /etc/nodename file, 100
- /etc/nsswitch.conf file, 98
 - Linux, 52
 - SGI ProPack, 68
 - Solaris, 97
- /etc/redhat-release, 57
- /etc/sgi-release, 72
- /etc/sys_id file, 100
- Event Viewer, 111
- examples
 - add a client-only node to the cluster, 161
 - CXFS software installation
 - AIX, 41

- Linux, 57
- SGI ProPack, 72
- Solaris, 103
- Windows, 140
- define a node, 160
- define a switch, 162
- /etc/hosts file
 - Linux, 54
 - SGI ProPack, 69
 - Solaris, 98
- /etc/inet/hosts file
 - Linux, 54
 - SGI ProPack, 69
 - Solaris, 98
- fabric enable, 84
- ifconfig for AIX, 40
- ifconfig for Linux, 56
- ifconfig for SGI ProPack, 71
- ifconfig for Solaris, 102
- JNI GUI screens, 87
- license properly installed, 25
- modify the cluster, 161
- modifying the CXFS
 - AIX, 46
 - Solaris, 107
 - Windows, 155
- mount filesystems, 165
- name services
 - Linux, 54
 - SGI ProPack, 69
 - Solaris, 98
- ping output for AIX, 40
- ping output for Linux, 56
- ping output for SGI ProPack, 71
- ping output for Solaris, 102
- private network interface test for AIX, 40
- private network interface test for Linux, 56
- private network interface test for SGI ProPack, 71
- private network interface test for Solaris, 102
- .rhosts, 101

- start CXFS services, 164
- upgrade CXFS for Windows software, 157
- verify the cluster configuration, 166
- Windows Client service command line
 - options, 156
- EZ Fibre GUI, 85
- ezf, 86

F

- fabric mode, 84
- fail action hierarchy, 160
- FailSafe coexecution, 7
- failure on restart, 180
- FcFabricEnabled, 84
- FcLoopEnabled, 84
- FCO 1056, 17
- fence specification in node definition, 160
- fencing feature, 19
- fencing.conf file, 59, 73, 92, 104, 147
 - AIX, 44
 - Linux, 59
 - SGI ProPack, 73
 - Solaris, 104
- Fibre Channel HBA
 - See "host bus adapter", 51, 66
- Fibre Channel requirements
 - AIX, 32
- Fibre Channel requirements for Linux, 48
- Fibre Channel requirements for Solaris, 78
- file size, 5
- file, filesystem size maximum
 - AIX, 35
 - Linux, 50
 - SGI ProPack, 66
 - Solaris, 81
- filesystem network access, 2
- filesystems not displayed on Windows, 179
- find and crontab, 8
- firmware for RAID, 17
- FLEXlm

- license key, 6
- license requirements for CXFS, 25
- license verification
 - AIX, 35
 - Linux, 51
 - SGI ProPack, 66
 - Solaris, 82
 - Windows, 146
- licenses, 3
- forced unmount, 9
- format command, 95
- free disk space required, 110
- fuser, 10

G

- generated dumps and output to gather, 183
- good practices, 9
- guided configuration, 159

H

- hangs and output to gather, 184
- hardware installed, 182, 183
- hardware platforms
 - AIX, 32, 78
- hardware profile, 151
- hardware requirements
 - AIX, 31
 - all platforms, 6
 - Linux, 48
 - SGI ProPack, 64
 - Solaris, 78
 - Windows, 110
- HBA, 84
 - AIX, 32
 - problems, 177
 - Solaris, 78
- hierarchy of fail actions, 160

- host bus adapter (HBA)
 - AIX, 36
 - Linux, 51
 - SGI ProPack, 66
 - Solaris, 83
- hostid command, 25, 27
- hostname, 26, 27
- hostname command, 25, 27
- hostname resolution, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
- hostname.<interface> file, 100
- hosts file, 160
 - Linux, 53
 - SGI ProPack, 68
 - Solaris, 97
- hub, 8

I

- I/O fencing, 7, 19
- I/O fencing file
 - Solaris, 92
- I/O operations, 2
- icons on JNI GUI, 88
- identifying problems
 - Linux on IA32, 170
 - SGI ProPack, 171
 - Solaris, 169, 171
 - Windows, 172
- ifconfig, 26, 27
- ifconfig command, 40, 56, 71, 98, 99, 102
- incorrect configuration, 177
- initial setup services, 1
- initialize the private network's interface for
 - Solaris, 99
- install.sh script, 86
- installation overview, 11–14
- installed packages, 184

- installed patches, 184
- installp, 41
- integrated Ethernet, 99
- Intel CPUs, 48
- Intel Pentium processor, 110
- interface for the private network, 99
- internode communication, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
- introduction, 1
- IP address, changing, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
- IP forwarding, 136
- ipconfig, 138
- IRIX
 - labels in warning messages, 95
 - node required, 4, 6
 - operating system level, 6

J

- JBOD, 6
- JNI
 - HBA requirement, 78
 - HBA requirement for Linux, 48
- JNIC146x, 84
- jnic146x.conf file, 84
- JNIsnia, 84

K

- kernel modules and versions, 184
- kernel running on Linux, 182
- kernel running on SGI ProPack, 182

/kernel/drv/jnic146x.conf file, 84, 85

L

large files, 2

LDAP generic user ID mapping method, 142

license

 Brocade, 3, 20

 CXFS, 3, 25

 FLEXlm for CXFS and XVM, 25

 installation, 28

 mirroring, 25

 SGI webpage, 29

 verification

 Solaris, 82

 verification on AIX, 35

 verification on Linux, 51

 verification on SGI ProPack, 66

 verification on Solaris, 82

 XVM, 3, 25

licenseshow command, 20

licensing, 6

link lights and JNI HBA, 177

Linux

 client software installation, 57

 commands installed by CXFS, 49

 error messages, 170, 171

 FLEXlm license verification, 51

 HBA installation, 51

 limitations, 49

 manual CXFS startup/shutdown, 60

 minimum block size, 50

 postinstallation steps, 59

 preinstallation steps, 52

 software

 maintenance, 60

 space requirements, 57

 user and group quotas, 49

Linux IA32

 requirements, 48

Linux on IA32

 identifying problems, 170

log files

 AIX, 33

 Solaris, 79

 Windows, 111, 179

lspp, 33, 43

lsmod, 182, 183

lspci, 182, 183

LUN

 logical unit, 93

 LUN 31 and mapping, 93

 mapping, 93

 maximum with TP9400, 18

 zoning, 92

M

 maintenance and cluster services, 9

 manual CXFS startup/shutdown

 AIX, 45

 Linux, 60

 SGI ProPack, 74

 Solaris, 107

 Windows, 154

 mapping

 LUNs, 93

 masking feature and Solaris, 162

 mdb, 183

 memory mapped shared files and CXFS, 6

 messages

 See "error messages", 170–172, 174

 meta data

 concept, 2

 server, 3

 transaction examples, 5

 transfer and performance considerations, 5

 minimum block size, 50, 65

 AIX, 34

 Solaris, 80

 mirroring feature and license, 3, 25

- missing filesystems on Windows, 179
- modify cluster command, 161
- modify the CXFS software
 - AIX, 46
 - Solaris, 107
 - Windows, 155
- modinfo, 184
- modules and versions, 184
- modules loaded on Linux, 182
- modules loaded on SGI ProPack, 183
- Mojave code, 17
- mount filesystems, 164
- mount-point nesting on Solaris, 79
- msgbuf, 183
- \$(msgbuf, 183
- multiOS cluster, 1

N

- name restrictions, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
- name service daemon, 98
- nested mount points on Solaris, 79
- netmasks file, 100
- network
 - information service, 98
 - interface configuration, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
 - issues, 8
 - requirements, 6
 - switch, 8
- NFS and CXFS, 5, 79
- NIS, 98
 - Linux, 54
 - SGI ProPack, 69

- Solaris, 98
- nsd, 98
- nsswitch.conf file, 98
 - Linux, 52
 - SGI ProPack, 68
 - Solaris, 97
- number of nodes supported, 7
- NVSRAM files, 17, 18

O

- O2, 6
- operating system level, 6
- order desk, 29
- OS level, 6
- output to gather
 - Linux, 182
 - SGI ProPack, 182
 - Solaris, 183
 - Windows 2000, 186
 - Windows NT, 185
- overview, 11–14

P

- packages installed, 184
- packages running on Linux, 182
- packages running on SGI ProPack, 183
- panics and output to gather, 183
- partitioned system licensing, 6
- partitioning key for TP9400, 18
- passwd and group files user ID mapping method, 117
- patches installed, 184
- PCI slot and JNI HBA, 177
- performance considerations
 - Solaris, 5
 - Windows, 5

- permissions for Windows passwd and group files, 151
- ping, 40, 56, 71, 102
- pkgadd command, 79, 103
- pkginfo command, 104, 184
- plexing license, 3
- plumb, 99
- postinstallation steps
 - AIX, 44
 - Linux, 59
 - SGI ProPack, 73
 - Solaris, 104
 - Windows, 145
- power management software and CXFS, 9
- power to JNI HBA, 177
- preinstallation steps
 - AIX, 36
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
 - Windows, 133
- premount and postmount scripts, 4
- primary hostname
 - Solaris, 97
 - Windows, 138
- private network
 - AIX, 37
 - heartbeat and control, 135
 - AIX, 37
 - Linux, 52
 - SGI ProPack, 67
 - Solaris, 96
 - interface test for AIX, 40
 - interface test for Linux, 56
 - interface test for SGI ProPack, 71
 - interface test for Solaris, 102
 - Linux, 53
 - required, 6
 - SGI ProPack, 68
 - Solaris, 97
- problem reporting
 - Linux, 182

- Solaris, 183
- Windows 2000, 186
- Windows NT, 185

processor type on Linux, 182

processor type on SGI ProPack, 183

pSeries, 32

public network

- Solaris, 99

Q

- Qlogic HBA installation, 129
- QLogic HBA model numbers and driver versions, 110

R

- \$r, 183
- RAID firmware, 17
- READ_CAPACITY SCSI command, 81
- recommendations for CXFS
 - general, 8
- recovery, 9
- Red Hat version, 48
- relocation, 9
- remove CXFS software
 - Windows, 157
- reporting problems
 - Linux, 182
 - SGI ProPack, 182
 - Solaris, 183
 - Windows 2000, 186
 - Windows NT, 185
- requirements for CXFS
 - AIX, 31
 - all platforms, 6
 - Linux, 48
 - SGI ProPack, 64

- Solaris, 78
- Windows, 110
- reset lines, 7
- /.rhosts file, 100, 101
- rolling upgrades, 6
- rpm, 57, 182, 183

S

- SANblade Manager, 110
- SANblade NT Agent, 110
- serial reset lines, 7
- service pack, 111
- set dflt_local_status, 165
- setup program for Windows, 139
- setup services, 1
- SGI ProPack
 - client software installation, 72
 - commands installed by CXFS, 64
 - FLEXlm license verification, 66
 - HBA installation, 66
 - identifying problems, 171
 - limitations, 65
 - manual CXFS startup/shutdown, 74
 - minimum block size, 65
 - postinstallation steps, 73
 - preinstallation steps, 67
 - requirements, 64
 - software
 - maintenance, 74
 - space requirements, 72
 - user and group quotas, 65
- SGIcxf package, 103
- shell for DOS, 146
- showrev command, 184
- Silicon Graphics O2, 6
- Silkworm switch, 19
- single-user mode in Solaris, 97
- size of the cluster, 7
- small files, 5
- SNIA API package, 84

- software maintenance
 - AIX, 46
 - Linux, 60
 - SGI ProPack, 74
 - Solaris, 107
 - Windows, 154
- software requirements
 - AIX, 31
 - all platforms, 6
 - Linux, 48
 - SGI ProPack, 64
 - Solaris, 78
 - Windows, 110
- software upgrades
 - AIX, 46
 - Solaris, 107
 - Windows, 156
- Solaris
 - client software installation, 102
 - commands installed by CXFS, 79
 - error messages, 172
 - FLEXlm license verification, 82
 - identifying problems, 169, 171
 - JNI HBA installation, 83
 - kernel
 - modules and versions, 184
 - limitations, 79
 - log files, 79
 - manual CXFS startup/shutdown, 107
 - minimum block size, 80
 - modify the CXFS software, 107
 - non-disk devices and JNI controllers, 81
 - operating system version, 78
 - performance considerations, 5
 - postinstallation steps, 104
 - preinstallation steps, 96
 - recommendations, 8
 - requirements, 6, 78
 - software
 - maintenance, 107
 - upgrades, 107

- space requirements, 103
- user
 - administration for CXFS, 5
 - user and group quotas, 80
- solutions to common problems
 - Windows, 179
- space requirements
 - AIX, 41
 - Linux, 57
 - SGI ProPack, 72
 - Solaris, 103
- standby node, 9
- start
 - CXFS processes
 - AIX, 45
 - Linux, 60
 - SGI ProPack, 74
 - Solaris, 106
 - Windows, 154
 - CXFS services, 154, 164
- stop CXFS processes on AIX, 45
- stop CXFS processes on Solaris, 60, 74, 107
- Storage Networking Industry Association
 - application programming interface package, 84
- subnet, 6
- Sun hardware, 78
- switch, 8, 19
- switch definition, 162
- switchshow, 44
 - Solaris, 105
 - Windows, 148
- sys_id file, 100
- system core files, 183
- System Event log, 174

T

- TCP/IP network requirements, 6
- telnet, 148
- telnet port and I/O fencing, 7
- \$(threadlist), 184

- TP9100, 18
- TP9400 RAID firmware, 17
- TPxxxx RAID troubleshooting, 177
- TRIX and Solaris nodes, 7
- troubleshooting, 169
- Trusted IRIX and Solaris nodes, 7

U

- UFS and CXFS, 79
- Ultra Enterprise platforms, 78
- umount, 10
- uname, 41, 59, 103, 182
- unmount filesystems good practices , 9
- unmount the filesystems, 9
- upgrade CXFS software
 - AIX, 46
 - Solaris, 107
 - Windows, 156
- upgrading IRIX, 6
- user and group quotas, 49, 65, 80
- User ID mapping methods, 117
 - Active Directory, 141
 - Generic LDAP, 142
- /usr/bin/hostid command , 25, 27
- /usr/bin/hostname command, 25, 27
- /usr/bin/showrev command, 184
- /usr/cxfs_cluster/bin/cxfslicense command , 25, 35, 51, 66, 82

V

- /var/cluster/cmgr-scripts/cluster_status, 166
- /var/crash/<hostname>, 183
- /var/log/cxfs_client log file, 33, 79
- verify
 - Brocade license and firmware, 20
 - cluster, 166
 - FLEXlm license

- Solaris, 82
- Windows, 146
- verify networks
 - Windows, 138
- version command , 20
- version of Windows NT, 185, 186
- versions of modules installed, 184
- volume plexing license, 3

W

- warning message and IRIX labels, 95
- when to use CXFS, 2
- Windows
 - client software installation steps, 138
 - crash dumps, 187
 - CXFS commands installed, 111
 - CXFS software removal, 157
 - debugging information, 187
 - DHCP, 136
 - error messages, 174
 - failure on restart, 180
 - filesystems not displayed, 179
 - FLEXlm license verification, 146
 - hardware profile, 151
 - heartbeat and control, 135
 - hostname resolution, 135
 - identifying problems, 172
 - IP address, changing, 135
 - IP forwarding, 136
 - ipconfig, 138
 - large log files, 179
 - log files, 111
 - LUNs, 130
 - manual CXFS startup/shutdown, 154
 - modify the CXFS software, 155
 - name restrictions, 135
 - network interface configuration, 135
 - performance

- considerations, 5
- postinstallation steps, 145
- preinstallation steps, 133
- private network, 135
- problems, 179
- Qlogic HBA installation, 129
- recommendations, 8
- requirements, 6, 110
- software maintenance, 154
- software upgrades, 156
- user, 5
- verify networks, 138
- version, 185, 186
- WINS server, 136
- Windows NT
 - build date, 185, 186
- winnt/setup.exe command, 139, 156
- WINS server, 136
- worldwide node name, 85
- worldwide number
 - AIX, 44
- worldwide port name, 59, 73, 85, 104, 147, 148
 - Linux, 59
 - SGI ProPack, 73
 - Solaris, 105
- WWNN, 85
- WWPN, 59, 73, 85, 104, 147, 148
 - AIX, 44
 - Linux, 59
 - SGI ProPack, 73
 - Solaris, 105

X

- xfrepair, 8
- xvm command, 79
- XVM mirroring license, 3, 25
- xvmprobe command, 79