# Version 1.10 System Manager's Guide



## Copyright

Copyright ©1998, Auspex Systems, Inc. All rights reserved. Printed in the United States of America. Part Number 850517-001.

No part of this publication may be reproduced, in any form or by any means, without the prior written consent of Auspex Systems, Inc. Auspex Systems, Inc., reserves the right to revise this publication and make changes in content from time to time without obligation on the part of Auspex Systems to provide prior notification of such revision or change.

U.S. GOVERNMENT RIGHTS: As specified in 48 C.F.R.12.212 of the FAR and in 48.C.F.R 227-7202-1 of the DFARS, the use, duplication or disclosure of licensed commercial software and documentation is subject to the Auspex System's license. Such rights and restrictions are similar to those set forth in FAR 52.227-19(c)(1)&(c)(2).

#### **Trademarks**

Auspex, Auspex logo design, Functional Multiprocessor, Functional Multi-processor, Functional Multi-processing, Functional Multiprocessing Kernel, FMK, and FMP are registered trademarks of Auspex Systems, Inc. NS 7000, NS 6000, NS 6002, NS 5500, NS 5502, NS 5000, NS 3000, NetServer, DataGuard, ServerGuard, Functional Multiprocessing, NeTservices, and Thrive Carefully are trademarks of Auspex Systems, Inc.

AT&T is a registered trademark of AT&T Corporation. Microsoft, MS, MS-DOS, Windows, Windows NT, and Backoffice are either registered trademarks or trademarks of Microsoft Corporation. Sun, Sun Microsystems, the Sun Logo, Solaris, SunOS, ONC, ONC/NFS, and NFS are trademarks or registered trademarks of Sun Microsystems, Inc. All SPARC trademarks are used under license and are the trademarks or registered trademarks of SPARC International, Inc. in the United States and other countries. UNIX is a registered trademark in the United States and other countries of The Open Group. VMEbus is a trademark of VMEbus Manufacturers Group. DEC and VT 510 are trademarks of Digital Equipment Corp. ForeRunner is a trademark of FORE Systems, Inc. Acrobat is a trademark of Adobe Systems, Inc.

Auspex NetServer<sup>TM</sup> System Software is derived from UNIX® licensed from The Santa Cruz Operation, Inc. and SunOs<sup>TM</sup> 4.1.4 and ONC<sup>TM</sup>/NFS<sup>TM</sup> 4.1 licensed from Sun Microsystems, Inc. Auspex NetServer System Software Version 1.10 incorporates AT&T's Advanced Server for UNIX® Systems. Auspex Optional Products Premier Software Series for NeTservices<sup>TM</sup> incorporates AT&T's Advanced Server for UNIX Systems and NETBIOS/ix<sup>TM</sup>. NETBIOS/ix is a registered U.S. trademark of Micro Computer Systems, Inc.

Microsoft may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property rights except as expressly provided in any written license agreement from Microsoft.

Auspex Systems, Inc. 2300 Central Expressway Santa Clara, California 95050 Phone: (408) 566-2000

Fax: (408) 566-2020 Internet: info@auspex.com

World Wide Web: http://www.auspex.com

## **Protection Against Electrostatic Discharge**

To prevent damage to the system due to electrostatic discharge, always wear the antistatic wrist strap provided with your network server when you come in contact with the system.

## **Publication Change Record**

The following table records all revisions to this publication. The first entry is always the publication's initial release. Each entry indicates the date of the release and the number of the system release to which the revision corresponds.

Part Number	Date	Description	
850517-001	March 1998	Release 1.10	



## **Contents**

Preface
Pretace
Chapter 1 Overview of NetServer Features
About This Chapter
Product Overview
NetServer Hardware Architecture1-5
Host Processor (HP)
Network Processor (NP)1-6
Storage Processor (SP)
I/O Cache Memory1-6
Enhanced VME Bus
Optional Products
ServerGuard1-7
DataGuard
DriveGuard1-7
NeTservices 1-7
EtherChannel
ATM1-8
FTP on the Network Processor
FastBackup
Hardware Implementation of the FMP Architecture
NetServer Software Architecture1-10
Advantages of the Auspex Environment
SunOS Enhancements
Chapter 2 Software Configuration
About This Chapter
New Installations
Installing the Console Terminal
Booting the NetServer
Run <b>NSconfig</b> to Customize Your Server for the First Time
Configuring the Server to Be an NIS Master or Slave Server
Configuring the NetServer to Be an NIS Master
Configuring the NetServer to Be an NIS Slave
NetServers and UDP Checksumming
Modifying Site-Specific Files
Kernel Parameters 2-6
Additional TTYs
Mailing Site Reports to Auspex. 2-7
NetServer Configuration Commands
Using Configuration Command Forms
NSconfig
Before Running NSconfig

Running NSconfig	
SetupTty	2-18
SetupExec	2-20
Do You Need to Run <b>SetupExec</b> ?	2-20
What Does <b>SetupExec</b> Do?	2-21
Things to Remember When Running SetupExec	2-21
Running SetupExec	2-22
SetupClient	
Do You Need to Run SetupClient?	2-28
What Does SetupClient Do?	2-28
Running SetupClient	2-29
Root Login and File System Isolation	2-31
100Base-T Ethernet Full-Duplex	2-39
100Dase 1 Edicinet full Duplex	
Chapter 3 Booting Up and Shutting Down the NetServer	
About This Chapter	3₋1
Booting the NetServer	
Single-User Mode vs. Multiuser Mode	
Accessing the PROM Monitor	
Entering Monitor Mode When the System is Powered Off	
Entering Monitor Mode When the System is Running	
Booting the Server at the Monitor Prompt	
The Default Boot Device	
Using the Boot Command	3-5
Exiting Single-User Mode to Multiuser Mode	3-6
Booting From a CD-ROM	3-6
Rebooting the Server to Multiuser Mode	
DataGuard and Rebooting	3-8
Commands for Shutting Down the NetServer	3-9
shutdown	3-9
halt	3-10
reboot	3-10
fasthalt and fastboot	
Chapter 4 NetServer Storage Devices and File Systems	
About This Chapter	
Device Names	4-2
Disk Drives	4-2
CD-ROM Drives	
Tape Drives	
Raw Disk Partition	
Device Special Files	
Using CD-ROM Drives	
Mounting or Unmounting File Systems on CD-ROM	
Removing a CD-ROM From the NetServer	
Disk Partitioning.	4-7
Disk Partition Tables	
Distribution of File Systems on Default Partitions	
Default File Systems on the Root Drive	
Default File Systems on Default Drives	
Naming File Systems	
Example of File System Distribution	
Auspex File System Structure	
Mounting File Systems on the FD and LID	4-12 1 م
Mounting File Systems on the FP and HP	4-18
File Systems Mounted on the File Processor	
File Systems Mounted on the HP	4-IX



Auspex's Modifications to /etc/fstab	. 4-19
Specifying the File System Type	
Mapping LFS File Systems to File Processors	
Enabling Write Acceleration in /etc/fstab	. 4-22
Other Commands Affected by LFS	. 4-22
File Processor Constraints	. 4-22
File System Formats Supported by UFS	. 4-22
NFS Version 3 and UFS Limitations	4-23
Specifying a Host Name in /etc/fstab	4-24
Adding or Replacing a Drive	4-25
Installing or Replacing a Drive After Booting	
Before You Start the Drive Installation Procedure	
Using ax_hot_plug to Add a Drive to the Server	
Installing or Replacing a Drive When the Operating System is Shut Dov	
Configuring a New Disk Drive	
Removing a Drive	
Removing a Drive When the Operating System is Booted	
Before You Remove a Drive	1_21
Using ax_hot_plug to Remove a Drive	1-31
Removing a Drive When the Operating System is Shut Down	
Using ax_hot_plug to Add and Remove Multiple Drives	
Attaching Drives to the Host Processor SCSI Port	1 25
Using the Format Command	
Running the Format Command in Single- or Multiuser Mode	
Accessing the Format Menu	
Defining a Disk Type	
Formatting a Disk Using the Format Option	4 20
Using ax_label to Partition and Label a Disk	1_11
Displaying a Disk's Partitioning Information	1_19
Displaying a Disk's Landdolling Information	. 4-44
Chapter 5 Virtual Partitions	
About This Chapter	5-1
Virtual Partition Overview	5-2
Maximum Size of a Virtual Partition	
Advantages of Using a Greater-than-2-GB Virtual Partition	
Different Types of Virtual Partitions	5-4
Concatenated Partition	
Striped Partition	
Mirrored Partition	5-5
Virtual Partition Restrictions	
Restrictions Common to All Kinds of Virtual Partitions	5-7
Restrictions Applicable to Concatenated and Striped Partitions	
Restrictions Applicable to Mirrored Partitions	
Virtual Partition Driver, File, and Commands	
Virtual Latition Bilver, Line, and Communities	5-8 5-9
Virtual Partition Driver	5-9
Virtual Partition Driver	5-9
Virtual Partition File	5-9 5-9 5-9
Virtual Partition File	5-9 5-9 5-10
Virtual Partition File	5-9 5-9 5-9 . 5-10 . 5-13
Virtual Partition File	5-9 5-9 5-10 . 5-13
Virtual Partition File Virtual Partition Commands Automatic Error Recovery on Mirrored Partitions Defining and Reconfiguring Virtual Partitions Defining a Virtual Partition	5-9 5-9 5-10 . 5-13 . 5-14
Virtual Partition File Virtual Partition Commands Automatic Error Recovery on Mirrored Partitions Defining and Reconfiguring Virtual Partitions Defining a Virtual Partition Expanding a Concatenated or Striped Partition	5-9 5-9 5-10 . 5-13 . 5-14 . 5-14
Virtual Partition File Virtual Partition Commands Automatic Error Recovery on Mirrored Partitions Defining and Reconfiguring Virtual Partitions Defining a Virtual Partition Expanding a Concatenated or Striped Partition Expanding a Concatenated Partition	5-9 5-9 5-10 . 5-13 . 5-14 . 5-14 . 5-15
Virtual Partition File Virtual Partition Commands Automatic Error Recovery on Mirrored Partitions Defining and Reconfiguring Virtual Partitions Defining a Virtual Partition Expanding a Concatenated or Striped Partition Expanding a Concatenated Partition Expanding a Striped Partition	5-9 5-9 5-10 5-13 5-14 5-15 5-16
Virtual Partition File Virtual Partition Commands Automatic Error Recovery on Mirrored Partitions Defining and Reconfiguring Virtual Partitions Defining a Virtual Partition Expanding a Concatenated or Striped Partition Expanding a Concatenated Partition	5-9 5-9 5-10 5-13 5-14 5-15 5-16 5-16

Expanding a One-Membered Striped Partition 5-19
Expanding a Two-Membered Mirrored Partition 5-19
Changing a Striped or Concatenated Partition to a Mirrored Partition 5-19
Adding a New Member to an Existing Mirrored Partition 5-21
Removing a Member from an Existing Mirrored Partition 5-21
Moving a Striped or Concatenated Partition
Moving a One-Membered Mirrored Partition
Moving a Two-Membered Mirrored Partition With No Unmount 5-22
Displaying Disk Configuration Information
Displaying Dish Comiguration information
Chapter 6 Recovering From Disk or File System Failures
About This Chapter 6-1
When Does a File System Become Unavailable?
Disk Problems at System Reboot
Is the Problem Caused by Poor Disk Installation? 6-4
Is the Problem Caused by a Malfunctioning Drive Slot?
Is the Problem Caused by a Disk Without a Disk Label? 6-4
Replacing a Malfunctioning Disk
Replacing a Drive in Single-User Mode
Entering Multiuser Mode and Replacing the Drive
Using the System Without the Failing Drive
Recoverable Disk Problems in Multiuser Mode
Unrecoverable Disk or File System Problems in Multiuser Mode 6-9
Error Messages Indicating Disk or File System Problems6-9
File System Isolation 6-10
Types of Errors that Cause File System Isolation
What Does the FP Do When an Attempted I/O Fails? 6-12
Repairing an Isolated File System with fsck
Releasing a File System After Repairing It by fsck 6-14
Restoring a File System that Cannot Be Repaired 6-15
Examples of Recovery Procedures After File System Isolation 6-15
File System Isolation Caused by Media Errors That Cannot Be Fixed . 6-15
File System Isolation After Media Errors That Can Be Fixed 6-17
File System Isolation Caused by Corruption 6-18
Recovering from Permanent Disk Errors Without File System Isolation 6-19
Replacing a Disk Containing Unmirrored Partitions 6-19
Replacing a Disk Containing Mirrored Partitions 6-20
Recovering from Intermittent Disk Errors Without File System Isolation . 6-21
Replacing a Drive that Is a One-Member Mirrored Partition 6-22
Replacing a Drive that Has Multiple Partitions
Repairing Disk Sectors After an Unrecovered Media Error 6-23
If Sector 0 (Disk Label) Is Unreadable
If Superblock Is Unreadable 6-24
If Other Disk Blocks Are Unreadable
Recovering a Damaged Root Disk
Booting From a Backup Root Disk
Booting From the CD-ROM in Single-User Mode 6-27
Using NSinstall to Install Software From CD-ROM
Starting NSinstall
NSinstall Form
Executing the NSinstall Form
Zacedding the Promottin Portin
Chapter 7 Write Acceleration
About This Chapter 7-1
Write Acceleration Overview
How the Write Accelerator Improves Throughput



How the SP Maintains Drive Information	. 7-4
Write Accelerator Restrictions	. 7-5
Write Cache States and Commands	. 7-6
Enabling the Write Accelerator	. 7-8
Disabling the Write Accelerator	. 7-9
Managing the Write Accelerator	. 7-9
Recovering From Write Cache (BADCHECKSUM or DIRTY) Errors .	. 7-9
Recovering From a Disk Media Error (DIRTY State)	
Recovering From a Disk Hardware Error	7-10
Recovering From a Drive Configuration Change	7-11
Chapter 8 Backing Up the NetServer	
About This Chapter	. 8-1
Using the Dump and Restore Commands	. 8-2
Using the Dump and Restore Commands	. 8-2
Restore Command	. 8-3
Tape Drive Capacity	
Data Format on 4-mm Tapes	
Tape Drive Capacities	
The Size, Density, and Blocking Factor Options for Dump	. 8-7
Backing Up With a Large Number of Disks	8-10
Backing Up the Root Disk	
Online Backup	8-12
Cloning a Virtual or Physical Partition	8-12
Important Guidelines for Creating a Clone Partition	8-13
Procedure for Cloning a Partition	8-14
Dumping a Clone Partition to Tape	
Restoring a File With a Damaged Inode	8-17
Restoring a File from 1.5.1-produced Tapes	
Common Tape Drive Errors	8-19
Operator Errors	8-19
I/O Error	8-19
Write Error	8-19
Blank Tape Error	8-19
Memory Error	8-20
End-of-tape Error	8-20
Tape Media Errors	8-20
Write Error	8-21
Read Error	
Write Failure Error	
Tape Jam Error	8-21
Tape Drive Errors	
Write Error	8-22
Online Error	8-22
Chapter 9 Measuring the NetServer's Performance	
About This Chapter	. 9-1
Capturing and Displaying Real Time Performance Data	
How ax_perfmon Works	
Starting the Performance Monitor	
Interactive Performance Monitor Commands	
Interpreting Statistics Screens	
Screen Header	
System Summary Screen	
NP Statistics Screen	
NP Protocol Screen	9-10

TID Co. 11 11 C	0.40
FP Statistics Screen	
SP Statistics Screen	
Virtual Partition Statistics Screen	
Using a Filter File with ax_perfmon	
Creating a Filter File	. 9-16
Specifying a Filter File on the ax_perfmon Command Line	
Displaying Performance Data in Histograms	
Overview	
Before Invoking ax_perfhist	
Starting ax_perfhist Interactively	
Loading Statistics From a File	
Clearing a Histogram	. 9-22
Displaying Multiple Histograms	. 9-23
Loading Statistics Using 2 Step Load	. 9-24
Eliminating a Histogram from the Data Display Window	. 9-24
Defining the Appearance of a Histogram	
Saving and Applying a Configuration File	
Examining Samples in Histograms	. 9-32
Menu for Zooming In and Zooming Out	
Summing Parameters in Histograms	. 9-35
Deleting an Input File	. 9-37
Pull-Down Menu Displayed in the Input File List	. 9-37
Printing Histograms	. 9-38
Print Command	. 9-38
Print Command Output	. 9-40
Types of Statistics Displayed by ax_perfhist	. 9-40
HP	. 9-41
NP	
FP	
SP	
Starting ax_perfhist with Options and Arguments	
Analyzing Performance Monitor Data	
Chapter 10 Special Maintenance	
About This Chapter	. 10-1
Configuration Management	. 10-2
Starting ax_config	. 10-4
Interactive Mode	
Displaying Configuration Data	. 10-5
Non-interactive Mode	. 10-5
Displaying Configuration Information	. 10-5
Copying Configuration Information to a File	. 10-6
Flash PROM Download Utility	. 10-7
ax_load_flash	. 10-7
Dealing with Server Problems	. 10-9
Rebuilding the System Kernel	. 10-9
System Panics and Crashes	10-10
Message Logging	10-11
Using the Continue Command	10-12
Remote Diagnostics	10-12
Managing Network Interfaces	10-14
Displaying and Flushing ARP Tables Used by an NP Board	10-14
Displaying Network Statistics for a Network Processor	
Displaying NFS Statistics	
Network Changes	10-19



Index



# **Figures**

Chapter 1	Overview of NetServer Features
Figure 1-1.	Hardware implementation of Functional Multi-processing architecture 1-9
Figure 1-2.	NetServer Functional Multi-processing architecture1-10
O	
Chapter 2	Software Configuration
Figure 2-1.	Flowchart for configuring the server for the first time 2-3
Figure 2-2.	Sample NSconfig form
Figure 2-3.	Flowchart for configuring a TTY port2-18
Figure 2-4.	The SetupTty form
Figure 2-5.	Configure client information on the server using SetupExec 2-20
Figure 2-6.	Sample SetupExec form for a local tape device
Figure 2-7.	Sample SetupExec form for a local CD-ROM device
Figure 2-8.	Sample SetupExec form for a remote tape device
Figure 2-9.	Sample SetupExec form for a remote CD-ROM device
Figure 2-10.	
Figure 2-11.	The SetupClient form
Chapter 3	Booting Up and Shutting Down the NetServer
Chapter 3	booting op and Shutting Down the NetServer
Chapter 4	NetServer Storage Devices and File Systems
Figure 4-1.	Partitioning a 9-GB disk using the default partition table 4-9
Figure 4-2.	Partitioning a 9-GB disk using the stripe partition table 4-9
	Turbusting a company and surperportation tubes continued to
Figure 4-3.	Partitioning a 4-GB disk using the root partition table4-9
Figure 4-3. Figure 4-4.	Partitioning a 4-GB disk using the root partition table
	Partitioning a 4-GB disk using the root partition table4-9
Figure 4-4.	Partitioning a 4-GB disk using the root partition table
Figure 4-4. Figure 4-5.	Partitioning a 4-GB disk using the root partition table
Figure 4-4. Figure 4-5. Figure 4-6.	Partitioning a 4-GB disk using the root partition table
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7.	Partitioning a 4-GB disk using the root partition table
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system 4-13 The Auspex /var file system 4-14 The Auspex /usr file system 4-15 The Auspex /usr/openwin file system 4-16 The /export file system for servers (with SunOS executables) 4-17 The /exportN file system for clients 4-17
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8. Figure 4-9. Figure 4-10.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system 4-13 The Auspex /var file system 4-14 The Auspex /usr file system 4-15 The Auspex /usr/openwin file system 4-16 The /export file system for servers (with SunOS executables) 4-17 The /exportN file system for clients 4-17 Default fstab file 4-20
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8. Figure 4-9. Figure 4-10. Chapter 5	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions Example of a concatenated partition of three physical partitions . 5-4
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2.	Partitioning a 4-GB disk using the root partition table
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system 4-13 The Auspex /var file system 4-14 The Auspex /usr file system 4-15 The Auspex /usr/openwin file system 4-16 The /export file system for servers (with SunOS executables) 4-17 The /exportN file system for clients 4-17 Default fstab file 4-20  Virtual Partitions Example of a concatenated partition of three physical partitions 5-4 Example of a striped partition of four members 5-5 Example of a mirrored partition 5-6
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3. Figure 5-4.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system 4-13 The Auspex /var file system 4-14 The Auspex /usr file system 4-15 The Auspex /usr/openwin file system 4-16 The /export file system for servers (with SunOS executables) 4-17 The /exportN file system for clients 4-17 Default fstab file 4-20  Virtual Partitions Example of a concatenated partition of three physical partitions 5-4 Example of a mirrored partition 5-6 Disk space truncated from members of a striped partition 5-8
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3. Figure 5-4. Figure 5-5.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions  Example of a concatenated partition of three physical partitions . 5-4 Example of a mirrored partition . 5-6 Disk space truncated from members of a striped partition . 5-8 Sample vpartab entries . 5-9
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3. Figure 5-5. Figure 5-6.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions  Example of a concatenated partition of three physical partitions . 5-4 Example of a striped partition of four members . 5-5 Example of a mirrored partition . 5-6 Disk space truncated from members of a striped partition . 5-8 Sample vpartab entries . 5-9 Example from using the ax_vpstat command . 5-12
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-8. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3. Figure 5-5. Figure 5-6. Figure 5-7.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions  Example of a concatenated partition of three physical partitions . 5-4 Example of a striped partition of four members . 5-5 Example of a mirrored partition . 5-6 Disk space truncated from members of a striped partition . 5-8 Sample vpartab entries . 5-9 Example from using the ax_vpstat command . 5-12 Automatic recovery from mirrored partition error . 5-13
Figure 4-4. Figure 4-5. Figure 4-6. Figure 4-7. Figure 4-9. Figure 4-10.  Chapter 5 Figure 5-1. Figure 5-2. Figure 5-3. Figure 5-5. Figure 5-6.	Partitioning a 4-GB disk using the root partition table. 4-9 The Auspex root file system . 4-13 The Auspex /var file system . 4-14 The Auspex /usr file system . 4-15 The Auspex /usr/openwin file system . 4-16 The /export file system for servers (with SunOS executables) . 4-17 The /exportN file system for clients . 4-17 Default fstab file . 4-20  Virtual Partitions  Example of a concatenated partition of three physical partitions . 5-4 Example of a striped partition of four members . 5-5 Example of a mirrored partition . 5-6 Disk space truncated from members of a striped partition . 5-8 Sample vpartab entries . 5-9 Example from using the ax_vpstat command . 5-12



Figure 5-10. Figure 5-11.	Example ax_diskconf display
<b>Chapter 6</b> Figure 6-1.	Recovering From Disk or File System Failures The NSinstall form
<b>Chapter 7</b> Figure 7-1.	Write Acceleration Write cache changing states after server power on
Chapter 8 Figure 8-1. Figure 8-2. Figure 8-3. Figure 8-4.	Backing Up the NetServerBackup scheme for 42 drives.8-10Example of a clone of a mirrored partition.8-141.5.1 tape encoding.8-181.9 tape encoding.8-18
Chapter 9 Figure 9-1. Figure 9-2. Figure 9-3. Figure 9-4. Figure 9-5. Figure 9-6. Figure 9-7. Figure 9-9. Figure 9-10. Figure 9-11. Figure 9-12. Figure 9-13. Figure 9-15. Figure 9-15. Figure 9-16. Figure 9-17. Figure 9-18. Figure 9-19. Figure 9-20. Figure 9-21. Figure 9-22.	Measuring the NetServer's PerformanceSample System Summary screen for an NS 7000/200 Series NetServer.9-7Sample System Summary screen for an NS 7000/600 Series NetServer.9-7Sample of an NP Statistics screen.9-9NP Protocol Statistics screen.9-10Sample FP Statistics screen.9-12Sample SP Statistics screen.9-13Example of a Virtual Partition Statistics screen.9-15Sample output file created by ax_perfmon with a filter file.9-17Data Display Window.9-20Buttons in the Data Display Window.9-21Displaying the Data Display and File Load windows simultaneously.9-21Displaying the Data Display and File Load windows simultaneously.9-22Data Display Window showing two histograms simultaneously.9-23Histogram Properties Window.9-25Effects of Histogram Properties parameters on a histogram.9-27Server statistics plotted on a continuous graph.9-28Overall Properties Window.9-28Configuration Window.9-30Displaying six histograms as specified in a configuration file.9-31Pull-down menu for zooming samples between the markers.9-33Histogram showing samples after zooming.9-34
Figure 9-23.	Summing two histograms
Chapter 10 Figure 10-1.	Special Maintenance How ax_config obtains configuration information10-3
Appendix A	UNIX Manual Pages
Appendix B	Online Documentation
Appendix C	Time Zones
Appendix D	Configuration Worksheets
Appendix E	Messages



## **Tables**

Chapter 1 Table 1-1. Table 1-2. Table 1-3. Chapter 2 Table 2-1.	Overview of NetServer Features         NetServer features       1-3         Changes to SunOS       1-12         Pointers to Sun documentation       1-16         Software Configuration
Table 2-1. Table 2-2. Table 2-3. Table 2-4. Table 2-5. Table 2-6. Table 2-7. Table 2-8. Table 2-9.	Key files that contain site-specific information2-5Keys for editing configuration forms2-10The NSconfig form fields2-14Date and time formats2-17Files modified by NSconfig2-17The SetupTty form fields2-19Sun system architectures2-22The SetupExec form fields2-25The SetupClient form fields2-30
Chapter 3 Table 3-1.	Booting Up and Shutting Down the NetServer Examples of the b command
Chapter 4 Table 4-1. Table 4-2. Table 4-3. Table 4-4. Table 4-5. Table 4-6. Table 4-7. Table 4-8. Table 4-9. Table 4-10.	NetServer Storage Devices and File Systems  CD-ROM naming conventions
<b>Chapter 5</b> Table 5-1.	Virtual Partitions Numbering virtual partitions
<b>Chapter 6</b> Table 6-1.	Recovering From Disk or File System Failures The NSinstall form fields
<b>Chapter 7</b> Table 7-1.	Write Acceleration ax_write_cache command options
<b>Chapter 8</b> Table 8-1.	Backing Up the NetServer Arguments for the dump command8-2



Table 8-2.	Tape drive data capacities8-6
Table 8-3.	Suggested dump and restore arguments8-7
Table 8-4.	Dump and restore command examples8-9
Table 8-5.	Differences between cloning and mirroring a partition8-12
Chapter 9	Measuring the NetServer's Performance
Table 9-1.	NP statistics9-9
Table 9-2.	NP Protocol Statistics
Table 9-3.	Fields in the Histogram Properties Window9-25
Table 9-4.	Fields in the Overall Properties Window9-29
Table 9-5.	Options in pull-down menu in Data Display Window9-35
Table 9-6.	Pull-down menu options displayed in the input file list9-37
Table 9-7.	Differences between xwd and screendump9-40
Table 9-8.	Options and arguments used in the ax_perfhist command
	1 0 -1
Chapter 10	Special Maintenance
Chapter 10 Appendix A	Special Maintenance
•	Special Maintenance UNIX Manual Pages
Appendix A	Special Maintenance UNIX Manual Pages
Appendix A Appendix B Appendix C	Special Maintenance UNIX Manual Pages Online Documentation Time Zones
Appendix A Appendix B Appendix C Table C-1.	Special Maintenance UNIX Manual Pages Online Documentation
Appendix A Appendix B Appendix C Table C-1.	Special Maintenance UNIX Manual Pages Online Documentation Time Zones Time zone listing C-1 Additional time zone listings. C-4
Appendix A Appendix B Appendix C Table C-1. Table C-2.	Special Maintenance UNIX Manual Pages Online Documentation Time Zones Time zone listing
Appendix A Appendix B Appendix C Table C-1. Table C-2. Appendix D Appendix E	Special Maintenance UNIX Manual Pages Online Documentation Time Zones Time zone listing



## **Preface**

## **About This Guide**

This guide gives you detailed information about how to operate and manage an Auspex NetServer. It offers procedures and recommendations on how to run the NetServer software efficiently to ensure reliable and fast file service at your site.

## **Applicable Documentation**

Because Auspex NetServers run the SunOS 4.1.4 operating system and are fully compatible with ONC/NFS, many Sun Microsystems documents are relevant to NetServer operations. This guide documents differences from SunOS and does not attempt to document SunOS or system administration in detail. For Auspex hardware information or SunOS administration information, refer to the appropriate manual in the following list:

- ▲ NS 7000 Model 150/250 Series Hardware Manual, Auspex Systems, Inc.
- ▲ NS 7000 Model 650 Series Hardware Manual, Auspex Systems, Inc.
- ▲ NS 7000 Model 700 Hardware Manual, Auspex Systems, Inc.
- ▲ NS 7000 Model 800 Hardware Manual, Auspex Systems, Inc.
- Storage Peripherals Manager's Guide, Auspex Systems, Inc.
- ▲ Version 1.10 Hardware Release Note, Auspex Systems, Inc.
- ▲ Version 1.10 Software Release Note, Auspex Systems, Inc.
- ▲ System and Network Administration, Sun Microsystems, Inc.

## **Terminology**

Several models of Auspex NetServers run NetServer version 1.10 software: the 200 series, 500 series, 600 series, 700 series, and 800 series. The NS 6000<sup>™</sup> NetServer is also supported, but must be upgraded. Version 1.10 software does not support:

- Host Processor IV (HP IV) or earlier boards.
- Ethernet Processor (EP) boards.
- ▲ HP V-based FDDI Processor (FDDIP) boards.
- ▲ Auspex Primary Memory (APM).



- ▲ File Processor (FP) boards.
- ▲ Storage Processor I (SP I), Storage Processor II (SP II), or Storage Processor III (SP III) boards. SP III boards can be upgraded to supported SP III-E boards.

In this guide, where information applies to all Auspex network servers regardless of model or Host Processor type, the text uses the generic term *NetServer*. Where information applies to a particular model or Host Processor type, the text uses the specific model number or Host Processor type.

## **Typographical Conventions**

In this guide, different typefaces indicate different kinds of information. The following table explains these typographical conventions.

Font	Meaning
Typewriter	Indicates a literal screen message.
Bold	In a command line, indicates information to be entered exactly as shown. In text, indicates a command name or device name.
Italics	In a command line, indicates a nonliteral item or variable for which you substitute an appropriate value. In text, indicates a path name or a special term.

Hexadecimal values in the text are preceded with "0x," and leading zeros are not always shown. For example, the notation 0x68 is used to indicate the hexadecimal address 00000068.



## **Special Messages**

The following special messages are used in this guide:



**Warning:** Warnings alert you to the danger of personal injury and call attention to instructions you must follow for your personal safety.



**Caution:** Cautions call attention to instructions you must follow to prevent damage to system hardware or software or loss of system data.



**Note:** Notes call attention to important information you should be aware of as you follow the procedures described in this guide.



**Recommendation:** Recommendations call attention to an item or procedure that is not required but might help improve performance, ease of use, and ease of installation or configuration.



#### **Tools**

The tools icon identifies the tools you need to complete a task.

## **Terminology**

Throughout this manual, certain terms refer to NetServer families generically. Where distinctions are necessary, the actual machine model is used.

The generic term	Describes these models
NS 7000/800	NS 7000 Model 800
NS 7000/700	NS 7000 Model 700, 710, and 720
NS 7000/600	NS 7000 Model 600, 602, 610, 612, and 650
NS 7000/200	NS 7000 Model 200, 210, 220, and 250
NS 7000/150	NS 7000 Model 150
NetServer or server	All models

Host Processor PROM monitor prompts are represented with "HP>" throughout this manual. Since this system prompt varies depending on the Host Processor model installed in the NetServer, the prompt that appears on your system might be different.



## **Getting Help**

Customer support and online help are described in this section.

## **Email Support**

Customer service and help through electronic mail are available to North American and International customers at the following address:

support@auspex.com

## **North America Telephone Support**

To reach Auspex customer service for North America at any time, dial 1-800-328-7739.

## **International Telephone Support**

Customers within the countries and territory listed as follows can reach Auspex customer service at any time by dialing the appropriate telephone number.

Country or territory	Telephone number	Country or territory	Telephone number
Australia	1-800-121-194	Luxembourg	0800-2721
Austria	0-660-7912	Malaysia	800-4509
Belgium	0800-1-0180	Netherlands	06-0222158
Denmark	80-01-03-60	Norway	800-1-1294
France	0800-91-21-32 (0800-26-38-22)*	Philippines	1-800-116-0005
Germany	0130-81-8306	Poland	00-800-441-1220
Hong Kong	800-4803	Spain	900-95-4445
Ireland	1800 55 3343	Sweden	020-797326
Israel	177 440 9285	Switzerland	0800-55-1339
Italy	1678-73541	Taiwan	0080-14-9580
Japan	0044-22-12-2414	United Kingdom	0800-28-7739
Korea	0038-13-0374		

<sup>\*</sup> France has two toll-free numbers. The number in () is directed to the Paris office during the working day.

International customers who purchase Auspex products from an authorized distributor or reseller should contact that distributor or reseller for support

To comment on the content of this online help facility, send email to Auspex Technical **Publications:** 

techpubs@auspex.com



#### World Wide Web

For more information about Auspex Systems, Inc. and its products, use the following resource location on the World Wide Web:

http://www.auspex.com

### **Printing Online Documents**

You can print individual pages or the entire contents of the online manual. To print individual pages, use the **ax\_docs** command and choose Premier Software Series Menu in the Main Menu window. Choose the <opf product> documentation, and pull down the File menu when the document appears. The Print option allows you to print a particular page, a range of pages, or the entire document. The Print Setup option allows you to select a paper size, including A4.

Alternately, to print the whole <opf product> document, mount the Premier Software Series CD, and use the lpr -s command to print 100btatmbackupdguardftpraid.ps in the /psdocs directory.



# **Overview of NetServer** 1 Features

## **About This Chapter**

This chapter provides an overview of the NetServer™ and describes the features specific to the NetServer's hardware and software architectures. In addition, this chapter points out the distinct advantages such as balanced client traffic, ease of administration, and scalability offered by the Auspex environment. The last section lists changes to SunOS Version 4.1.4 that Auspex has made in the NetServer software.



## **Product Overview**

The Auspex NetServer is specifically designed to deliver industry-leading performance and reliability that meets the demands of large networks in real-world production environments. With its unique Functional Multi-processing® (FMP®) architecture, the NetServer delivers high-speed network I/O performance. The NetServer is also highly scalable—you can improve its performance by adding processor boards and storage devices as the number of network clients grows. In addition, the NetServer's software, which is based on the Sun Microsystems SunOS 4.1.4, offers Auspex-specific system administration commands for minimizing system down time. Procedures such as drive replacement and file system backup can be completed while the server is in multiuser mode.

All models of NetServers have the features described in Table 1-1.



#### Table 1-1. NetServer features

System architecture

- Functional Multi-processing (FMP) architecture with separate dedicated processors for network, file, UNIX, and storage processing.
- ▲ Host Processor (HP VII and HP VIII), which incorporates a Ross Technology HyperSPARC Mbus module running at 125 MHz. The HP offers full compatibility with SPARC Compliance Definition 1.1.
- ▲ WarmStart: Upon booting, the system automatically bypasses certain diagnostic tests, reducing the time required to return to service.
- ▲ 100Base-T Ethernet full-duplex software is an integral part of the system software, ready for configuration and use with no additional installation after system software installation. 100Base-T Ethernet runs in full-duplex mode with the appropriate adapter card.
- Optional nonvolatile write cache for improved Network File System (NFS) write operation performance.
- ▲ Optional ServerGuard<sup>™</sup> software, which allows mirroring of an NFS filesystem across a pair of NetServers. One NetServer acts as the primary server, and the other, which may be physically remote, acts as the secondary server. With ServerGuard, the NFS filesystem is independent of local power failures and natural disasters, resulting in uninterrupted NFS service to clients.
- ▲ Optional DriveGuard<sup>™</sup> software (RAID 5 implementation), which protects user data in the event of a single device failure. If any single disk in a DriveGuard array fails, the data can be reconstructed using information on the remaining disks in the array. See the Auspex DriveGuard documentation on the Auspex Premier Software Series CD-ROM for more information.
- ▲ Optional DataGuard™ software, allowing the HP to reboot without affecting other system components or NFS service to clients. This feature improves overall reliability of the NetServer by isolating any failures to the HP and services directly provided by it.
- ▲ Optional EtherChannel combines multiple 100Base-T Ethernet interfaces into a single channel. All the interfaces in the EtherChannel load balance the traffic among themselves, giving the concept of one fat pipe.
- Optional NP (ATM Release 2) software, is Auspex's Network Processor (NP)-resident implementation of Asynchronous Transfer Mode (ATM). It includes LAN Emulation (LANE) for Ethernet clients, and allows you to configure three types of network interfaces: FORE IP, Classical IP, and LEC.
- Optional NeTservices software, a UNIX-based network operating system, is fully compatible with Microsoft networking technology. NeTservices provides file services for the efficient sharing of computing resources among a community of desktop users.



#### Table 1-1. NetServer features (Continued)

Storage subsystems	▲ High-performance storage subsystems: Drives are organized in racks or drawers of seven drives. The maximum number of drives supported depends on the NetServer model. Software commands allow a Storage Processor to pause and restart if you need to hot plug drives (that is, remove or insert drives while the NetServer is running).		
	▲ Auspex Virtual Partition Manager: Mirrored, striped, and concatenated partitions provide an efficient and reliable way of using the disk space in a NetServer. You can enlarge a striped or concatenated virtual partition without interrupting user access to the file system on the virtual partition.		
	■ Very Large File (VLF) support: The size of an individual file is limited only by file system size for LFS files accessable over NFSv3. Both file and file system sizes may be over 2 GB. Clients accessing such files must also have VLF capabilities.		
Auspex system management tools	▲ The Auspex Performance Monitor, which displays performance data for all processors in the system as well as disk subsystems and nonvolatile RAM (NVRAM) used for write caching.		
	▲ The Auspex Virtual Partition Manager, which comprises commands that display and manage the configuration of all virtual partitions.		
	■ Online hardware and software information in Adobe <sup>™</sup> PDF format and an Acrobat reader utility to read the files.		
	For a more detailed list of Auspex-specific commands and utilities, refer to "SunOS Enhancements" on page 1-12.		
Compatibility with industry standards	Compatible with UNIX (SunOS), ONC/NFS, TCP/IP, Ethernet, VME, SCSI, X11 (OpenWindows), and FDDI standards. Compliant with POSIX 1003.1, FIPS 151-a, SVID Issue 2, and X/Open Portability Guide Version 2.		
Distribution medium	CD-ROM.		

For hardware features such as the maximum number of network interfaces supported, amount of I/O cache memory, and storage capacity, refer to the hardware manual appropriate to your NetServer model.



## **NetServer Hardware Architecture**

The combination of processor boards and drive subsystems within a NetServer varies from one configuration to another. However, on all NetServer models, the Auspex FMP architecture distributes performance-limiting I/O functions to multiple dedicated processors, bypassing the UNIX operating system to significantly reduce software overhead. As a result, Auspex's FMP architecture delivers a dramatic improvement in network I/O performance.

An example of a processor board is the Network Processor (NP). An NP board contains one CPU for protocol processing and another for file system processing. When this guide uses a processor name (for example, "Network Processor"), it refers to the CPU that performs a specific processing function, whether or not the CPU shares a board with another CPU.

The following lists the major hardware subsystems in the NetServer:

- FMP processors (Host, Network, and Storage Processors)
- High-performance disk subsystem
- CD-ROM and tape drives



**Note:** NetServer Version 1.10 software supports only servers with HP VII and HP VIII, NP III and NP IV, and SP IV and SP V boards.

As more users join the network, processor and storage capacity can be added to accommodate the increased work load.

The following sections discuss the FMP processors, the shared cache memory, and the Enhanced VME Bus over which the processors communicate with each other.

## **Host Processor (HP)**

The *Host Processor* performs these functions:

- Booting the system.
- Downloading software to the other processors.
- Running UNIX (SunOS) and applications.
- Exporting the standard Sun ONC services.
- Running the NetServer system utilities and diagnostics.

The HP manages NFS Version 2 services for requests to file systems mounted on the HP. Most NFS requests for Auspex Local File Systems (LFS) are processed by the NP.

The NetServer's SPARC-based HP is binary-compatible with SPARC application environments and provides a compute capability independent of NFS loading.



**Caution:** The architecture of the HP VIII is **aushp**. Loading kernel modules compiled for an architecture different from that of the HP might cause the server to crash.

The HP VIII is configurable from 128 to 384 MB of memory, with 16-MB, 32-MB, 64-MB, or 128-MB add-on modules. Memory modules must be installed in pairs of equal size.



### **Network Processor (NP)**

The Network Processor combines network-processing and file-processing functions, as well as onboard cache memory. An NP may have multiple, mixed network connections including Ethernet and FDDI connections. An FDDI NP can support either fiber or MLT-3 interfaces, and can be either a Single Attach Station (SAS) or a Dual Attach Station (DAS). (The DAS connection supports fiber only.)

NP network interface adapter boards supporting ATM, FDDI, and Fast Ethernet (100Base-T) interfaces are available as optional products.

For a complete list of supported network interface combinations, see the "Network Processor Configurations" section of the Version 1.10 Hardware Release Note.

## **Storage Processor (SP)**

The Storage Processor operates multiple parallel SCSI channels. The SP V supports six 10-MB per second fast SCSI channels. The SP is responsible for SCSI channel management, SCSI drive control and optimization, DMA data transfers to the primary I/O cache memory, and virtual partition management.

Optionally, the SP supports write acceleration for NFS write operations through write caching, write coalescing, and redundant write elimination. Write acceleration can improve client NFS response time and NetServer throughput by reducing the response time involved in writing data to disk. The write cache on an SP V with the optional Write Accelerator III and current PROM Flash code contains 8 MB of NVRAM with battery backup to protect data when a power failure occurs.

## I/O Cache Memory

The shared *I/O cache memory*, which resides on all NP boards, is used as a large disk cache. No processor instructions are stored in or fetched from this memory. Memory and backplane bandwidth are devoted exclusively to I/O.

#### **Enhanced VME Bus**

The Enhanced VME Bus implementation provides 100-MB per second block transfers between the NP and SP boards.



## **Optional Products**

Auspex provides a system for installing optional software products on the NetServer. Software packages are immediately accessible through the acquisition of a decrypting key and license from Auspex. Optional product packages are located on the Auspex Premier Software Series CD-ROM. The packages must be installed with **pkgadd**(1M) onto the NetServer system software to run. Installation instructions are in the CD-ROM booklet and in the Version 1.10 Software Release Note.

Documentation for all the software packages is provided on the Auspex Premier Software Series CD-ROM. Installation and printing instructions for documentation are in the CD-ROM booklet and in Appendix B of this manual.

The following optional software products are available at the time of printing this manual. Refer to the Auspex Premier Software Series CD-ROM for the latest information.

#### ServerGuard

ServerGuard allows mirroring of an NFS filesystem across a pair of NetServers. One NetServer acts as the primary server, and the other, which may be physically remote (anywhere in the world), acts as the secondary server. With ServerGuard, the NFS filesystem is independent of local power failures and natural disasters, resulting in uninterrupted NFS service to clients.

#### **DataGuard**

DataGuard permits HP only reboots. This confines operating system failures to the HP and any services directly provided by it. DataGuard makes the NetServer more reliable by reducing the impact of UNIX operating system failures and improving the HP reboot time.

#### **DriveGuard**

DriveGuard (RAID 5 implementation) protects user data in the event of a single device failure. If any single disk in a DriveGuard array fails, the data can be reconstructed using information on the remaining disks in the array. See the DriveGuard documentation on the Auspex Premier Software Series CD-ROM for more information.

#### **NeTservices**

*NeTservices* is a UNIX-based network operating system that is fully compatible with Microsoft networking technology. NeTservices provides file services for the efficient sharing of computing resources among a community of desktop users. It incorporates the latest Microsoft technology to deliver powerful new network administration and enhanced security features. Clients using Windows NT Server, Windows NT Workstation, and Windows 95 get e the ability to access UNIX-based file systems on Auspex NetServers. An Auspex NetServer using NeTservices can act as a Primary Domain Controller (PDC) or Backup Domain Controller (BDC) in a Windows NT domain.



#### **EtherChannel**

Ether Channel combines multiple 100Base-T Ethernet interfaces into a single channel called a fat pipe. All the interfaces in the EtherChannel load balance the traffic among themselves, giving the concept of one fat pipe.

The Auspex EtherChannel implementation takes care of traffic directed to a router through the same port, since EtherChannel does the load balancing based on IP addresses, MAC addresses, or with round robin algorithm. The Auspex EtherChannel is a software-only product. When the packet is ready to go out on the net, EtherChannel makes a decision at the driver level about which port to use.

#### **ATM**

NP (ATM Release 2) provides asynchronous communication from the NP board with data transfer rates up to 155 MB per second. NP (ATM Release 2) includes LAN Emulation (LANE) for Ethernet clients, and allows you to configure three types of network interfaces: FORE IP, Classical IP, and LEC.

#### FTP on the Network Processor

FTP on the Network Processor improves File Transfer Protocol (FTP) transfer rates between NetServer disk storage and a network. Data can flow directly from the SP to an NP, bypassing the HP. FTP (NP) handles all the Internet Transmission Control Protocol processing.

## **FastBackup**

FastBackup increases the speed of Auspex systems doing full-system backups and restores. With FastBackup, backup and restore operations take place entirely between SP boards and the backup devices. Up to two backup devices can be added for each SP board. The backup device can be a tape drive on the NetServer, a jukebox, or tape stacker, allowing the system administrator to start the process and then let the process run without further intervention.



## Hardware Implementation of the FMP Architecture

Figure 1-1 illustrates the NetServer hardware architecture. Remember that because you can mix and match some of the processor boards in one system, the configuration of your server may be different from the one shown. Also, there are restrictions on the maximum number of boards that can be installed in a system, depending on the board combination.

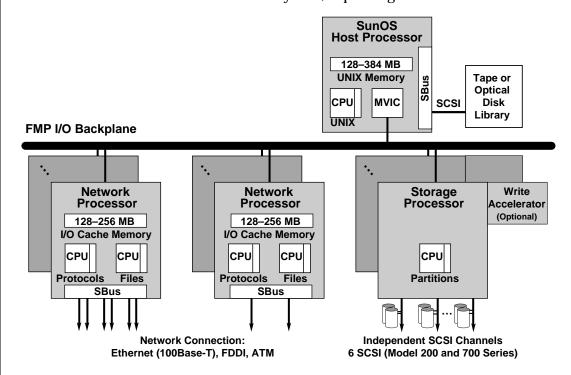


Figure 1-1. Hardware implementation of Functional Multi-processing architecture



## **NetServer Software Architecture**

Figure 1-2 illustrates the software structure of the NetServer. Notice that the NetServer's Functional Multiprocessing Kernel<sup>®</sup> (FMK<sup>®</sup>) executes on each Auspex processor.

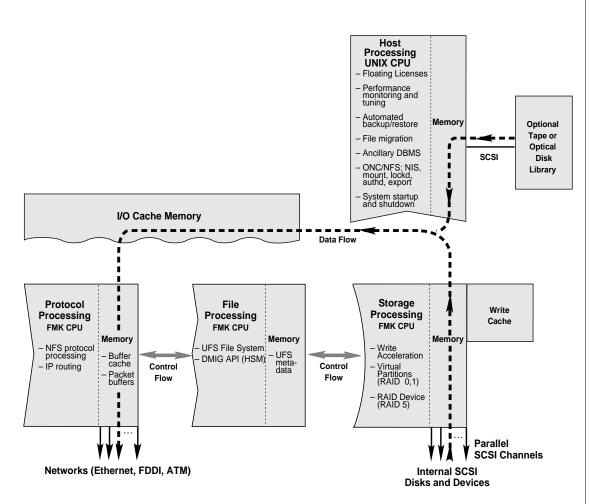


Figure 1-2. NetServer Functional Multi-processing architecture



## Advantages of the Auspex Environment

The Auspex environment is distinguished by its multiple processors, drives, and networks centralized in a single server. This feature provides several system management advantages:

Consolidation

The NetServer's high-performance storage subsystems reduce the need for maintaining and backing up multiple copies of files and programs spread across multiple conventional servers.

Balanced client traffic

You can distribute client traffic more evenly for these reasons:

- Client data can be distributed among disk drives to provide maximum I/O efficiency.
- ▲ The virtual partition capability allows you to distribute logical disk partitions over multiple physical disk drives, thus improving file access times to the file system mounted on the partition. This feature provides a dramatic performance improvement on heavily used file systems.
- The client load can be balanced among multiple networks. Distributing traffic on several networks reduces delays and increases efficiency.

**Expandability** 

If the number of clients outgrows the server's current capacity, the server can be expanded to support more clients or networks. The NetServer's performance increases proportionately with the increased number of drives and functional processors supported by the server.

**Availability** 

The NetServer offers uninterrupted service to network users when performing system administration procedures such as backing up file systems; adding, removing, or replacing drives; and reconfiguring or enlarging a virtual partition. The file system isolation capability ensures that a corrupted file system does not affect the operation of the entire server; the File Processor takes the file system offline so that you can repair it and make it available to users again.

Backup speed

Backup operations can take place entirely between SP boards and the backup devices, giving increased speed when backing up large amounts of data. Backup devices can be tape drives, tape jukeboxes, or tape stackers.

In administering the NetServer, be aware that some standard administrative practices are modified to take into account the number of devices and networks that exist in the centralized NetServer environment. Some procedures are altered to reflect the unique features of the NetServer environment. However, system administration procedures in many respects remain identical to those used for the SunOS operating system by Sun Microsystems, on which the Auspex operating system is based. Refer to the following section, "SunOS Enhancements", for information about Auspex changes to SunOS.



## **SunOS Enhancements**

Each NetServer runs a modified SunOS kernel that communicates with the intelligent Auspex processors to enhance NetServer performance. Most UNIX services come directly from SunOS for complete compatibility. However, Auspex makes some changes to the Sun operating system to integrate the Auspex processors into the standard operating system.



**Note:** The majority of the SunOS standard utilities and files remain unchanged. This publication documents the changed portions of SunOS only; it does not document the unchanged portions. Consequently, it is very important that you have the SunOS documentation. To obtain the latest documentation, contact Sun Express at 1-800-USE-SUNX, or Auspex.

The most significant changes to SunOS are listed in Table 1-2. Not all of the changes are described in the rest of this guide because some changes (for example, the ax\_statd(8) daemon) are transparent to the system administrator. For information about the more visible changes, see the appropriate man page.

Table 1-2. Changes to SunOS

Types of changes	Description	For further information, see	
Added or modified daemons	Three added daemons coordinate the processors: ax_startup(8), ax_errd(8), and ax_timed(8). These daemons, like init(8), must run even in single-user mode.	Man pages for: ax_startup ax_errd ax_timed ax_statd	
	Additional daemons supporting the NetServer in multiuser mode are ax_statd(8), ax_statd2(8), ax_keyenvoyd(8), ax_lfsd(8), ax_nfsd(8), and ax_vold(8).	ax_statd2 ax_keyenvoyd ax_lfsd ax_nfsd ax_vold	
Networking commands and daemons	The <b>nfsstat</b> (8C) command is modified to obtain statistics for the NP and UNIX. Also, an added option (-I <i>interface</i> ) displays NFS and RPC statistics about a network interface.	Man pages for: nfsstat ax_netstat stopnfsd ax arp	
	A new command, <b>ax_netstat</b> (8C), displays NP network statistics.	ax_fddistat	
	A new command, <b>stopnfsd</b> (8), shuts down the NFS daemons on the NPs.		
	A new command, <b>ax_arp</b> (8C), displays or flushes the Address Resolution Protocol (ARP) table on an NP.		
	A new command, <b>ax_fddistat</b> (8C), displays status of FDDI and MLT-3 network interfaces.		
Performance monitor commands	The Auspex Performance Monitor command, ax_perfmon(8), gives the system manager the ability to gather and display NetServer performance data.	Chapter 9	
	The <b>ax_perfhist</b> (8) command displays server performance data in histograms.		



Table 1-2. Changes to SunOS (Continued)

Types of changes	Description	For further information, see
Commands for virtual partitions	A disk management tool, <b>ax_diskconf</b> (8), displays NetServer disk configuration data, including information about virtual partitions.	"Virtual Partition Commands" on page 5-10
	<b>vp</b> (4) is the virtual partition device driver.	"Virtual Partition File" on page 5-9
	vpartab(5) is the virtual partition table.	"Expanding a One-
	<pre>ax_loadvpar(8) loads the partition table.</pre>	Membered Mirrored Partition" on page 5-18
	<pre>ax_vold(8) is a daemon that manages the state of virtual partitions.</pre>	vp man page
	<pre>ax_vpstat(8) displays the status of virtual partitions.</pre>	
	<b>ax_mattach</b> (8) attaches a new member to a mirrored partition.	
	<pre>ax_mdetach(8) detaches a member from a mirrored partition.</pre>	
	<b>ax_mconvert</b> (8) converts a striped or concatenated virtual partition to a one-membered mirrored partition.	
	<pre>ax_mrestore(8) restores mirrored partitions to an active state.</pre>	
Commands for Auspex devices	The device <b>ad</b> (for Auspex disk and CD-ROM) was added to the device options that may be used with <b>boot</b> (8S).	"Device Names" on page 4-2
	The command <b>MAKEDEV.auspex</b> defines Auspex-specific devices.	"Using the Format Command"on page 4-36 for <b>format</b>
	The <b>dkinfo</b> (8) and <b>format</b> (8) commands are modified to support Auspex disks; <b>eject</b> (8) supports <b>acd</b> (for Auspex CD-ROMs). Also, <b>elected</b> (8) partitions and	"Using CD-ROM Drives"on page 4-5 for eject
	labels a drive, and <b>ax_lslabel</b> (8) shows a disk's label and partitioning.	"Adding or Replacing
	The ax_hot_plug(8) command provides the ability to hot plug drives, that is, the ability to add, replace, or remove disk, tape, or CD-ROM drives while the operating system is running. Removing and adding drives can be done from a single command line. Two additional device commands, ax_add_device(8) and ax_remove_device(8), also provide hot-plug capabilities.	Drive" on page 4-25  Man pages for:  MAKEDEV.auspex dkinfo ax_Islabel ax_label
Naming conventions for storage devices	Auspex disk drives are named <b>ad</b> <i>n</i> . Tape drives are named <b>rast</b> <i>n</i> or <b>nrast</b> <i>n</i> . CD-ROM drives are named <b>acd</b> <i>n</i> . ( <i>n</i> is the slot where the drive is installed.)	"Device Names" on page 4-2
	DriveGuard arrays are named <b>ard</b> <i>n</i> , where <i>n</i> is the number of the array.	



Table 1-2. Changes to SunOS (Continued)

Types of changes	Description	For further information, see
File systems	An added file system type, LFS, acts as an interface between the operating system and file systems mounted on the File Processor. In general, LFS remains transparent.	"Mounting File Systems on the FP and HP" on page 4-18
	The NetServer uses the standard 4.2 file system type for its / (root), /usr, and /var file systems.	"Maximum Size of a Virtual Partition" on page 5-3
	exports(5) is modified to add an option allowing asynchronous writes to exported file systems.  Many other standard utilities, including df and du, are modified to support LFS and 72-GB file system	"Online Backup" on page 8-12
	capacity.	"File System Isolation" on page 6-10
	A file system on a virtual partition can exceed 2 GB. Up to 72-GB file systems can be created on an Extended Virtual Partition (EVP).	"Repairing an Isolated File System with fsck" on page 6-13
	ax_clonefs(8) creates a clone partition of a specified physical or virtual partition to facilitate online backups.	"Restoring a File System that Cannot Be Repaired" on page 6-15
	When the server detects a corrupted LFS file system, the File Processor isolates the file system so it does not affect the operation of the server. When you are finished repairing the file system, use <b>ax_fsutil</b> (8) to put it back on line.	Man pages for: mount exports ax_expand
	The <b>ax_expand</b> (8) command enlarges a file system on a concatenated or striped virtual partition.	
	<b>fsck</b> (8) allows you to repair a file system even when it is mounted, provided the File Processor has taken the file system offline.	
	ax_kill(8) kills all processes sleeping on an isolated file system. It is useful when the isolated file system can not be repaired by <b>fsck</b> and added back online. Once the processes are killed, you can unmount and replace the bad disk that caused the file system isolation.	



Table 1-2. Changes to SunOS (Continued)

Types of changes	Description	For further information, see
System administration tools	Upon a system crash, the <b>getcores.sh</b> (8) command compresses and <i>tars</i> all core files for delivery to Auspex.	Man pages for: getcores.sh reporter.sh corescreen.sh
	When enabled, the <b>reporter.sh</b> (8) command prepares and mails site reports to Auspex Customer Service.	mailmessages.sh ax_admmsg ax_config
	When enabled, and in the event of a system crash, the <b>corescreen.sh</b> (8) command prepares and mails a subset of Processor board cores to Auspex Customer Service.	"Flash PROM Download Utility" on page 10-7
	When enabled, <b>mailmessages.sh</b> (8) prepares and mails reboot messages to Auspex Customer Service.	
	The <b>ax_admmsg</b> (8) command searches for Auspex error messages and generates an error report.	
	The <b>ax_config</b> (8) command accesses configuration date and generates a configuration report.	
	The <b>ax_load_flash</b> (8) command downloads Flash PROM firmware to specified processor boards.	
Other commands unique to Auspex systems	<b>rc</b> (8) is modified to download and start the Auspex processors.	Man pages for: rc shutdown
	The ax_write_cache(8) command provides the	rwalld
	system manager with a tool for managing and controlling the write cache. To use the write cache feature, you must install the Auspex Write Accelerator on the Storage Processor.	Chapter 7, "Write Acceleration"
	dump(8) is modified to allow you to specify the time and file system name to be recorded in /etc/dumpdates. This change ensures that the	"Dumping a Clone Partition to Tape" on page 8-15
	appropriate information is recorded in /etc/dumpdates when you dump a partition created by <b>ax_clonefs</b> .	"halt", "reboot" and "fasthalt and fastboot" on page 3-10
	halt(8) and reboot(8) are modified to include the -s option, which causes the system not to run sync(2) at reboot. With this change, reboot can complete even if the NetServer has a file system mounted from an unreachable machine. halt(8), reboot(8), and fastboot(8) now include the -c option, which instructs the system to perform a cold boot in which all diagnostics are performed.	
	<b>shutdown</b> (8) is modified with the <b>-I</b> option to send warning messages only to logged-in users. The option speeds up the shutdown process if the list of machines mounting files is not up to date, or if machines mounting files do not have <b>rwalld</b> (8C) support.	

Table 1-2. Changes to SunOS (Continued)

Types of changes	Description	For further information, see
NeTservices commands	The <b>asu</b> (8) command allows an administrator to manually stop NeTservices, start NeTservices, or reset the daemons with a restart.	NeTservices Product Guide, NeTservices Administration Guide, and NeTservices
	stop shuts down NeTservices on all interfaces.	Concept and Planning Guide
	start enables and starts NeTservices on all interfaces.	
	restart stops NeTservices on all interfaces, then immediately starts NeTservices again. This option is useful as a troubleshooting tool. If NeTservices is not functioning correctly, a restart may clear up problems.	
	<b>help</b> gives information about using the <b>asu</b> command.	

For a complete list of files and programs added or modified in SunOS, see the **auspex**(8) man page, or Appendix A. To see a printed version of the man pages, refer to the *Command Reference Guide*.

For each of the system management procedures described in this guide, the Sun Microsystems documentation can provide additional information. Table 1-3 lists basic system management topics and indicates which chapter to consult in Sun's *System and Network Administration* manual.

Table 1-3. Pointers to Sun documentation

Topic	Sun chapter number
System administrator's role	1
Overview of SunOS	2
Booting and shutdown	4
File maintenance	5
System backup and restore	5
Maintaining disks with format	6
Monitoring file system usage	6
Administering security	7
Reconfiguring the system kernel	10
System accounting	11
fsck	12
Adding a modem	13
Line printer	13
Modifying termcap	14
System crash and accompanying messages	16
Description of networking concepts	17
The Sun network environment	17



Table 1-3. Pointers to Sun documentation (Continued)

Topic	Sun chapter number
NFS	18
The Name Information Service (NIS)	20
Domain Name Service	21
Electronic mail and communications	23
UUCP	22
sendmail	23
Monitor and boot program error messages	Appendix E



# 2 Software Configuration

# **About This Chapter**

This chapter describes how to configure your server for your environment. It assumes that the Version 1.10 root drive is already installed. If you are upgrading your system, refer to the Version 1.10 Software Release Note for information on updating your root drive to Version 1.10 before proceeding.

This chapter includes the following major topics:

- Booting the server, setting up the console, and changing some system files on a newly installed system.
- Overview of configuration commands and description of cursor control within configuration forms.
- Configuring the NetServer software for your network environment. The configuration procedures are required after you install a new NetServer or when you change the server configuration.

Software configuration described in this chapter falls into two categories:

Server-specific customization

Specifying local time zone; defining server and interface names and IP addresses; setting up services such as NIS, routing, and configuring TTY ports. The Auspex commands for server-specific customization are **NSconfig** and **SetupTty**.

Client-specific customization

Installing architecture-dependent SunOS executables for client workstations that boot from the NetServer, and notifying the NetServer which client workstations can boot from the NetServer. The commands for client-specific customization are **SetupExec** and SetupClient.

Although any NFS client can access data on a NetServer, only diskless SunOS clients can boot from the NetServer.



## **New Installations**

The following procedure outlines the steps for configuring your NetServer after you install the server for the first time with **NSinstall**(8) or **NSupdate**(8). See the *Version 1.10 Software Release Note* for software installation instructions. The following sections provide details about each configuration step.

- 1. Install your console terminal.
- 2. Boot the server to single-user mode.
- 3. Run NSconfig, and reboot the server.
- 4. Change the root password.
- 5. If appropriate, configure the server as an NIS master or slave in multiuser mode.
- 6. Modify site-specific files as necessary and reboot the server if needed.
- 7. Run **SetupTty**.
- 8. Run SetupExec.
- 9. Run SetupClient.

## **Installing the Console Terminal**

With each NetServer, Auspex provides a DEC VT510 ANSI-compatible ASCII terminal as a system console. You can also use another model ANSI-compatible terminal, such as a Link terminal, as well as an X terminal or UNIX workstation. Using an X terminal or workstation allows you to view several windows simultaneously and invoke the online NetServer documentation. The exact procedure for setting up the console depends on the terminal type; the following procedure provides some general guidelines for setting up a console terminal.

- 1. Start with the NetServer powered off.
- 2. Set up the console terminal hardware, and attach the terminal to the server's console port. (Refer to your NetServer's hardware manual for more information.)
- 3. Set up the console terminal serial parameters. (Refer to your NetServer's hardware manual for more information.)

## **Booting the NetServer**

This section gives you enough information about booting the server so you can get started with the server configuration commands. For more information on booting, refer to "Booting the NetServer" on page 3-2.

- 1. After you power on the server, it displays the boot messages on the console screen and boots in multiuser mode by default. Refer to the NetServer's hardware manual for information on how to power on the server.
- 2. When the server displays the login prompt, log in as root. Press Return when it prompts you for the password.
- 3. After you have logged in, enter the **shutdown** command without arguments to go to single-user mode.



## Run NSconfig to Customize Your Server for the First Time

In single-user mode, run **NSconfig** to configure your server. For overall information on server configuration, refer to "NetServer Configuration Commands" on page 2-9; for information on the **NSconfig** command, refer to "NSconfig" on page 2-10.

After you are finished with **NSconfig**, continue with the procedure in "Changing the Root Password" on page 2-3. Figure 2-1 outlines the procedures after you run **NSconfig**.

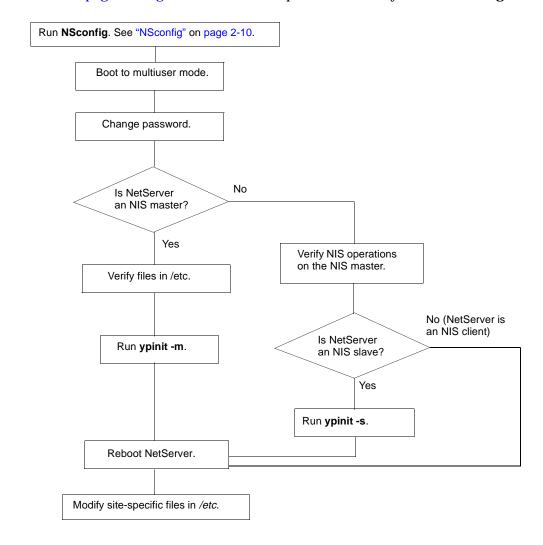


Figure 2-1. Flowchart for configuring the server for the first time

## **Changing the Root Password**

After running **NSconfig** and exiting to multiuser mode, enter the **passwd root** command to create a root password. This step is very important for system security.



## Configuring the Server to Be an NIS Master or Slave Server

If you do not use the NIS name service at your site, skip to "Modifying Site-Specific Files" on page 2-5.

### Configuring the NetServer to Be an NIS Master

If you use the server as an NIS master, follow these steps:

- 1. Verify that the current versions of the NIS-related files (/etc/ethers, /etc/hosts, and /etc/bootparams) are located in /etc on the NetServer.
- 2. Verify the domain name using the following command:

/bin/domainname

3. Run **ypinit** to build NIS maps for master and slave configurations:

```
/usr/etc/yp/ypinit -m
```

For more information on **ypinit**, refer to the man page.

4. Reboot the NetServer, which starts the NIS daemons, including **ypbind** and **ypserv**.

For more information about NIS, refer to Sun's System and Network Administration manual.

### Configuring the NetServer to Be an NIS Slave

If you use the server as an NIS slave, follow these steps:

- 1. Edit the /etc/ethers, /etc/hosts, /etc/bootparams, and /var/yp/<domain>/ypservers files on the NIS master. The /etc/ethers and /etc/bootparams files on the master must contain the slave server client information. The /etc/hosts file on the master must contain the slave NetServer and slave client host name and IP address. The /var/yp/<domain>/ypservers file, which is a list of slave servers in the domain, needs the slave NetServer name.
- 2. Verify the domain name:

/bin/domainname

3. Enter the **ypinit** command on the master server as follows:

```
/usr/etc/yp/ypinit -m
```

Answer all the questions displayed by **ypinit**, and add the slave NetServer name to the list of servers. For more information on **ypinit**, refer to the man page.

- 4. If you want to make sure you have set the type of NIS service and domain name properly in the **NSconfig** form on the slave server, enter **NSconfig** to display the form again. Verify the settings in the **NSconfig** form:
  - ▲ **slave** for "Type of NIS service"
  - ▲ NIS master's domain for "Domain Name"
- 5. On the slave NetServer, run the **ypinit** command.

```
/usr/etc/yp/ypinit -s mastername
```

6. Reboot the slave NetServer, which starts the NIS daemons, including **ypbind** and **ypserv**.

For more information about NIS, refer to Sun's System and Network Administration manual.



### **NetServers and UDP Checksumming**

On the NetServer, no flag needs setting for UDP checksummed packets from clients. Auspex NetServers check the UDP flag in a packet and deal with the packet accordingly. Clients can send packets checksummed or not; the NetServer correctly deals with the packets and then sends replies in kind.

## **Modifying Site-Specific Files**

In addition to the preceding procedures, you may also need to customize two types of site-specific files:

#### ▲ The SunOS kernel

Reconfigure the kernel in two situations: if you need to add devices that are currently not supported by the generic kernel (/usr/sys/aushp/conf/AUSPEX1), or if you change the site-specific parameters in the kernel. Refer to Sun's Network and System Administration manual for more information on reconfiguring the kernel. Refer to "Kernel Parameters" on page 2-6 for information about site-specific kernel parameters. See the section "Rebuilding the System Kernel" on page 10-9 for reconfiguration instructions.

#### ▲ Various files in the /etc directory

Table 2-1 lists the files that contain site-specific information such as mail aliases and trusted hosts. The table also describes the purpose of each file and lists man pages containing more information.

Table 2-1. Key files that contain site-specific information

File name	Purpose of the file	
/etc/aliases	Defines mail addresses or aliases, recognized by <b>sendmail</b> (8) for the local host. See <b>aliases</b> (5).	
/etc/bootparams	Contains new clients (this is configured automatically by <b>SetupClient</b> ). Se <b>bootparams</b> (5).	
/etc/configuration	Contains the server's primary host name, interface flags (each of which determines whether a network interface is enabled or disabled), NIS type, domain name, and route daemon flag. (This is configured automatically by <b>NSconfig</b> .)	
/etc/dumpdates	Updates dump records for individual file systems. See <b>dump</b> (8).	
/etc/ethers <sup>*</sup>	Defines Ethernet address-to-hostname mapping for network hosts, which is required for booting diskless SunOS clients. (This file is automatically updated by <b>SetupClient</b> .) See <b>ethers</b> (5).	
/etc/exports	Defines all exported file systems—for example, root and swap partitions to specific clients. (File systems for diskless clients are configured automatically by <b>SetupClient</b> and <b>SetupExec</b> .) Each diskless or dataless client must have an entry in this file. See <b>exports</b> (5).	
/etc/format.dat	Holds data for the <b>format</b> (8) program. This file defines the known disk types and partition information.	
/etc/fstab	Defines local and network file system mount points. See <b>fstab</b> (5).	
/etc/gateways	Defines IP router information. See routed(8C).	
/etc/group	Defines groups recognized by the system. See group(5).	
/etc/hosts <sup>*</sup>	Defines hosts and IP addresses for network devices. See <b>hosts</b> (5).	

Table 2-1. Key files that contain site-specific information (Continued)

File name	Purpose of the file
/etc/hosts.equiv	Defines trusted hosts. See <b>hosts.equiv</b> (5).
/etc/inetd.conf	Lists servers to be started by <b>inetd</b> (8C). See <b>inetd.conf</b> (5).
/etc/netgroup <sup>*</sup>	Defines network groups recognized by the system. See <b>netgroup</b> (5).
/etc/networks <sup>*</sup>	Defines network name information. See <b>networks</b> (5).
/etc/passwd	Defines users' password and account information. See <b>passwd</b> (5).
/etc/printcap	Defines printers. See <b>printcap</b> (5).
/etc/netmasks	Defines network masks used to implement IP subnetting. See <b>netmasks</b> (5). Holds the network masks that /etc/rc.boot uses for running the <b>ifconfig</b> command. (/etc/netmasks is configured automatically by <b>NSconfig</b> .) See <b>rc</b> (8).
/etc/raidtab	Holds DriveGuard (RAID) configuration information.
/etc/rc.local	Starts site-configured and general programs at system boot time, including NIS services, <b>keyserv</b> , other Sun utilities, and local utilities. (/etc/rc.local is configured automatically by <b>NSconfig</b> .) See <b>rc</b> (8).
/etc/rc.local. <hostname></hostname>	Starts daemons and does setup for local applications and machine-specific services.
/etc/rc.boot	Sets ifconfigs for broadcast addresses. (/etc/rc.boot is configured automatically by <b>NSconfig</b> .) See <b>rc</b> (8).
/etc/remote	Customizes serial port information used by tip(1).
/etc/rpc, /etc/protocols, and /etc/services*	Contains information about protocols used in TCP/IP. See <b>services</b> (5).
/etc/sendmail.cf	Defines site-specific mail-handling procedures. See <b>sendmail</b> (8).
/etc/syslog.conf	Defines any special logging requirements for system messages; that is, assigns special log files for certain messages. See <b>syslog.conf</b> (5).
/etc/ttytab	Defines serial ports. (/etc/ttytab is configured automatically by <b>NSconfig</b> and <b>SetupTty</b> .) See <b>ttytab</b> (5).
/etc/vpartab	Defines virtual partitions. See <b>vpartab</b> (5).

<sup>\*</sup> If your site uses NIS, these files are used only at boot time (before NIS is up).

#### **Kernel Parameters**

A kernel has three site-configurable parameters. The parameters in the generic kernel /usr/kvm/sys/aushp/conf/AUSPEX1 are maxusers and netinterfaces. Edit the file AUSPEX1 to change the parameters. Maxusers determines the approximate size of kernel data structures. Read the comments for maxusers in *AUSPEX1* for guidance.

#### **Additional TTYs**

In order to provide logins for additional TTYs, in the generic kernel /usr/kvm/sys/aushp/conf/AUSPEX1, change the parameter from pseudo-device pty to pseudo-device pty255. The default number of TTYs is 16.



The netinterfaces value determines how much memory is reserved for the exclusive use of interfaces. Each increment in the netinterfaces value adds 56 KB of memory to the amount reserved for interfaces.

A netinterfaces value of 0 (zero), which is the default, causes the system to compute its own number for the amount of memory to reserve. The computation takes the number of NP boards in the system, multiplies that by six, and then multiplies that by 56 KB. For example, a system with four NPs would get a netinterfaces working value of four times six, or 24. The amount of memory reserved for interfaces would be 56 KB times 24.

In general, the most efficient value to insert for netinterfaces is the actual number of interfaces in the system. A system with four NPs and three interfaces on each would need a netinterfaces value of 12, using only half as much memory as the default computation.

The maximum netinterfaces value recognized by the system is 64 for each NP up to a total system maximum of 256. For example, if you give a netinterfaces value of 100 on a system with one NP, the system reserves memory for a value of only 64.

A site's use of interfaces may call for more memory than the guideline given here, in which case you can increase reserved memory by changing the netinterfaces parameter. Remember that any change to this parameter is effective only after rebuilding and rebooting the kernel. See the "Rebuilding the System Kernel" on page 10-9 for rebuilding instructions.

## **Mailing Site Reports to Auspex**

If your site can communicate with Auspex using email, Auspex recommends that you edit /var/spool/cron/crontabs/root to ensure that weekly customer site reports are sent to Auspex. These reports contain uptime information, NFS statistics, and so on.

Add the following line to /var/spool/cron/crontabs/root so your server automatically sends the email message:

```
0 0 * * 5 /usr/auspex/reporter.sh >/dev/null 2>&1
```

The script file that sends the email message is /usr/auspex/reporter.sh. The message uses a subject line that includes the string you specify in the Customer Site field in the **NSconfig** form. Refer to the man page for **reporter.sh**(8) for more information.

**reporter.sh** calls **nfsstat**(8C), with the -**z** option, for network file statistics. The -**z** option zeros out and reinitializes the **nfsstat** statistics fields. If you do not want the fields to automatically reinitialize, edit the script /usr/auspex/reporter.sh. Comment out the **nfsstat** command that uses the -**z** option and uncomment the **nfsstat** command that does not use the -**z** option.

If you set up your system to mail weekly reports, you can also modify the destination address to add recipients other than Auspex to the mailing list. You add (and modify) recipients by editing the following line in /usr/auspex/mailmessages.sh and /usr/auspex/reporter.sh:

```
REPORT_STAFF="dl-customermessages@auspex.com"
```

You can also direct the report to a local user by adding his or her email address to the line. For example:

```
REPORT_STAFF="dl-customermessages@auspex.com jsmith@abc.com"
```



Additionally, you can specify recipients for system core dumps. By default, *usr/auspex/mailmessages.sh* identifies the recipient in the following line:

CORE\_STAFF="dl-corescreener@auspex.com"

You can also direct the system core dump to a local user by adding his or her email address to the line. For example:

CORE\_STAFF="dl-corescreener@auspex.com jsmith@abc.com"



# **NetServer Configuration Commands**

Auspex provides a set of configuration commands that simplify the configuration process:

NSconfig configures information about the NetServer, its network interfaces, and many of the services it supports. You must use this command after installing the NetServer for the first time.



**Note:** For specific information regarding ATM interfaces, refer to the appropriate documentation provided on the Auspex Premier Software Series CD-ROM.

- **SetupTty** configures a TTY port on the HP for a serial device.
- **SetupExec** installs architecture-dependent executables for diskless SunOS clients.
- SetupClient adds one or multiple SunOS client workstations to the list of clients that can boot from the NetServer or removes clients from the list.



**Note:** When you customize a NetServer for the first time, the order in which you use the commands is important. Use the commands in the order they are listed previously.

The commands reside in /usr/etc, and may be invoked from any directory. However, you must be root to invoke these commands.



**Note:** If you use the NetServer as a router, then after adding a new interface or changing an IP address on a NetServer, you must restart the daemon **in.routed**. See the **in.routed**(8C) man page for further information.

## **Using Configuration Command Forms**

When invoked, each configuration command displays a form. This form contains a set of fields that provide information to the program. Many fields contain a default, which you can either accept or override by entering other data into the field. To accept a field's default value, simply leave the field unedited and the default is used when the form is executed. Each field defaults either to the last value you used for that field or to the current value in /etc/configuration (if the configuration file has been modified since the last time the configuration form was executed).

After you fill in all the fields in the form and execute the form, the configuration program automatically executes a set of installation and configuration tasks.

Each form contains three regions (see Figure 2-2 on page 2-13 for an example):

- The form's title, date, and time are at the top of the screen.
- The main body of the form in the center of the screen contains two types of data fields:
  - Toggle fields are preceded by a greater-than symbol (>). These fields may contain any value from a predetermined set of values. To select a value, type the first character of the value or use the space bar to scroll through the available values one by one. (You may also accept the default by leaving the field unchanged.)



- Input fields are enclosed by brackets ([]). These fields require you to type an explicit value, or to accept the default value.
- Below the dotted line at the bottom of the screen, the commands for controlling the form appear (for example, commands for resetting the form and accepting the form). This area is also a message area, displaying explanatory messages when the cursor is on an input field and error messages when user input is invalid.

All configuration forms can be scrolled. If part of the form is invisible because the form is larger than the screen or window, press the Tab, Down Arrow, or Right Arrow key until the desired field appears. The string "MORE" followed by a number on the dotted line at the top and bottom of the screen indicates the number of lines currently not displayed.

Table 2-2 lists the commands and keys that control form functions.

Table 2-2. Keys for editing configuration forms

Key	Function
?	Selects the HELP function, which displays a help screen.
Ctrl-C	Cancels this form, and returns to the shell prompt.
Ctrl-F	Executes the form.
Ctrl-H, Backspace, or Delete	Deletes the previous character.
Ctrl-L	Repaints the screen (for X terminals only).
Ctrl-R	Resets each field to the default value.
Ctrl-U	Deletes the current line.
Ctrl-W	Deletes the word preceding the cursor.
Home	Scrolls the form to the top, and moves the cursor to the first field.
Page-Up	Scrolls the form up.
Page-Down	Scrolls the form down.
Space Bar	Toggles a value. When you set the time zone in the <b>NSconfig</b> form, keep pressing the space bar until the appropriate time zone appears.
Tab or Right Arrow	Moves the cursor to the next field.
Return	Moves the cursor to the first field on the next line.
Down Arrow	Moves the cursor down to the same field on the next line.
Left Arrow	Moves the cursor to the previous field.
Up Arrow	Moves the cursor up to the same field on the previous line.

## **NSconfig**

NSconfig configures NetServer software only. Use the SetupExec and SetupClient commands to install and configure software for diskless SunOS clients.



#### Before Running NSconfig

Before you begin, you need the following information:

- The primary host name of your server.
- For each network interface, the primary host name, Internet address, subnet mask, and the host part of the broadcast address (either all zeros or all ones).
- The X terminal name and Internet address if you are using an X terminal as the server console.
- The NIS service type and NIS domain name.
- Whether or not the route daemon runs on the NetServer.



**Note:** Load balancing through network routers or hubs is supported with the NetServer. However, the NetServer network interfaces that are designated as belonging to the same host must be on the same NP board and must be of the same network type.

For convenience in preparing this information, you may copy and use the configuration worksheets provided in Appendix D.



**Note:** The server hostname must have a corresponding IP address in /etc/hosts. You can explicitly set the primary hostname of one of the interfaces to the server hostname. For example, if the server name is Auspex and the server's IP address is the same as that of interface 0, then the name Auspex must be specified in the Server Hostname and AEO's Primary Hostname fields of the **NSconfig** form. If the server's IP address is different from any of the interface IP addresses, you must first specify the server's address in /etc/hosts before running NSconfig. NSconfig gives a warning if the server hostname does not have a corresponding IP address in /etc/hosts.

#### Running NSconfig

You can run **NSconfig** either in single-user or multiuser mode:

- 1. Enter **NSconfig** (the letters *NS* must be capitalized):
  - # NSconfig
- 2. If you have not set the TERM variable for your terminal, the server displays a numbered list of terminal types and prompts you to enter your terminal type. Select the number from the menu that corresponds to the terminal type you are using:
  - 1) NCD X terminal
  - 2) ANSI-compatible terminal (including VT220 and Link MC5)
  - 3) Wyse Model 30/50
  - 4) Televideo 925
  - 5) Sun Shell Tool
  - 6) Sun Command Tool
  - 7) DEC VT510
  - 8) Other

Select a number for your terminal type >>

For example, select 7 for a DEC VT510 terminal or 2 for a Link terminal.



If you select 8 (Other), you are prompted to enter the terminal type, as shown in the following example. The type must be in /etc/termcap and the terminfo database.

Enter terminal type (must be in both /etc/termcap and terminfo):

3. After you enter a valid terminal type, **NSconfig** displays the following notice that warns of changes in **NSconfig** functionality from previous system software releases. With System Software Release 1.10, NSconfig automatically does the configuration file updates and system modifications without requiring a reboot in most cases. The exceptions are noted in the message.

WARNING: DO NOT ALTER THE IFCONFIG LINES OF /etc/rc.boot. NSconfig relies heavily on them to function correctly.

PLEASE NOTE: NSCONFIG NOW EXECUTES IFCONFIG'S OF NETWORK INTERFACES

WHETHER RUN IN SINGLE OR MULTI-USER MODE. A reboot after completion of NSconfig will only be required if run in multi-user and when there is a change in IP address or netmask of an interface or when NIS type is changed to master or slave. Want to proceed with this ([y]/n)?

4. The **NSconfig** form displays. Figure 2-2 on page 2-13 shows an example of the form. Complete the form.

Table 2-3 on page 2-14 describes the fields in the NSconfig form. In some cases, if you select a certain value for a field, an additional field appears on the form. For example, if you select *Master, Slave*, or *Client* for the type of NIS service, you are prompted for a domain name. These additional fields are indented in the table. The table also lists the possible values for each field when appropriate.



**Note:** The **NSconfig** log file, *NSconfig.log*, is located in the directory /var/log.



Auspex NetServer N	NSconfig	Form		r 2 10:01 [TOP]
Server Hostname [SrvA]	]			
ast Ethernet Interface:ahme0				
Enable/Disable Interface[ahme0]:		>UP		
Mostname:		[SrvA]		
Internet Address:		[192.222.222.2]		
Reflect Mode:		>NO		
Subnet Mask:		[ffffff00]		
Broadcast Type:		>0		
Mode:		>Manual		
Ouplex Mode:		>full		
Speed:		>100		
Enable SMB on ahme0:>N				
Fast Ethernet Interface:ahmel				
Enable/Disable Interface[ahme1]:	>UP			
Mostname:		[SrvA]		
Internet Address		[192.211.221.2]		
Reflect Mode:		>NO		
Subnet Mask:		[ffffff00]		
Broadcast Type:		>1		
Mode:		>Manual		
Ouplex Mode:		>half		
Speed		>100		
Enable SMB on ahmel:>Y				
MetBIOS Node Type:>B				
Console>NO				
NIS service type		>slave		
Domain Name		[yourcompany.com		]
Create Default Route>Yes				
Hostname or IP address of def	ault gat	ceway [192.11.11.1	_]	
Run RouteD		>NO		
Run SnmpD		>NO		
Change Date and Time>NO				
Change Time Zone>NO				
Notify Auspex on reboot >NO				
Customer Site		[yourcompany [BOTT		]
				+
Jse TAB, CR or arrow keys to move P=HELP				
sp>=SELECT-VAL				

Figure 2-2. Sample NSconfig form



Table 2-3. The NSconfig form fields

Field	Description	Possible values
Server Hostname	Host name of this server. The <b>hostname</b> command uses this name to set the host name for this server. Normally, the name should be the primary host name of one of the interfaces below. Host name is set during the execution of <b>NSconfig</b> and updated in the /etc/configuration file and the X console configuration file.	
Ethernet Intf, Fast Ethernet Intf, FDDI Intf, ATM Intf	Numbered interfaces. Names and states of the interfaces reflect the current configuration of the system. For each interface, use the space bar to toggle between <b>UP</b> and <b>Dwn</b> . See the <b>NSconfig</b> (8) man page for ATM specific information.	UP or Dwn.
	NOTE: ATM interfaces are available as optional products. Refer to the appropriate documentation on the Auspex Premier Software Series CD-ROM for more information.	
Primary hostname	Host name of the interface. Only one interface needs to have a name that is the same as the server. Other interfaces can have names that are independent of the server name.  Although you can use aliases in /etc/hosts for a host name, specify the primary name, not an alias, in this field. For example, if /etc/hosts contains an entry as follows:  144.48.252.2 MKT marketing	The default name is the server name followed by <b>-en</b> (10Base-T Ethernet interfaces), <b>-ahmen</b> (100Base-T Ethernet on an HME adaptor), or <b>-fn</b> (FDDI interfaces) where <i>n</i> is the interface number.
	specifying "marketing" for 144.48.252.2 in the form causes <b>NSconfig</b> to comment out the above entry and to make "marketing" the primary name for the interface. The entry for this interface in /etc/hosts file will be changed to the following:  #144.48.252.2 MKT marketing  144.48.252.2 marketing	NOTE: ATM and interfaces are available as optional products. Refer to the appropriate documentation on the Auspex Premier Software Series CD-ROM for more information.
	Unless the server hostname already has an IP address associated with it in the /etc/hosts file, one of the primary hostnames must be the same as the server hostname. Otherwise, an error message, "Hostname unknown" appears when you use the <b>hostname</b> command.	mornation.
Internet Address	Internet address for each Network interface.	An IP address in decimal-separated octets format, for example, 192.9.200.100.
REFLECT Mode	A mode for Ethernet and FDDI interfaces. <b>Yes</b> turns on the REFLECT mode for that interface, which then reflects back all NFS responses from the interface to the MAC address from which the request arrived. REFLECT bypasses normal IP routing. See the <b>ifconfig</b> (8C) man page for further information.	Yes or No.



Table 2-3. The NSconfig form fields (Continued)

Field	Description	Possible values
Subnet Mask	Subnet mask for each network interface.	Null, the decimal- separated octets of the subnet mask such as 255.255.255.0, or an 8- character hexadecimal string such as fffff00. If you do not use subnets, delete all characters in this field. Default is ffffff00.
Broadcast Type	Select <b>0</b> if the host part of the broadcast address is all 0s; select <b>1</b> if it is all 1s. The broadcast address is generated based on the broadcast value, the subnet mask, and the Internet address.	0 (default) or 1.
FDDI MTU	Size of the maximum transmission unit in traffic sent over the FDDI network directly attached to the server. The MTU limits the size of a frame that can be sent by an FDDI interface. This value is tunable for each FDDI interface on the server.	<b>1500</b> or <b>4352</b> (default). It must be equal to or greater than <b>offnet MTU</b> for a given FDDI interface.



Note: Set MTU to 1500 if you use an internetwork bridge to connect an FDDI network and Ethernet network. (Ethernet supports a smaller MTU.) The smaller value allows all traffic to be forwarded between the FDDI and Ethernet networks. Also, if a station from another vendor cannot handle the maximum MTU size, set **MTU** to 1500.

FDDI Offnet MTU	Size of the maximum transmission unit for FDDI traffic addressed to a network not directly attached to the server. This value is tunable for each FDDI interface on the server.	1500 or 4352 (default). Because some FDDI-to-Ethernet routers cannot handle packet fragmentation effectively, if this value is set to 4352, these routers might drop packets that originate from the NetServer. Setting FDDI Offnet MTU to 1500 allows these routers to route packets between FDDI and Ethernet networks without fragmentation while achieving the highest possible speed for FDDI-to-FDDI traffic.
X Console	Specifies whether an X terminal is used as the system console.	Yes or No (default).
NIS service type	Type of NIS service the NetServer is providing to clients, if any.	Master, Slave, Client, or NONE (default).
Domain Name	NIS domain name. This field appears only if type of NIS service is <b>Master, Client</b> , or <b>Slave</b> . If type of NIS service is <b>NONE</b> , this field does not appear.	



Table 2-3. The NSconfig form fields (Continued)

Field	Description	Possible values
Create Default Route	Set up a default route. <b>Yes</b> displays a field in which to specify a gateway for default routing.	Yes or NO.
Run RouteD	Specifies how to run the route daemon.	Yes, No, or Quiet. If set to Yes (default), routing information is distributed to all other machines on the network that run the Routing Information Protocol (RIP). If set to Quiet, the route daemon still receives broadcasts from other machines but does not forward routing information from the NetServer. If set to NO, the route daemon is disabled. See routed(8C) for more details.
System Contact	Email address or name and telephone number of the person who is responsible for this server (for example, <b>jsmith@companyname.com</b> ).	
System Location	Location of the server (for example, 2nd FI, Building 2, ABC Corp., San Jose).	
Change Date and Time	Specifies whether to set <i>both</i> the date and time on the server. The setting does not affect the time zone setting. If you enter <b>Yes</b> , enter the date and time in the associated field. A one-second difference exists between the time when the form is executed and when the time is set. Auspex recommends that you set this field last, right before executing the form.	Yes or NO (default). Table 2-4 on page 2-17 lists the various date and time formats that are accepted in the NSconfig form.
Change Time Zone	Specifies whether to change the local time zone. The setting does not affect the date and time. If you enter <b>Yes</b> , select the time zone name in the associated field.	Yes or NO (default). Refer to Appendix C for a list of time zones. When entering the time zone name, type the initial letter of the name to display the first time zone starting with that letter. Pressing the space bar displays the next value.
Notify Auspex on reboot	When your server is rebooted, this field determines whether the server automatically sends an email message containing the message log to Auspex. The subject line in the message includes the string you specified for Customer Site, notifying Auspex Technical Support that your NetServer might have problems that require attention.  The script file that sends the email message is /usr/auspex/mailmessages.sh. Refer to the mailmessages.sh(8) man page for more information.	Yes or NO. If set to Yes, the NetServer sends an email message to Auspex upon reboot. Auspex recommends that you set it to Yes if your site can communicate with Auspex using email.



Table 2-3. The NSconfig form fields (Continued)

Field	Description	Possible values
Customer Site	The name of your site and the NetServer serial number. If specified, this information appears in the mail message your server sends to Auspex upon reboot.	

Table 2-4. Date and time formats

Date format	Time format	
dd/mm/yy dd/mm/yyy dd.mm.yyyy dd-mm-yyyy dd-mm-yy month dd, yyyy dd month yyyy	hh am/pm hh:mm am/pm hh.mm hh:mm am/pm hh:mm:ss am/pm hh:mm:ss hh.mm.ss am/pm hh:mss	



**Note:** Although the **NSconfig** form includes information related to the NIS name service, **NSconfig** does not configure the NetServer to be a client, a master server, or a slave server in the NIS environment. A separate procedure is required to prepare the server for using the NIS name service. Refer to "Configuring the Server to Be an NIS Master or Slave Server" on page 2-4 for more information.

5. After you complete the form, execute it by typing Ctrl-F. NSconfig prints several messages on the screen. Follow the instructions printed by NSconfig to complete the configuration procedure.

The NSconfig procedure is complete. Table 2-5 lists this files modified by NSconfig.

Table 2-5. Files modified by NSconfig

File	Description of modification
/etc/configuration	Enables or disables mail messages to Auspex upon system reboot. Adds the site name and serial number to the file, which are included in the message to Auspex.  Also, adds the server's primary host name, interface flags (each of which determines whether an interface is enabled or disabled), NIS type, domain name, route daemon flag, and console X terminal name.
/etc/hosts	Sets the IP address of each interface; adds the host names.
/etc/netmasks	Adds the IP subnet masks.
/etc/netconf	Adds the subnet masks to the <b>ifconfig</b> commands.
/etc/ttytab	Sets the console terminal type.
/usr/share/lib/zoneinfo/lo caltime	Links to the local time zone.

## **SetupTty**

Your NetServer supports two serial ports, referred to by the software as **ttya** and **ttyb**. To configure the TTY ports, use the **SetupTty** command. The flowchart in Figure 2-3 describes what you need to do when configuring the TTY ports.



**Note:** The **ttya** port is preconfigured to support a system console. Do not use **SetupTty** to change the configuration of this port.

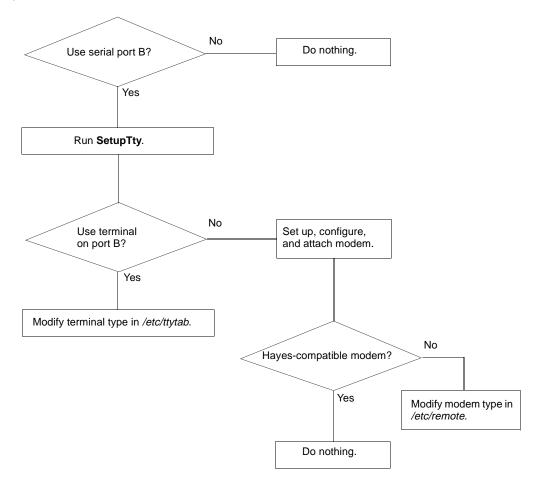


Figure 2-3. Flowchart for configuring a TTY port

The following describes the functions of the two serial ports:

- ▲ The first serial port (ttya) is preconfigured as a console port. You must configure the console terminal to match the configuration of ttya.
- ▲ The second serial port (ttyb) can support a modem or a terminal. Use **SetupTty** to configure ttyb for the device you intend to attach to the second serial port.



**Note:** If you specify **ttyb** as a modem port, **SetupTty** configures it to support a default modem type of hayes. If you plan to connect a Hayes-compatible modem, no configuration of this port is required. If you plan to connect a non-Hayes-compatible modem, edit /etc/remote to change the default settings manually.



**SetupTty** enables or disables logins using the file /etc/ttytab and signaling init(8). If you specify **ttyb** as a modem port, the baud rate you specify is set automatically in /etc/remote.



**Note:** SetupTty does not set the type and status fields in /etc/ttytab; you need to modify these fields manually.

If you specify ttyb as a terminal, and you want the port to default to a specific terminal type when you log in, you must modify the terminal type in /etc/ttytab.

Follow these steps to run **SetupTty**:

- 1. Log in as root. The system must be in multiuser mode for this procedure.
- 2. Run **SetupTty**:
  - # SetupTty

The SetupTty form appears as shown in Figure 2-4. The fields in the form are explained in Table 2-6.

spex NetServer	SetupTty Form	Wed Sep 13 13:1
Which port: (ttya/ttyb)	>ttyb	
Port baud rate: 300, 1200, 2400, 4800, 9600, :	>9600 19200	
Device connected to this port: 0 - no device connected 1 - dial-out modem 2 - dial-in modem 3 - local terminal	>0	
■Se TAB, CR or arrow keys to m HELP ^F=EXECUTE ^C=ABORT ^L=RED		

Figure 2-4. The SetupTty form

Table 2-6. The SetupTty form fields

Field	Description
Which port	Port to set up (ttya or ttyb).
Port baud rate	Baud rate: 300, 1200, 2400, 4800, 9600, or 19,200. By default, ttyb is set to 9600 baud.
Device connected to this port	Kind of device connected to the port being configured. The default is 0 (no device connected).
	<ul> <li>0 — no device connected, logins disabled</li> <li>1 — dial-out modem, logins disabled</li> <li>2 — dial-in modem, logins enabled</li> <li>3 — local terminal, logins enabled</li> <li>4 — serial printer, logins disabled</li> </ul>

3. Complete the fields in the form, and execute it by typing Ctrl-F. The command then returns you to the system prompt.

This completes the procedure for setting up the serial ports.



## **SetupExec**

Figure 2-5 is a flowchart describing the procedure for configuring client information on the server using **SetupExec**.

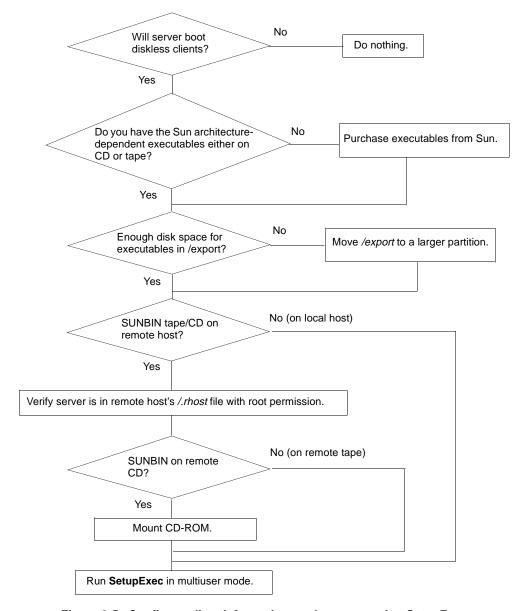


Figure 2-5. Configure client information on the server using SetupExec

## Do You Need to Run SetupExec?

**SetupExec** is required only if your installation includes diskless SunOS client workstations that boot from the NetServer. Client workstations with a local disk simply mount the exported file systems they need to access. Also, if clients require /usr from the NetServer, you must execute **SetupExec**. For more details on this command, refer to the **SetupExec**(8) man page.



#### What Does SetupExec Do?

SetupExec installs architecture-dependent executables for diskless SunOS client workstations from either a tape or CD-ROM, using either a local drive or a drive on a remote host. It adds a supported SunOS architecture to the NetServer. For example, the SunOS Version 4.1.3 Sun-4 executables are loaded into /export/exec/sun4.sunos.4.1.3, and a Sun-4 client mounts this directory as the client's /usr file system. Figure 4-8 on page 4-17 provides an illustration of the /export directory structures.

**SetupExec** updates two database files as follows:

/etc/exports **SetupExec** updates /etc/exports (see the exportfs(8) man page) to export the

executable directories it has installed. Check this file to verify that the

directory was exported to the correct groups.

/etc/hosts **SetupExec** updates /etc/hosts to add any remote CD-ROM host or tape host

names and IP addresses not currently in the database.

## Things to Remember When Running SetupExec

Before running **SetupExec**, be aware of the following:

- Run **SetupExec** in multiuser mode, only by the root user.
- Auspex does not supply the architecture-dependent executables; purchase the executables from your Sun supplier.
- Each architecture-dependent executable uses a variable amount of disk space, depending on the release. The /export file system is installed by default on a partition that allows enough space for about two executables. If you need room for more executables, move /export to a larger partition before running **SetupExec**. If **SetupExec** runs out of space trying to install the executables, it generates an error message and terminates.



**Recommendation:** Installing /export on a nonroot disk allows you to replace the root disk if necessary without having to restore /export.

If the CD-ROM containing the client executables is on a remote host, execute the **mount** command on the remote host before running **SetupExec** on the NetServer:

mount -rt hsfs /dev/sr0 /cdrom

if the remote host is a non-Auspex server, or

mount -rt hsfs /dev/acdn /cdrom

if the remote host is an Auspex server, where *n* is the slot number for the CD-ROM.

- When using either a remote tape or a remote CD-ROM, **SetupExec** accesses the remote host using **rsh**. The hosts file (/.rhosts) on the remote machine must include the name of your NetServer and have the correct root permissions.
- Run SetupExec before you run SetupClient. If you add a client with SetupClient and the appropriate architecture-dependent executables are not found, **SetupClient** generates an error message and terminates.

The Sun architectures used on various Sun workstations are listed in Table 2-7.

Table 2-7. Sun system architectures

Application architecture	Kernel architecture	Sun system models supported
sun3	sun3	3/50, 3/60, 3/75, 3/110, 3/140, 3/160, 3/260, 3/280
sun3	sun3x	3/80, 3/470, 3/480
sun4	sun4	4/110, 4/260, 4/280, SPARCsystem 330
sun4	sun4m	SPARCstation20, SPARCstation10, SPARCstation5, SPARCclassic
sun4	sun4c	SPARCstation 1 (4/60), SPARCstation 1+ (4/65), SPARCstation 2 (4/75), SPARCstation SLC (4/20), SPARCstation IPC (4/40), SPARCstation IPX (4/50), SPARCstation ELC (4/25)

#### Running SetupExec

The following procedure describes how to start **SetupExec** and fill out the **SetupExec** form:

- 1. Make sure the system is in multiuser mode. Log in as root.
- 2. Insert the SUNBIN CD or tape into the appropriate drive. If the SUNBIN medium is a CD-ROM and is installed on a remote host, mount the CD-ROM (see "Things to Remember When Running SetupExec" on page 2-21).
- 3. Execute **SetupExec**:
  - # SetupExec
- 4. The command displays the **SetupExec** form. Sample forms appear in Figure 2-6, Figure 2-7, Figure 2-8, and Figure 2-9.

Table 2-8 explains the fields in the SetupExec form. In some cases, if you select a certain value for a field, an additional field appears on the form. For example, if you select **Remote** for "Drive location," you are prompted for the media host and the host's IP address. The additional fields are indented in the table.



**Note:** If you specify a remote tape or a remote CD-ROM drive, you must have root permission on the remote host. To obtain root permission, edit the file /.rhosts on the remote host, and add an entry in this format:

hostname root

where *hostname* is the name of the Ethernet interface through which you send requests to the remote host. For example, if your local server is named "Engineering" and "Engineering-e0" is the interface attached to the network to which the remote host is connected, specify "Engineering-e0 root" in .rhosts.



Auspex NetServer	SetupExec Fo	rm	Fri Feb	2 15:02
Architecture type, \$AR SunOS version, \$VERS:	CH: [sun4c_ [4.1.4_			
FileServer partition, Path to executables: Path to kernel executa Path to /usr/share fil	[\$EXPORT les: [\$EXPORT]	1 /exec/\$ARCH.su /exec/kvm/\$ARC /share/sunos.\$	H.sunos.\$VERS_	
Distribution medium: ( Drive location: (Local Tape device name		e.g., rast1	, rast2, r	ast209
	Select the distribut			

Figure 2-6. Sample SetupExec form for a local tape device

Auspex NetServer	SetupExec Form	Fri Feb	2 15:00
Architecture type, \$ARCH: SunOS version, \$VERS:	[sun4c] [4.1.4]		
FileServer partition, \$EXPORT: Path to executables: Path to kernel executables: Path to /usr/share files:	[/export1 [\$EXPORT/exec/\$ARCH.sunos [\$EXPORT/exec/kvm/\$ARCH.s [\$EXPORT/share/sunos.\$VER:	unos.\$VERS_	
Distribution medium: (CDROM/Tape Drive location: (Local/Remote) CD-ROM device name Mount point of CD-ROM:	e) >CDROM >Local [acd1] e.g., acd1, ac [/cdrom	d2,, ac	:d209 ]
	among fields, when done hit ( EDRAW-SCRN ^R=INITIAL-VAL <		

Figure 2-7. Sample SetupExec form for a local CD-ROM device



Auspex NetServer	SetupExec Form	Fri Feb 2 15:03
Architecture type, \$ARCH: SunOS version, \$VERS:	[sun4c] [4,1,4]	
FileServer partition, \$EXPOR Path to executables: Path to kernel executables: Path to /usr/share files:	RT: [/export1 [\$EXPORT/exec/\$ARCH.; [\$EXPORT/exec/kvm/\$Al [\$EXPORT/share/sunos	sunos.\$VERS] RCH.sunos.\$VERS]
Distribution medium: (CDROM/ Drive location: (Local/Remot Tape device name	ce) Remote	0, rst8, rmt0, rxt0
Media host: [tapehost	] IP Addr	: [000,000,000,000]
	ne location of the tape driv "L=REDRAW-SCRN ^R=INITIAL-V	

Figure 2-8. Sample SetupExec form for a remote tape device

	 [sun4c]		
SunOS version, \$VERS:	[4.1.4]		
FileServer partition, \$EXPORT:	[/export1 [\$EXPORT/exec/\$ARCH.sunos	4UEDC	]
Path to executables: Path to kernel executables:	[\$EXPORT/exec/kvm/\$ARCH.s		
Path to /usr/share files:	[\$EXPORT/share/sunos.\$VER		
Distribution medium: (CDROM/Tape			
Drive location: (Local/Remote) CD-ROM device name	>Local [acd1] e.g., acd1, ac	:d2 ac	d209
Mount point of CD-ROM:	[/cdrom		]
<b>]</b> se TAB, CR or arrow keys to move =HELP ^F=EXECUTE ^C=ABORT ^L=RI			

Figure 2-9. Sample SetupExec form for a remote CD-ROM device



Table 2-8. The SetupExec form fields

Field	Description	
Architecture type	Architecture to install (for example, sun4, aushp). For further explanation of application architecture, see the <b>arch</b> (1) man page or Table 2-7.	
SunOS version	The SunOS version to install (for example, 4.1.3, 4.1.3_U1, 4.1.4).	
Path to executables	Path name of the directory in which to install the application executables. When <b>SetupExec</b> is complete, mount this directory as /usr by the diskless SunOS clients of architecture specified by "Architecture type."	
Path to kernel executables	Path name of the directory in which to install the kernel-specific executables. When <b>SetupExec</b> is complete, mount the path to executables as /usr/kvm by the diskless SunOS clients of the architecture specified by "Architecture type."	
Path to /usr/share files	Path name of the directory in which to install the architecture-independent executables. When <b>SetupExec</b> is complete, this director is ready to mount as /usr/share by the diskless SunOS clients of the architecture specified by "Architecture type."	
Distribution medium	Type of distribution medium; either tape or CD-ROM.	
Drive location	Location of the tape or CD-ROM drive. <b>Local</b> for a drive on the servor <b>Remvhote</b> for a drive on a remote host.	
Tape device name or CD- ROM device name	Tape or CD-ROM device name and slot number (for example, <b>acd1</b> for the Auspex CD-ROM drive in slot 1, or <b>rast9</b> for the Auspex tape drive in slot 9).	
Media host	Host name of the machine with the tape or CD-ROM drive. This field appears only if "Drive location" is <b>Remote</b> .	
IP Addr	IP address of media host. This field appears only if "Drive location" is <b>Remote</b> .	
Mount point of CD-ROM	Path name of the directory on the host on which the CD-ROM drive is mounted. This field appears only if distribution medium is <b>CD-ROM</b> . The default mount point is /cdrom. If drive location is <b>Local</b> , <b>SetupExec</b> automatically executes this <b>mount</b> command: mount -rt hsfs /dev/MEDIA_NAME /cdrom If drive location is <b>Remote</b> , execute <b>mount</b> on the remote host as follows before running <b>SetupExec</b> : <b>mount -rt hsfs /dev/sr0 /cdrom</b> if the remote host is a non-Auspex machine, or <b>mount -rt hsfs /dev/acdn /cdrom</b> if the remote host is an Auspex server, where <i>n</i> is the drive number.	



5. After you fill in all the fields in the **SetupExec** form, execute the form by typing Ctrl-F. The system displays messages similar to the following, which shows sun4c executables being loaded:

```
SetupExec: reading distribution CD-ROM...
Extracting the SunOS 4.1.3 sun4c "root" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "usr" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Kvm" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Install" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Networking" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "System_V" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Sys" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "SunView_Users" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "SunView_Demo" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Text" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Demo" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "OpenWindows_Users" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "OpenWindows Demo" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "OpenWindows_Fonts" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "User_Diag" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Manual" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "TLI" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "RFS" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Debugging" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "SunView_Programmers" CD-ROM
file.
Extracting the SunOS 4.1.3 sun4c "Shlib_Custom" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Graphics" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "uucp" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Games" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "Security" CD-ROM file.
Extracting the SunOS 4.1.3 sun4c "OpenWindows_Programmers" CD-ROM
file.
```

This final message indicates that **SetupExec** is finished.



## **SetupClient**

Figure 2-10 is a flowchart describing the procedure for configuring client information on the server using **SetupClient**.

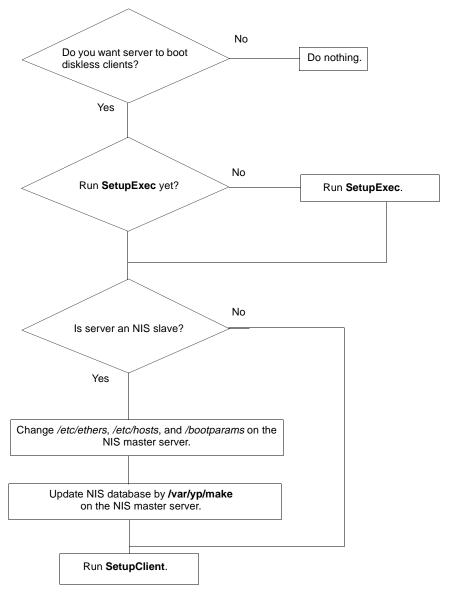


Figure 2-10. Configure client information on the server using SetupClient



#### Do You Need to Run SetupClient?

**SetupClient** is required only if your installation includes diskless SunOS client workstations that boot from the server. Client workstations with a local disk simply mount the exported file systems they need to access.

#### What Does SetupClient Do?

**SetupClient** adds or removes one or multiple diskless SunOS clients. If the NetServer is running NIS, **SetupClient** updates the NIS host's /etc/hosts, /etc/ethers, and /etc/bootparams databases.

If the NetServer is an NIS master, **SetupClient** updates the local NIS database; however, it does not propagate the local update to other NIS servers. If the NetServer runs NIS but is not an NIS master, **SetupClient** issues a warning to indicate that the database is out of date.



**Note:** Do not run this command unless you have already run **SetupExec** to install the architecture-dependent executables for the diskless SunOS client machine. The executable directory for the client's application architecture must be present on the NetServer. If this directory is absent, **SetupClient** generates an error.

**SetupClient** updates the */etc/exports* file to allow root access to each client's root file system. It exports the client's swap and dump partitions only to the client.

When executed, **SetupClient** performs the following steps:

- Adds an entry for the client to /etc/ethers, /etc/hosts, /etc/bootparams, and /etc/exports.
- Builds a root directory to be mounted by the client machine.
- ▲ Builds a swap file to be used as swap space by the client machine.
- ▲ Uses **make** and **push** on the NIS maps (/etc/ethers, /etc/bootparams, and /etc/exports) if the server is an NIS master.
- ▲ Runs the **exportfs** program.
- ▲ Creates a link to the primary boot program in /tftpboot.



**Note:** Analyze the client *root* and *swap* needs of your environment before deciding how to partition the disks on your server. Refer to "Example of File System Distribution" on page 4-11 for more information on setting up partitions for clients' *root* and *swap* directories.

After running **SetupClient** to add a client, you can then boot that client machine.



#### Running SetupClient

1. Determine the IP address and Ethernet address of the client.

One way to determine the client's Ethernet address is to power on the client and watch it display the power-on messages. For convenience in organizing this information, copy and use the configuration worksheets in Appendix D.

- 2. If your network uses NIS, and if the NetServer on which you run **SetupClient** is not the NIS master, follow these steps:
  - a. Change the NIS text maps of these files on the NIS master:

ethers hosts bootparams

- b. Enter the following commands on the NIS master:
  - # cd /var/yp; make
- 3. Log in as root on the NetServer to be configured. The system must be in multiuser mode for this procedure.
- 4. Execute **SetupClient**:
  - # SetupClient

Figure 2-11 illustrates the **SetupClient** form. Table 2-9 describes the fields in the **SetupClient** form.

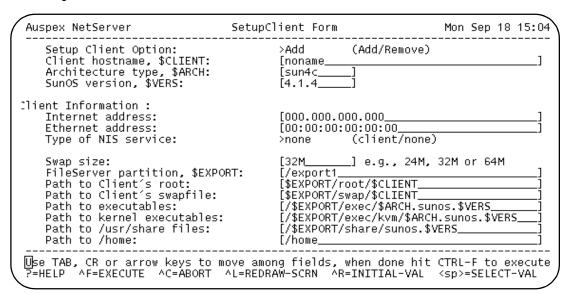


Figure 2-11. The SetupClient form

Table 2-9. The SetupClient form fields

Field	Description
Setup Client Option	Indicates whether to add or remove a client.
Client hostname	Host name of the client machine.
Architecture type	Client's architecture (for instance, sun4c, sun4, sun3x, or sun3). See Table 2-7.
SunOS version	The SunOS version running on the client (for example, 4.1.3 or 4.1.4).
Internet address	Internet address of the client.
Ethernet address	Ethernet address of the client.
Type of NIS service	Type of NIS service to provide to the client, if any. It can be either <b>client</b> or <b>none</b> . A diskless SunOS client cannot be an NIS master.
Domain name	NIS Domain Name. This field appears only if the client is running NIS service.
Swap size	Number of bytes reserved for the client's swap file. <b>SetupClient</b> creates <i>SWAP_PATH/CLIENTNAME</i> with this size. You can append one of <b>K</b> or <b>k</b> to indicate kilobytes, or <b>M</b> or <b>m</b> to indicate megabytes. (For example, 16M or 16000K.)
FileServer partition	Path name of the export partition in which various client directories and files reside.
Path to Client's root	Path name of the client root directory. This is the directory the client mounts as /. You must enter the client's host name by itself.
Path to Client's swap file	Path name of this client's swap file. You must enter the client's host name by itself.
Path to executables	Path name of the directory in which the architecture-dependent executables reside. This is the directory that the client mounts as /usr.
Path to kernel executables	Path name of the directory in which the kernel-specific executables reside. This is the directory that the client mounts as /usr/kvm.
Path to /usr/share files	Path name of the directory in which the architecture-independent operating system files reside. This is the directory that the client mounts as /usr/share.
Path to /home	Path name of the directory in which various home directories reside. It is the path name of the directory that the client mounts as /home. This field is optional; if omitted, no /home mount point is created in the client's /etc/fstab.

5. Fill in all fields of the **SetupClient** form, and execute it by typing Ctrl-F. The following messages appear:

```
updated ethers
pushed ethers
updated hosts
pushed hosts
Creating root for client "CLIENTNAME"
Creating swap for client "CLIENTNAME"
Updating /etc/exports to export "CLIENTNAME" info.
Updating /etc/bootparams...
updated bootparams
pushed bootparams
Completed creating ARCH client "CLIENTNAME".
#
```

Now reboot the client machine.



# Root Login and File System Isolation

When a file system managed by the FP (an LFS-mounted file system) becomes corrupted, the FP isolates the file system so you can repair it. However, you cannot log in to the server as root after a file system is isolated if the root's .login or any shell startup file contains references to that file system. For example, if /usr/local is LFS-mounted and /usr/local/bin is included in the PATH of .login, you cannot log in as root once /usr/local is isolated. To ensure that you can log in to a server as root after a file system is isolated, do not include commands that try to access an LFS file system in the root's .login file or any shell startup file such as .cshrc and .profile.

For more information on file system isolation, refer to "File System Isolation" on page 6-10. For more information on LFS file systems, refer to "Mounting File Systems on the FP and HP" on page 4-18.



# 100Base-T Ethernet Full-Duplex

Auspex 100Base-T Ethernet full-duplex supports both full-duplex and half-duplex modes with the HME SBus adapter card. (The older BMAC SBus adapter card supports only halfduplex mode.) 100Base-T Ethernet full-duplex is bundled into the Auspex base system software and requires no separate installation. Configuring 100Base-T Ethernet connections is the same as configuring 10Base-T Ethernet connections. Use NSconfig(8) to assign each connection an IP address, subnet mask, and so on.

HME interface performance may be hindered when the interface is set up with autonegotiation mode. Use the NSconfig(8) or ifconfig(8) command to manually set the HME parameters of duplex mode and speed.

A 100Base-T Ethernet interface connects to either a full-duplex interface, a half-duplex interface or an Ethernet switch port. Be sure that the Ethernet switch port is set to the proper mode: full- or half-duplex.

The 100Base-T Ethernet interface uses either an RJ45 or an MII connector. The connector is labeled 100Base-T. If the connector is cabled to a 100Base-T Ethernet hub on the other end, the interface functions at 100Base-T speed. If the connector is cabled to a 10Base-T Ethernet hub on the other end, the interface functions at 10Base-T speed. In the same way, to run in full-duplex mode, the connector must be cabled to another full-duplex interface or to an Ethernet switch that supports full-duplex operation on the other end.

Sun clients set up for Ethernet communications use auto-negotiation to determine the duplex mode of the other end of the connection. When connected to a NetServer through a switch, a hub, or directly back-to-back, the auto-negotiation feature causes unpredictable results. A Sun client using any such connection must be manually set to the proper duplex mode. See the Sun client documentation for details on setting the duplex mode.



# **3** Booting Up and Shutting Down the NetServer

## **About This Chapter**

This chapter provides procedures for accessing the PROM monitor and booting a server. It also describes the commands that shut down a server. This chapter assumes that you have finished the initial server configuration described in Chapter 2.

Before you boot the server, familiarize yourself with the major changes that Auspex makes to SunOS. These changes are described in "SunOS Enhancements" on page 1-12.



## **Booting the NetServer**

Auspex boot, shutdown, and halt procedures are similar to SunOS procedures with these differences:

The permissible boot devices are limited to ad for Auspex disk and CD-ROM drives. (The NetServer does not support booting from tape drives.)



**Note:** The device type **ad** applies to both disk and CD-ROM drives only when booting the NetServer. At all other times, the CD-ROM device type is acd.

- The Auspex boot procedure automatically implements the WarmStart feature in which boot or reboot time is improved by bypassing selected Power-On-Self-Testing (POST) diagnostics. If you wish to perform a "cold" boot, which does perform all diagnostics, the boot, reboot, and fastboot commands include a -c flag, which causes the reboot to perform the standard POST tests.
- You cannot boot the NetServer remotely from another device on the network.
- The commands fasthalt and fastboot have the same effect as halt and reboot on file system checking upon reboot. That is, all these commands cause the server not to check the file systems that unmounted cleanly before the operating system was halted. The fasthalt and fastboot commands, however, also skip system cache I/O testing, which is performed when the system reboots after a halt or reboot command.
- The **reboot** and **halt** commands include the -s flag, which causes the reboot not to perform a sync(2) operation on each file system. The -s flag ensures that the reboot does not hang if the NetServer has mounted a file system from another server that can not be reached. (See "Commands for Shutting Down the NetServer" on page 3-9 on page 3-9.)



**Note:** When booting the NetServer, verify that the diagnostic switch on the HP is set to the NORM position, not the DIAG position. Refer to your hardware documentation for more information.

The **shutdown**(8) command includes the **-l** option to send warning messages only to logged-in users. This option speeds up the shutdown process if the list of machines mounting files is not up to date, or if machines mounting files do not have rwalld(8C) support.

## Single-User Mode vs. Multiuser Mode

The UNIX operating system supports two modes of operation: single-user and multiuser. In multiuser mode, users may log in to the UNIX host, whereas in single-user mode only the system console is available. Commands typed in single-user mode execute with superuser (root) privileges. Furthermore, many network services and daemons start only when the system enters multiuser mode. Thus, when the system runs in single-user mode, it is inaccessible from the network, and it cannot serve up files for clients.

The NetServer boots automatically to multiuser mode at power on. Under normal operating conditions, you do not need to boot the system manually. However, some kinds of tasks (such as the system installation procedure and some system management procedures) require manual booting using the PROM monitor boot facility.





**Note:** A system on its way to booting in multiuser mode attempts to mount all exportable file systems. If a file system mount fails because of a problem with a virtual partition or a bad disk, the system boots to single-user mode with error messages.

The procedure for booting in single-user mode can be considered a modification of the usual multiuser boot procedure; that is, appending the -s flag to the boot command line causes the system to boot in single-user mode.

The procedures for booting in each mode are described in the following sections.

For more detailed information about system booting and shutdown procedures, see Chapter 5 in Sun's System and Network Administration.

## Accessing the PROM Monitor

The PROM monitor program resides in PROM on the HP board. This program provides a set of functions that control the system before the operating system is booted. These control functions include:

- **Bootstrap operations**
- Low-level system configuration
- Power-on diagnostics
- Extended diagnostics
- Resetting the subsystem controllers

Use the PROM monitor program whenever you want to use one of the functions listed above. This section discusses only how to access the PROM monitor so that you can boot the server manually. Only Auspex-qualified field personnel need the information about executing diagnostics commands.

#### **Entering Monitor Mode When the System is Powered Off**



**Caution:** Do *not* use this procedure after the automatic boot process has started or while the operating system is running. File system damage and data loss can result.

- 1. Locate the power switch on the power distribution unit (PDU). Refer to the hardware manual for your NetServer for more information on the PDU location.
- 2. Set the switch to the ON position.
- Wait while the system performs the power-on self tests. If the self tests execute successfully, this message appears:

```
Self test passed
```

4. Once this message appears, escape the automatic boot procedures by pressing the Caps Lock and Pause/Break keys simultaneously (on a DEC VT510 terminal), the Break key (on an X or a Link terminal), or the Shift and Break keys simultaneously (on a Wyse terminal).



5. The system enters the PROM monitor. The monitor prompt appears as follows: HP>

The monitor prompt indicates that the system is ready to accept PROM monitor commands.

#### **Entering Monitor Mode When the System is Running**

To access the PROM monitor when the operating system is running, halt the operating system. Use one of the following procedures, depending on whether the server is running in single-user or multiuser mode:

**In single-user mode.** Halt the operating system using the **halt** command. Once halted, the system automatically transfers control to the PROM monitor program, as indicated by the HP> prompt.

The following example shows a typical **halt** command and the system's response:

```
# halt
Syncing disks ... done
```

In multiuser mode. Halt the operating system using the /etc/shutdown command with the -lh options. This command brings down the system gracefully, issuing warning messages to users before transferring control to the PROM monitor program.

For example, the following command halts the system five minutes after the command is executed:

```
# /etc/shutdown -lh +5 "system going down in 5 minutes"
System shutting down at 14:58 PST
[The system shutdown message is repeated every 60 seconds]
```

The -l option sends the warning message only to users who are logged in, shortening the time required for the shutdown if there are PC or Macintosh users on the network who mount file systems from the server.

The monitor prompt indicates that the system is ready to accept PROM monitor commands.

For more information about commands related to shutting down the system, refer to "Commands for Shutting Down the NetServer" on page 3-9.

## **Booting the Server at the Monitor Prompt**

This section describes how to boot the server after you halt the operating system. The procedure in this section boots the server from the default boot device or the device you specify. This section also explains the various arguments for the **boot** command. (The **boot** command can be abbreviated to **b**.)



**Note:** The NetServer does not support booting to multiuser mode from the CD-ROM.



#### The Default Boot Device

The default boot device is disk unit 0 (ad0): if no device is found at this location, the boot attempt fails. The default boot device value is recorded in the system EEPROM and can be changed by the system administrator; for instructions on how to do this, refer to the eeprom man page.

You can also specify the device from which the server boots in the **b** command. The next section discusses the various arguments for the **boot** command.

#### **Using the Boot Command**

The following is the **boot** command syntax:

HP> b dev(sp,unit,part) path options

The **b** command takes the following arguments:

dev Device type, which is ad for both Auspex disk or CD-ROM drives. The NetServer cannot boot from a tape drive.



**Note:** The device type **ad** applies to both disk and CD-ROM drives only when booting the NetServer. For other commands, the CD-ROM device type is acd. Refer to "CD-ROM Drives" on page 4-2 for more information on CD-ROM drive naming conventions.

SP number, which can be 0, 1, 2, 3, or 4 for the NS 7000/700 and NS 7000/800 Series sp NetServers (the default is 0), and 0 for the NS 7000/250 and NS 7000/150.

A disk number relative to the SP number: The supported range is 0 to 9. If you are unit booting an NS 7000/250, the unit number is the same as the disk slot number.

For example, to boot from a disk drive in disk slot 5, determine the SP number (0-4), and enter the following command:

HP > b ad(0,5,0)

This example specifies a boot from SP0 (the first SP in the NetServer), drive 5, and partition 0 (see the part argument below). Here are some other typical boot examples:

HP> b ad(0,0,0) (Boot the NetServer from the root drive.)

HP> b ad(0,2,0) (Boot the NetServer from the back-up drive in disk slot 2.)

HP> b ad(0,1,1) (Boot the NetServer from the CD-ROM drive in disk slot 1, part 1, which is described as follows.)

Disk partition number (if booting from disk) or HP identification number (if part booting from CD-ROM) for the boot device.

For disk drives, this number ranges from 0 to 7, where 0 corresponds to partition **a**, and 7 corresponds to partition **h**. The default is 0 for partition **a**.

For CD-ROM drives, this number is 1.

The full path specification of the file to boot. The default is *vmunix*. path

*options* A list of options for the boot program:

- -a Includes the ask me option on the **boot** command. This option is required when booting from nonstandard partitions. When using the ask me option, the system asks you to specify the root and swap partitions explicitly. (Normally, you do not use this option when booting the NetServer.)
- -s Boots in single-user mode. If you do not specify -s, the system boots to the default multiuser mode. This option is not required when booting from the CD-ROM because the kernel on the CD-ROM automatically boots in singleuser mode.
- -c Performs a cold boot. If you do not specify -c, the system performs a WarmStart in which several POST tests are skipped, speeding up the time required to boot up to the console prompt. This option does not modify WarmStart's default enabling. Subsequent restarts skip POST tests unless the -c flag is included in the command.

To invoke the default values for the **b** command, enter the following:

HP> b

The system boots vmunix from device **ad0** in multi-user mode. The command is equivalent to the following:

HP> b ad(0,0,0)vmunix

To boot in single-user mode, use the following command:

HP> b -s

Table 3-1 shows some other examples:

Table 3-1. Examples of the b command

Command	Explanation
b ad(0,2,0) -a	Boot vmunix from the backup root disk in slot 2 in multiuser mode with the <b>ask me</b> option, which allows you to specify the root and swap partitions.
b ad(0,0,0) -c	Boot vmunix from the root disk in slot 0 in multiuser mode with the <b>cold boot</b> option, which specifies that all POST tests be performed upon booting.
b vmunix.backup	Boot from an alternate kernel file on <b>ad0</b> in multiuser mode.

#### **Exiting Single-User Mode to Multiuser Mode**

After you boot the server in single-user mode, the system provides you with a single-user shell. After you complete the procedure that requires single-user mode, and are ready to exit to multiuser mode, run the fsck -p command to check all file systems. After all file systems are checked, type Ctrl-D to go to multiuser mode. (You do not have to reboot the NetServer to return to multiuser mode.)

#### **Booting From a CD-ROM**

This section describes how to boot from the CD-ROM in single-user mode to perform system maintenance. (You cannot boot the server in multiuser mode from a CD-ROM.) This section does not describe how to install the server software from the CD-ROM; for the



software installation procedure, refer to "Using NSinstall to Install Software From CD-ROM" on page 6-29.



**Note:** When booting from a CD-ROM, the CD-ROM must be in slot 1. Also, if a root disk is installed in slot 0, the server uses partition **b** of the root disk as swap space when booting from a CD-ROM.

If you keep the CD-ROM in slot 1 after booting and the /etc/fstab file has /dev/ad1 entries (for example, entries for /dev/ad1a, /dev/ad1c, and so on) for the disk that the CD-ROM replaced, remember to modify the /dev/ad1 entries.

#### Loading the CD

Depending on the type of CD-ROM drive, follow the appropriate steps to load the CD.

If the CD-ROM uses a CD caddy:

- 1. Open the CD-ROM caddy and place the CD in the caddy with the CD artwork facing out.
- 2. Close the CD caddy completely.
- 3. Gently push the caddy into the drive. The drive engages and automatically loads the caddy.
- 4. Close the drive door.

If the CD is installed directly into the CD-ROM drive:

- 1. Open the CD-ROM drive.
- 2. Place the CD in the CD-ROM drive with the CD artwork facing out. Make sure that the CD is secured by the three plastic latches at the corners of the CD-ROM tray.
- 3. Close the CD-ROM drive door.

#### Booting the server from the CD

1. Access the PROM monitor. The PROM monitor prompt appears:

HP>

2. Enter the **boot** command as follows:

```
HP > b ad(0,1,1)
```

The NetServer displays the following messages:

```
Remounting and replenishing /etc
Remounting and replenishing /var
Remounting and replenishing /tmp
```

AUSPEX CD-ROM MAINTENANCE MENU

#### OPTION:

- 1) Format/label the root drive
- 2) NSinstall
- 3) Maintenance shell

Select an option? [1/2/3] (2)



If drive 0 is not labeled as a root drive, or the drive is not present, error messages appear, prompting you with the appropriate action.

3. Select an option from the Maintenance Menu.

Selection 1 displays the Format menu. Selection 2 displays a message about adding swap space and then runs NSinstall. For example:

```
Adding swap on /dev/ad0b
```

The message appears if you have swap mounted on the **b** partition of drive 0. See Chapter 6 of this manual for instructions on using NSintall(8).

Selection 3 puts you into a maintenance (single-user mode) shell from which you can run commands with root privilege. For example, select this option if you need to repair a damaged root disk (refer to "Recovering a Damaged Root Disk" on page 6-26). To return to the Maintenance Menu, type "exit" on the command line.

This completes the procedure for booting the server from the CD.

## Rebooting the Server to Multiuser Mode

If the server is currently in single-user mode, use the reboot command to reboot the server without first going to monitor mode. The reboot command executes the sync command to complete all the disk writes, loads the kernel into memory, and uses **fsck** to check unclean file systems before bringing up the server in multiuser mode.

If the server is currently in multiuser mode, do not use the reboot command. Instead, use **shutdown** to notify users of the reboot and bring the system down to single-user mode. Then use **reboot** to boot the system to multiuser mode.

## DataGuard and Rebooting

A reboot after a crash on a system running DataGuard can either interrupt or maintain functionality on processors other than the HP.

- If you used the **Break**> key to get to the PROM monitor prompt after a crash, use the **hpboot** -d command to reboot the HP only. Without the -d option, all the processors will reboot, losing the DataGuard protection of uninterrupted functionality.
- If you use **hphalt** to get to the PROM monitor prompt after a crash, be sure to use the -d option with **hphalt**. To reboot the HP only, use **hpboot** with no options.
- If you want to get crash dumps from all processors, then all the processors must be rebooted. Use the **Break**> key to get to the PROM monitor prompt, then give the following command, which gives crash dumps and reboots all processors:

```
HP> go 0
```



## Commands for Shutting Down the NetServer

This section summarizes the characteristics of the commands and scripts that shut down or reboot a system. The Auspex versions of these commands work in the same way as the ones in SunOS unless noted otherwise in this section. For detailed information about these commands, refer to the man pages:

- shutdown(8)
- halt(8)
- reboot(8)
- fasthalt(8)
- fastboot(8)

All of these commands execute **sync** by default to write out information to the disk before shutting down or rebooting the system.

#### shutdown

Use the **shutdown** command to shut down the system if it is currently in multiuser mode. The command allows you to specify an amount of time before shutdown occurs and broadcast a message to all system users. When executed without the -h option, shutdown brings the system down to single-user mode; with the -h option, the system enters monitor mode after the shutdown is complete.

The syntax of the **shutdown** command is as follows:

/usr/etc/shutdown [-dfhklnr] [-p processor] time [warning message]

The options are as follows:

-d Dump system core.

-f Shut down the system in the manner of **fasthalt**.

-h Execute halt.

Simulate shutdown of the system. Do not actually shut down the -k

system.

-1 Send the **shutdown** message only to users logged in to the system,

not to users who have file systems mounted from the system. This option causes the **shutdown** command to complete in a shorter

time.



**Recommendation:** When using the -l option, be sure to notify the users who have mounted file systems from the server before you shut it down.

Avoid running /etc/rc.shutdown and executing sync. rc.shutdown is -n

an Auspex script that attempts to unmount all the mounted LFS and UFS file systems except /usr and /. Because this option avoids unmounting the file systems, if an LFS file system was active before the reboot, it is not marked "clean" and is checked by fsck

at reboot. This option is used during emergencies.

-r Execute reboot.



-p *processor* Shut down only the named processor if *processor* is the HP.

*time* The time that the system is shut down.

warning message The message you want to broadcast to users who have files

mounted on this server, explaining why the server will be shut

down, when it will be back up, and so on.

If you specified **Yes** in the "Notify Auspex on reboot" field in the **NSconfig** form, your NetServer automatically sends electronic mail to Auspex at reboot. The mail messages allow Auspex to be informed of NetServer reliability at customer sites. Because the mail message contains the **shutdown** message, Auspex Technical Support can more easily diagnose your server problems.

#### halt

The **halt** command stops the operating system more quickly than **shutdown**, but its syntax does not allow you to send out a message to system users warning them of the impending loss of service. Use **halt** only when the system is currently in single-user mode, or when you are sure that no users are logged in to the system or have file systems mounted from the system. The **halt** command brings the system down to monitor mode without delay.

The following is the **halt** command syntax:

```
/usr/etc/halt [ -dngs ] [ -p processor ]
```

The options are as follows:

-d Dump system core before halting.

-n Avoid running /etc/rc.shutdown and executing sync. rc.shutdown is an Auspex script that attempts to unmount all the mounted LFS and UFS file systems except /usr and /. Because this option avoids unmounting the file systems, if an LFS file system was active before the reboot, it is not marked "clean" and is checked by fsck at reboot. This option is used during

emergencies.

-q Reboot the system quickly and ungracefully, without first shutting down

the running processes.

-s Avoid calling **sync**(2) at reboot so **reboot** does not hang if the machine from

which the NetServer has mounted a file system cannot be reached. File

systems are still synced during the multiuser reboot.

-p *processor* Shut down only the named processor if *processor* is the HP.

The system uses **fsck** to check the unclean file systems when rebooting after a **halt** command.

#### reboot

The **reboot** command runs /etc/rc.shutdown to unmount all mounted file systems except /usr and /, shuts down the running processes, and reboots the server to multiuser mode.

Use **reboot** only if the server is currently in single-user mode. If not, use **shutdown** first to warn users of the impending loss of service, and then reboot the server from single-user mode.



At reboot, **fsck** only checks the file systems that are neither "clean" nor "stable." A file system is marked clean if it has been unmounted before the reboot; it is marked stable if its file system structure was inactive since the last **sync** before the reboot.

The following is the **reboot** command syntax:

/usr/etc/reboot [ -dnqs ]	] [	[ -p processor	]	[ .	boot	arguments	]
---------------------------	-----	----------------	---	-----	------	-----------	---

The options are as follows:

- -d Dump system core before booting.
- Avoid running /etc/rc.shutdown and executing sync. rc.shutdown is an -n Auspex script that attempts to unmount all the mounted LFS and UFS file systems except /usr and /. Because this option avoids unmounting the file systems, if a file system was active before the reboot, it is not marked "clean" and is checked by **fsck** at reboot. This option is used during emergencies.
- Reboot the system quickly and ungracefully, without first shutting down -q the running processes.
- Avoid calling **sync**(2) so that **reboot** does not hang if the machine from -S which the NetServer has mounted a file system cannot be reached. File systems are still synced during the multiuser reboot.
- -p processor If *processor* is the HP, then reboot only the HP.

A boot argument is a string passed as an argument to the **boot** command in the PROM monitor. The string must be preceded by an option terminator string '--'.

Perform a cold boot. If you do not specify -c, the system performs a -- -c WarmStart, which skips several POST tests and speeds up the time required to boot. This option does not modify WarmStart's default enabling. Subsequent restarts skip POST tests unless the -c argument is included in the command.

Refer to "Using the Boot Command" on page 3-5 for more information on the arguments to **boot** command.

#### fasthalt and fastboot

The **fasthalt** and **fastboot** scripts perform the same functions as the **halt** and **reboot** commands, respectively. The only difference is that fasthalt and fastboot disable testing of the system cache I/O memory upon reboot. Use **reboot** if you suspect the system is experiencing problems with system cache I/O memory.

The following are the **fasthalt** and **fastboot** syntaxes:

```
/usr/etc/fasthalt [ halt options ]
/usr/etc/fastboot [ boot options ]
```

The **fasthalt** options are the same as those used for the **halt** command.

The **fastboot** options are the same as those used for the **reboot** command.



## **NetServer Storage Devices** and File Systems

## **About This Chapter**

This chapter discusses various features of the NetServer storage subsystems. In addition, it describes a file system type unique to Auspex—the Local File System (LFS). This file system type is a key feature of the NetServer software architecture, in which the UNIX operating system is separated from the file system.

The following lists the major topics covered:

- The Auspex device naming conventions
- **Default disk partitions**
- Distribution of file systems
- Adding new drives
- Removing drives
- Formatting drives



**Note:** The naming convention for 4-mm tape drives changed in software Version 1.5.1. If you are upgrading your server from a pre-1.5.1 version, be sure to read the "Tape Drives" section on page 4-2.



## **Device Names**

This section describes the conventions for assigning device names to Auspex disk, CD-ROM, and tape drives.

#### **Disk Drives**

Disk drive devices are named **ad***n*, where *n* corresponds to the drive slot. The maximum number of slots depends on your NetServer model. DriveGuard arrays are named **ard***n*. See the *DriveGuard Manager's Guide* for array numbering information.

#### **CD-ROM Drives**

How you refer to a CD-ROM drive depends on where the device name is used. Table 4-1 describes the name for the CD-ROM drive in various commands. The letter n in the table corresponds to the drive slot and is in the range of slot numbers available in your server; the letter p is the name of the partition.

Table 4-1. CD-ROM naming conventions

Command	Naming convention	Examples
boot	ad n (where n defaults to 1 because the CD-ROM must be in slot 1 to boot the server)	b ad(0,1,1)
eject, dumpfs	acdn	eject acd1 dumpfs /dev/acd1
dkinfo	ad <i>n</i>	dkinfo ad1
mount (the first partition)	acdn	mount -rt lfs -o fs=4.2 /dev/acd1 /mnt mount -rt hsfs /dev/acd1 /mnt
mount (any partition)	ad <i>np</i> acd <i>np</i> , ad <i>np</i>	mount -rt lfs -o fs=4.2 /dev/ad1a /mnt_ad1a mount -r hsfs /dev/acd1a /mnt_ad1a

## **Tape Drives**

Auspex does not currently ship all the drives in the following tables. We include information for previously shipped drives for those sites upgrading system software. Follow these guidelines when naming tape drives:

- ▲ Tape drive devices are named **rast***n*, where *n* corresponds to the drive slot and is in the range 1-209. (Slot 0 is always occupied by the root disk.) A tape drive with this name uses the *rewind* option. That is, when a program finishes using a tape drive, the drive rewinds the tape.
- ▲ Tape drive devices with the *no-rewind* option are named **nrast***n*. With this option, when a program finishes using a tape drive, the drive does not rewind the tape.
- ▲ To indicate tape density, append **lo**, **loc**, or **c** to the device name. Refer to Table 4-2 and Table 4-3 for the differences among **lo**, **loc**, and **c** in 4-mm and 8-mm tape drives, respectively. These tables use the tape drive in the fifth drive slot as an example and assume that the drive uses the rewind option.

Always use **rast** or **nrast** for Exabyte 8200 tape drives because these drives do not use data compression.



Table 4-2. Naming a 4-mm tape drive in slot 5

Tape drive name	Meaning
rast5	Default tape density, which is set by a hardware switch on the drive. If the switch is set to ON, the drive uses compression. If the switch is set to OFF, the tape accepts data in either low-density or compression format, depending on the format of the existing data on the tape. If the tape is blank, data is written in low-density format.
rast5c	Forced compression (WangDAT compression), overriding the hardware switch on the drive.
rast5lo	Without compression, overriding the hardware switch on the drive.
rast5loc	Not applicable.

Table 4-3. Naming an 8-mm tape drive in slot 5

Tape drive name	Meaning (Exabyte 8500 drives)	Meaning (Exabyte 8505 and 8505XL drives)
rast5	Exabyte 8500 format without compression	Exabyte 8500 format without compression
rast5c	Not applicable	Exabyte 8500 format with compression
rast5lo	Exabyte 8200 format without compression	Exabyte 8200 format without compression
rast5loc	Not applicable	Exabyte 8200 format with compression

Table 4-4. Naming a DLT4000 tape drive in slot 7

Tape drive name	Meaning
rast7	DLT4000 format without compression
rast7c	DLT4000 format with compression
rast7lo	DLT4000 format without compression
rast7loc	DLT4000 format with compression

For more information on how data compression affects tape capacity, refer to Chapter 8.

#### **Raw Disk Partition**

Where a command refers to a raw disk partition, which gives an application direct access to the partition, the device name is preceded by an  $\mathbf{r}$ , as in  $\mathbf{rad7c}$ .



## **Device Special Files**

The Auspex operating system software is shipped with all necessary device special files.

If you need to create more Auspex device special files, use the MAKEDEV.auspex command. Refer to the MAKEDEV.auspex(8) man page for details.

If you want to create device special files for the SCSI port located on the HP, refer to "Attaching Drives to the Host Processor SCSI Port" on page 4-35.

To ensure that devices exist, boot the system and watch for the kernel to recognize each device immediately after loading vmunix. The kernel displays a single line for each device, which contains the device name and hardware information. (For disk devices, this information is displayed if the disk drive is formatted and labeled; an unlabeled drive is listed as such.) This information is also saved to the file /var/adm/messages.

An example of the boot sequence is provided in Appendix E.



## Using CD-ROM Drives

Auspex ships its system software on CD-ROM; however, NetServer software is installed at the factory, and it is usually unnecessary to reinstall the software when you take delivery of the system. The CD-ROM provides a convenient, permanent medium for repairing a damaged root drive.

For instructions on installing the system software, refer to "Using NSinstall to Install Software From CD-ROM" on page 6-29. To repair a damaged root drive, first boot the NetServer from the CD-ROM in single-user mode (refer to "Booting the Server at the Monitor Prompt" on page 3-4), and then refer to "Recovering a Damaged Root Disk" on page 6-26.

Slot 0 is always reserved for the root disk drive. The device name for CD-ROM drives installed in the NetServer is **acd***n*, where *n* is a number in the range of slot numbers available in your server, and is the slot number of the drive. The Auspex device driver, acd(4), conforms to Sony-Philips red-book and yellow-book specifications. The NetServer does not support audio CDs.



**Note:** If you want to boot the NetServer from the distribution CD, by Auspex convention you must have the CD-ROM drive installed in slot 1.

The rest of this section discusses procedures for mounting and unmounting file systems, and ejecting the CD from the drive.

## Mounting or Unmounting File Systems on CD-ROM

CDs are read-only media and have fixed file systems and partitioning schemes. The following describes the CD-ROM file system formats supported by the NetServer and their characteristics:

**HSFS** You can mount an HSFS file system but only on the HP, not as an LFS file

system on an FP. For example, you can mount a file system on the SunOS

CD-ROM as an HSFS file system.

UFS You can mount a file system of type 4.2 or type lfs with an fs=4.2 option.

For example, you can mount a file system on a Catalyst CD-ROM as a UFS

file system.

The file system must be mounted read-only; any attempt to write to the CD-ROM generates an error message. The following instructions provide an example for mounting a CD-ROM:

1. Edit the /etc/fstab file to add an entry for the file system using the read-only (ro) option. In the following example, the CD-ROM in slot 1 is in UFS format and is mounted as an LFS file system:

/dev/acd1 lfs /library fs=4.2,ro

In the following example, the CD-ROM is in HSFS format:

/dev/acd1 hsfs /library ro



**Caution:** If you include the CD-ROM in /etc/fstab, be sure that the CD is in the drive each time the system boots. Otherwise, **fsck** fails at reboot, and the system cannot boot to multiuser mode.





**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

```
#VP256 ENABLED (do not delete)
```

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

2. At the root prompt, create a directory for the file system:

```
mkdir /library
```

3. At the root prompt, mount the file system:

```
mount /library
```

As usual, if the CD-ROM does not need to be mounted each time the system boots, do not edit *fstab* (skip step 1 in the previous example). Simply mount the CD-ROM with the **mount** command as in the following examples (the first for a UFS CD-ROM and the second for an HSFS CD-ROM):

```
mount -rt lfs -o fs=4.2 /dev/acd1 /library
mount -rt hsfs /dev/acd1 /library
```

Unmount the file system as you would file systems on a disk drive; for example:

```
umount /library
```

#### Removing a CD-ROM From the NetServer

You can eject a CD-ROM from the drive in one of two ways:

▲ Enter the **eject acd***n* command at the shell prompt, where *n* is the slot number for the drive. For example:

```
eject acd1
```

ejects the CD-ROM from the drive installed in slot 1 of the NetServer. Depending on the type of CD-ROM drive, the CD is either in a CD-ROM caddy or directly in the CD-ROM drive tray.

If file systems are mounted on the CD-ROM drive, the NetServer first unmounts the file systems and then ejects the CD caddy.

▲ Press the eject button on the CD-ROM drive.

If file systems are mounted on the drive, the CD does not eject. If the button does not work and there are no mounted file systems on the CD, check to see if the jumper setting on the drive is set correctly. The drive must be removed from the NetServer to check the jumper setting. Jumper settings are described in the hardware manual.



**Note:** If you are using a CD-ROM drive that requires a CD-ROM caddy and, if for some reason, you have inserted an empty caddy into the drive, you must use the eject button to remove the caddy; you cannot use the eject command.



## **Disk Partitioning**

Making decisions about disk partitioning and assigning client space are tasks that are vital to the efficient operation of your system. This section describes the default disk partition schemes available for Auspex disks and illustrates how file systems are distributed by default among available partitions on the root drive.



**Note:** CD-ROM drives are read-only devices and cannot be partitioned.

#### **Disk Partition Tables**

Each disk can be divided into seven or fewer partitions named partition **a**, **b**, **d**, **e**, **f**, **g**, or **h**. (Partition **c** is reserved for the entire disk.) The partition sizes are defined in the disk label, which is the disk's first sector.

The NS 7000/800, and NS 7000/700 Series NetServers support 1-GB, 1.35-GB, 1.76-GB, 2-GB, 3-GB, 4-GB, and 9-GB disk drives; the NS 7000/200 supports the 1.76-GB and 4-GB drives only. (Each disk size referenced here refers to the disk capacity after formatting.) The HDDA drive drawer supports only 9-GB drives.

Each disk size has three partition tables to choose from: default, stripe, and root. The number of partitions and the individual partition sizes vary according to the partition table the disk uses.



**Note:** When describing the size of a partition, the term "MB" represents 2,048 disk sectors, which is equivalent to 1,048,576 bytes. When used to describe the size of a disk, MB is 1,000,000 bytes. As a result, the size of the  $\bf c$  partition may look smaller than the size of the entire disk. For example, the  $\bf c$  partition of a 1 GB (1,000 MB) disk is described as 954 MB in the partition table.

In most cases, Auspex labels the disks at the factory using the default partition table. Use this default partition table for clients' root, swap, and home directories, and the stripe drives for striped virtual partitions (see "Different Types of Virtual Partitions" on page 5-4). The root partition table is used for the root disk (ad0).

If the current partitioning scheme on a drive does not meet your needs, use the **format** or **ax\_label** command to repartition your disk. For example, if the 4-GB drive you received from Auspex is partitioned as a data disk and you want to use it as a root disk, use **format** or **ax\_label** to change the partition table to root. Refer to "Using the Format Command" on page 4-36 or "Using ax\_label to Partition and Label a Disk" on page 4-41 for information on how to partition a disk.

Table 4-5 through Table 4-7 list the partition sizes for the default, stripe, and root partition tables as specified in /etc/format.dat, which is the data file used by the format command. Some of the drives listed are no longer available from Auspex, but are still supported with this system software release.



**Recommendation:** Although /etc/format.dat includes a root partition table for 9-GB disks, such disks should not be set up as the root disk. The 9-GB disk is not supported as a root drive.

Table 4-5. Default disk drive partition tables

Partition	9-GB drive	4-GB drive	3-GB drive	2-GB drive <sup>*</sup>	1.76-GB drive	1.35-GB drive	1-GB drive
а	1,445 MB	683 MB	480 MB	334 MB	281 MB	215 MB	159 MB
С	8,668 MB	4,094 MB	2,882 MB	2,000 MB	1,682 MB	1,291 MB	954 MB
d	4,335 MB	2,048 MB	1,441 MB	1,001 MB	842 MB	646 MB	477 MB
g	2,890 MB	1,365 MB	961 MB	667 MB	561 MB	430 MB	318 MB
h	4,333 MB	2,046 MB	1,440 MB	999 MB	840 MB	645 MB	477 MB

<sup>\*</sup> Auspex NetServers use three different types of 2-GB disk drives. Your 2-GB partition tables may differ from those shown here.

Table 4-6. Stripe drive partition tables

Partition	9-GB drive	4-GB drive	3-GB drive	2-GB drive	1.76-GB drive	1.35-GB drive	1-GB drive
а	2,167 MB	1,023 MB	720 MB	500 MB	420 MB	322 MB	238 MB
b	2,167 MB	1,023 MB	720 MB	500 MB	420 MB	332 MB	238 MB
С	8,668 MB	4,094 MB	2,882 MB	2,000 MB	1,682 MB	1,291 MB	954 MB
е	4,334 MB	2,046 MB	1,441 MB	1,000 MB	840 MB	645 MB	477 MB
f	4,334 MB	2,046 MB	1,441 MB	1,000 MB	840 MB	645 MB	477 MB
g	2,167 MB	1,023 MB	720 MB	500 MB	420 MB	322 MB	238 MB
h	2,167 MB	1,023 MB	720 MB	500 MB	420 MB	322 MB	238 MB

Table 4-7. Root drive partition table (in MBs)

Partition	4-GB	3-GB	HP 2.1-GB	Microp 2-GB	HP 2-GB	1.76-GB	1.35-GB	1-GB
а	100	100	50	50	50	50	50	50
b	600	650	650	650	650	500	450	400
С	4,094	2,882	2031	2,000	1909	1,682	1,291	954
d	200	150	120	120	120	120	120	120
е	1000	300	90	80	40	70	30	10
f	1000	1000	800	800	800	650	450	200
g	250	250	220	220	220	220	160	160
h	944	432	98	80	29	70	31	14

Based on the partition sizes defined in the selected partition table (default, stripe, or root), you can partition the disk in different ways. For example, you can partition a 9-GB stripe drive into a, b, g, and h partitions, or you can partition it into e and f partitions. See Figure 4-1 through Figure 4-3 for the different ways to partition a 9-GB disk. (Partitions for other size drives have identical proportions.) In the figures, each bar represents a partition scheme in which the size of each partition is defined by a partition table. The numbers shown along the bottom of the diagram indicate the partition sizes.



To display the partition table for a disk, use the **dkinfo** or **ax\_lslabel** command. The following are examples of these commands:

dkinfo adn
ax\_lslabel adn

where *n* is the disk drive number (a number in the range of available drive slots).

Each partition table is defined as an entry in /etc/format.dat, the data file used by the **format** command. For more detailed information about the tables, see the commented text in /etc/format.dat.

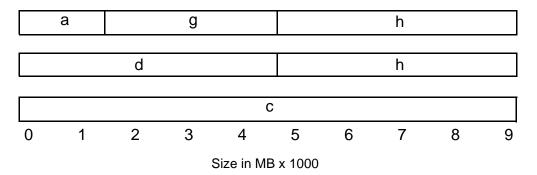


Figure 4-1. Partitioning a 9-GB disk using the default partition table

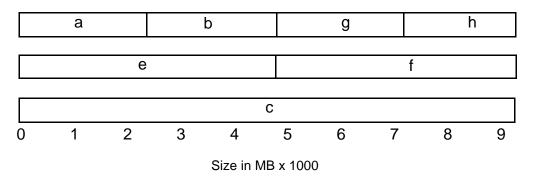


Figure 4-2. Partitioning a 9-GB disk using the stripe partition table

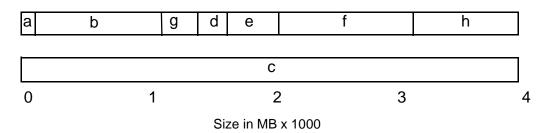


Figure 4-3. Partitioning a 4-GB disk using the root partition table



## **Distribution of File Systems on Default Partitions**

All disks from the factory have default partitions when you receive the disks.

#### Default File Systems on the Root Drive

The factory default partitioning scheme on the root drive places file systems in ad0a, ad0g, ad0d, and ad0f. These file systems are mounted onto the directories /, /usr, /usr/openwin, and /var, respectively.



**Note:** The file system /var/crash can be mounted from another file system, or it may remain in /var. On smaller root drives, /var/crash should be mounted from another file system.

#### **Default File Systems on Default Drives**

All other drives are formatted at the factory and labeled with the default partition table. File systems are initialized using **newfs** on partitions **a**, **g**, and **h**, but these file systems are not mounted to any directories.

If the partitions defined on default drives at the factory do not meet your needs, use ax\_label(8) or format(8) to relabel the data drives and initialize file systems on the appropriate partitions using newfs(8). For more information on these commands, refer to the ax\_label man page, format man page, and newfs man page.



**Note:** Before using **newfs** to initialize a file system on a physical partition, make sure the partition does not overlap another partition that is already mounted or opened. For example, if /dev/ad10h is mounted, trying to use **newfs** to initialize /**dev/rad10c** generates a message similar to the following:

/dev/radlc: cannot create: Device busy

## Naming File Systems

Because any server disk partitioning scheme includes multiple disks and partitions, it is important to develop a logical naming convention to identify the file systems.



**Recommendation:** Do not base file system names on physical device names. Using names other than physical device names makes it easier to reconfigure disk partitions when you use virtual partitions. For example, name the file system used by the engineering department /home/engineering instead of /home/ad9, even though ad9 is the disk on which the file system resides.



## **Example of File System Distribution**

The following list explains the file systems that typically exist on a server. Table 4-8 illustrates how these file systems are distributed.

- ▲ The file systems ad0a, ad0b, ad0g, ad0d, and ad0f are reserved for the server.
- ▲ In the example shown in Table 4-8, the /export file system is in ad7c. This file system contains the architecture-dependent executables that correspond to a diskless client's architecture, allowing clients to boot their architecture-specific operating system. When you execute SetupExec, the directories are loaded into the /export directory.
- ▲ The /export/root and /export/swap file systems contain root and swap directories for several diskless clients. When you execute the **SetupClient** command, it places the client root and swap directories in the partitions you specified in the **SetupClient** form.

In Table 4-8, /export/root and /export/swap reside on virtual partitions that consist of physical partitions on different drives. Using virtual partitions allows you to back up these file systems easily and balance disk access among multiple disks. For more information on the Auspex Virtual Partition Manager, refer to Chapter 5.

However, you can choose to store the clients' *root* and *swap* directories on physical partitions if:

- you prefer to manage smaller file systems because newfs and fsck on these systems take a shorter time to complete than on a large virtual partition
- you want to be able to add client diskless workstations without having to reconfigure a virtual partition

By default, the **SetupClient** program sets up *root* and *swap* in /*exportN* (where *N* is the disk number you specify). For example, you can store one group of *root* and *swap* directories in /*export1* and another group in /*export2*. Distributing client files among multiple disks in this way reduces disk contention.

Under each /exportN, set up a root file system (for example, on the **a** partition) and a swap file system (for example, on the **g** partition). Having separate file systems for swap and root allows you to back up root directories without backing up swap directories.

Analyze the client *root* and *swap* needs of your environment before deciding how to partition the disks on your server. If the partitions defined at the factory do not meet your needs, use **ax\_label** or **format** to customize the disk partitions.



Table 4-8. Example of disk partitioning among multiple disks

Partition	Mount point	File system type <sup>*</sup>	Description
ad0a	/	4.2	root
ad0b		swap	swap
ad0g	/usr	4.2	/usr
ad0d	/usr/openwin	Ifs	OpenWindows 3.0
ad0f	/var	4.2	spool and log files
ad2a	/backuproot	4.2	copy of root partition
ad2g	/backupusr	4.2	copy of /usr partition
ad2h	/home	Ifs	home directories
ad4c	/home/dvlp	Ifs	client data
ad5c	/home/rlse	Ifs	client data
ad6c	/home/src	Ifs	client data
ad7c	/export	Ifs	client exec, share, kvm
vp1 (for example, a striped partition consisting of ad8a and ad9a)	/export/root	lfs	client root directories
vp2 (for example, a striped partition consisting of ad8g and ad9g)	/export/swap	lfs	client swap space
ad9h	/home/sysadmin	Ifs	client data
ad10c	/home/rcs	Ifs	client data

<sup>\*</sup> File systems managed by the File Processor must be mounted type LFS; file systems managed by the HP must be mounted type 4.2. LFS is discussed in "Mounting File Systems on the FP and HP" on page 4-18.

## **Auspex File System Structure**

Figure 4-4 through Figure 4-9 illustrate the Auspex file system structure. The following file systems are shown:

- / (root)
- /var
- /usr
- /usr/openwin
- /export (Your /export file system structure may be different from the one described in this section, depending on your version of the SunOS executables.)
- /exportN (The /exportN file system is not a default file system shipped with the server. To create it, run **SetupExec** and **SetupClient** as described in **Chapter 2**.)

In the figures, an arrow (—>) indicates a symbolic link to the directory following the arrow.

For a more detailed description of the functions of the files mentioned in Figure 4-4 through Figure 4-9, see Sun's System and Network Administration. Note that Auspex modifies the SunOS file structure to some extent (however, from the client's perspective, the modifications are invisible).



This section mentions two file system types, 4.2 and LFS, which appear in /etc/fstab. They are discussed in greater detail in "Mounting File Systems on the FP and HP" on page 4-18.

#### Root file system

Figure 4-4 shows the server's *root* file system. The standard SunOS configuration is modified to contain the directory */auspex*, which contains files specific to Auspex systems. The *root* file system is defined in */etc/fstab* as type 4.2.

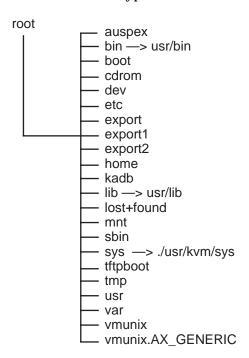


Figure 4-4. The Auspex root file system



#### /var file system

Figure 4-5 shows the server's /var file system. By default, this file system is UFS-mounted (that is, its file system type is defined as 4.2 in /etc/fstab) for these reasons:

- /var contains the /var/adm/messages files to which system messages are written. Mounting this file system as a UFS instead of an LFS file system ensures that the server continues to write system messages to /var/adm even when the File Processor has failed.
- If you mount /var as an LFS file system, using make or yppush to make a new NIS map might generate an error message and corrupt the maps that you try to build. The message is as follows:

Can't bind master to send ypclear message to ypserv for map mapname on the master

By mounting /var as a UFS file system, you can modify NIS maps without this problem.

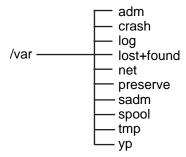


Figure 4-5. The Auspex /var file system



#### /usr file system

Figure 4-6 shows the /usr file system. Although most of the files in /usr are SunOS files, treat them as Auspex-specific software. Because the files in /usr are not intended to run on workstations, /usr is defined in /etc/fstab as type 4.2 and cannot be exported to client machines.

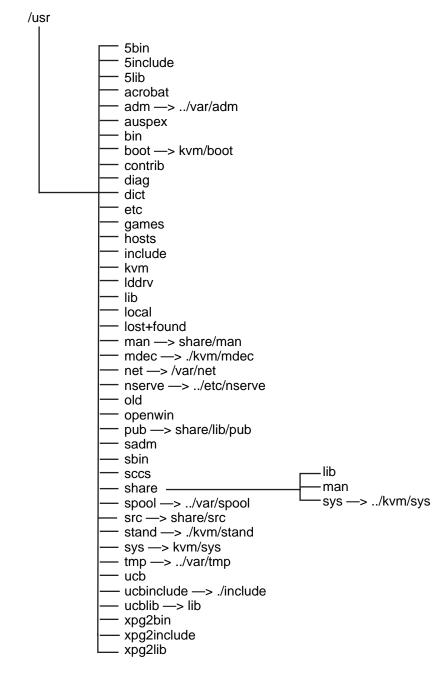


Figure 4-6. The Auspex /usr file system



#### /usr/openwin file system

The /usr/openwin file system contains most of the OpenWindows 3.0 files but not the X11/NeWS server. Consider these files, like the ones in /usr, Auspex-specific software; they are not to be mounted and run by diskless workstations on the network. Diskless Sun or Sun-compatible clients can obtain SunOS and OpenWindows from the /export file system, where the software must be installed from distribution media from Sun or other vendors.

The purposes of /usr/openwin are as follows:

Users with X terminals or workstations on the network can log in to the NetServer to run X applications. For example, the xdm files in /usr/openwin provide a login form for X terminal users on the network to log in to a NetServer. After they log in, they can run X programs such as xterm, or view the online NetServer information.



**Note:** If you want to read the man pages for the OpenWindows commands and files shipped with the Auspex software, set MANPATH to /usr/openwin/man using the following syntax:

#### setenv MANPATH /usr/openwin/man

/usr/openwin contains font files and remote configuration files for an X terminal that is used as the system console. Because /usr/openwin is NFS-exported, an X terminal can fetch the font files and download the configuration file from a NetServer.

Figure 4-7 shows the structure of the /usr/openwin file system.

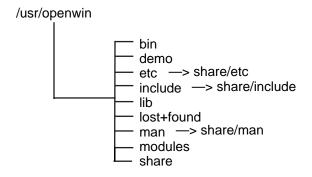


Figure 4-7. The Auspex /usr/openwin file system



#### /export file system

Figure 4-8 shows a sample /export file system, which contains directories that are exported to client systems, including the architecture-dependent executables in /export/exec. The directory structure allows you to support multiple versions of each client architecture.

Because /export directories are exported, the /export file system is assigned the LFS file system type in /etc/fstab.

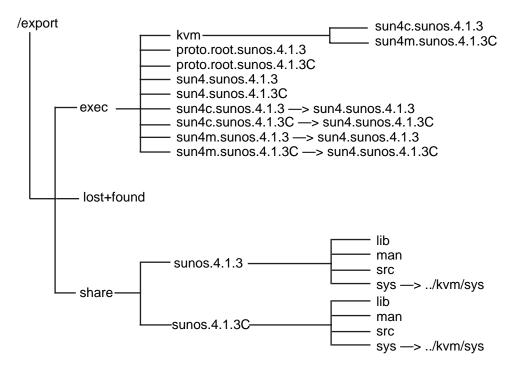


Figure 4-8. The /export file system for servers (with SunOS executables)

#### /exportN file system

Figure 4-9 shows the /exportN file system structure (where N represents the disk number). The /exportN directories contain the client root and swap file systems. Because these directories are exported, this partition is assigned the LFS file system type in /etc/fstab.

The /exportN file system is unnecessary if you use one virtual partition for /export/root and another for /export/swap. Refer to "Example of File System Distribution" on page 4-11 for more information on how to configure partitions for client root and swap directories.

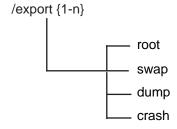


Figure 4-9. The /exportN file system for clients

For information on how to configure clients on the system, see "SetupClient" on page 2-27.



## Mounting File Systems on the FP and HP

This section describes mounting file systems on the FP and HP. It also describes LFS, which is specific to Auspex servers, the modifications that Auspex made to the /etc/fstab file format, and the commands related to LFS.

### File Systems Mounted on the File Processor

The FPs, which support the LFS file system, handle file system requests from the NPs. For UNIX to mount file systems managed by the FPs, it must use a file system type specific to the Auspex NetServers called Local File System (LFS). This is the file system type specified in the /etc/fstab file for the mounted file system. LFS is much like NFS, but the requests go across the VME backplane instead of over the network.

Individual files in LFS file systems can exceed 2 GB in size when available over NFS Version 3. Such files can be as big as the file system itself. Clients accessing files over 2 GB in size must also have large file capabilities.

File systems normally are mounted on the FP. Also, only file systems managed by the FP can take advantage of a NetServer feature called "file system isolation," as described in Chapter 6 Recovering From Disk or File System Failures. This feature allows an LFSmounted file system to go offline if it is corrupted or if there is a disk problem—without affecting the operation of the entire server.

Mount a file system on the HP only if the conditions described in the next section apply.

## File Systems Mounted on the HP

You can mount file systems on the HP if any of the following conditions apply:

- The file system being mounted is not supported by the FP. For example, if the file system is the High Sierra File System or a third-party file system, mount it on the HP.
- If the file system is primarily used by the HP and rarely used by clients of the NetServer, mount the file system on the HP. This enables the applications running on the HP to access the file system faster than they would if they had to access the file system through the FP.
- If you want the HP to be able to use the file system even after the FP has failed, mount the file system on the HP. For example, in order for the HP to log an FP failure to /var/adm/messages, the /var file system must be mounted on the HP instead of the FP.



**Note:** Operations on file systems mounted on the HP require HP CPU cycles, and the entire SunOS file system and virtual-memory system. As a result, NFS access to these file systems is slower than access to LFS-mounted file systems. Mounting several file systems on the HP runs a greater risk of exhausting HP resources such as CPU cycles and virtual or physical memory. Use the criteria described previously to determine whether it is necessary for the HP to manage the file systems.

In the /etc/fstab file, the file type for file systems mounted on the HP is 4.2.



## Auspex's Modifications to /etc/fstab

This section discusses how /etc/fstab on the NetServer is different from the one shipping with standard SunOS.



**Recommendation:** Print out a copy of /etc/fstab each time you modify the file. If a NetServer drive fails, having a record of the file systems that reside on each drive makes it easier to recover from the failure.



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

#VP256 ENABLED (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

#### Specifying the File System Type

The implementation of the LFS file system type requires a modification to entries in the /etc/fstab file. In general, file systems used only by the server are configured as the 4.2 file system type (for example, / and /usr), while file systems exported to clients are configured as the LFS file system type (for example, /home).

Figure 4-10 shows the default /etc/fstab file shipped with the NetServer. For further details, see the **fstab**(5) man page.



	1	pass count fiel						
		so that / and /u fstab(5), which		-		_		
	# accive. (see	ISCAD(5), WILLCI	ı ıllay üll	rer rrom	Sunos I	SLO	aD(5).)	
	# /dev/ad0a	/	4.2		1 0			
	#/dev/ad0a #/dev/ad0b	,		rw	0 0			
	,	swap	swap 4.2	rw	1 0			
File systems ——	/dev/ad0g /dev/ad0f	/usr	4.2	ro	0 2			
managed by the	/dev/adui /dev/ad0d	/var	4.2 lfs	rw f- 4 0	1 2			
FP and exported	,,	/usr/openwin		fs=4.2				
to clients are	/dev/ad0e	/export	lfs	fs=4.2	1 2			
mounted as type LFS. File systems	/dev/ad0h	/home	lfs	fs=4.2	1 2			
managed by the	#			. 1				
HP are of type		e is usually occ	upled by	the CDR	OM			
4.2.	#		1 6			_	0	
	#/dev/acd1	/cdrom	hsfs	ro,noau	to	0	0	
	#							
	# note: by convention, slot 2 is the "backup" root drive							
	#							
	#/dev/ad2a	/.BackupRoot		4.2	ro		0	
	#/dev/ad2g	/.BackupRoot/us		4.2	ro		0	
	#/dev/ad2f	/.BackupRoot/va		4.2	ro	-	0	
The set-UID and -	#/dev/ad2d	/.BackupRoot/us	r/openwi	n 4.2	ro	0	0	
set-GID bits on	#			▼				
program files on	# /dev/ad4a	/export1/root	lfs	fs=4.2,			2	
this file system	# /dev/ad4g	/export1/swap	lfs	fs=4.2,	WC	0	2	
are ignored.	#							
	# /dev/ad5a	/export2/root	lfs	fs=4.2,		_	2	
	# /dev/ad5g	/export2/swap	lfs	fs=4.2,	WC	0	2	
	#							
	# Concatenated	virtual partitio	n - see	/etc/vpa	rtab			
	# /dev/vp1	/home	lfs	fs=4.2,	WC	1	2	
	#							
	# Mirrored virt	ual partition -	see /etc	c/vpartab				
	# /dev/vp5	/disks	lfs	fs=4.2,	WC	1	2	

Figure 4-10. Default fstab file

In *fstab* entries for file systems containing client root directories, be sure to specify the *nosuid* option. This option minimizes security violations, because it prevents users on the clients' systems, who have root permission on their root directories, from creating **setuid** programs in their root file systems and executing them on the server. (Of course, the client's /etc/fstab must be configured without the *nosuid* option so the client has root permission on its own root directory.)

#### Mapping LFS File Systems to File Processors

If the server contains multiple FPs, it needs to decide which FP to use to mount a file system. You can either specify the FP assignment in /etc/fstab or let the server map the file systems to the FPs.



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

```
#VP256 ENABLED (do not delete)
```

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.



#### Using /etc/fstab to assign file systems to FPs

To specify the FP on which a particular file system should be mounted, use the fp=n option in /etc/fstab. The n in this option is the FP number. The default FP number is 0; you do not need to use this option if you want to mount a file system on FP0.

For example, to mount the /cad file system on FP1, include the following entry in /etc/fstab:

```
/dev/ad10c
            /cad lfs
                        fs=4.2, fp=1 1
```

#### Automatic file system mapping

If no *fp=n* option is specified in /etc/fstab, the server automatically maps file systems to FPs when you execute the **mount** -a command. However, if /etc/fstab contains an fp=n option, the server assigns the FPs using the default method as described in the "Using /etc/fstab to assign file systems to FPs" section. For example, if /etc/fstab contains the following entries, the server assigns /backupdata and /disks to FP0 and /cad to FP1:

```
/dev/ad3g
              /backupdata
                             lfs
                                      fs=4.2
                                                  1
                                                         2
                                                         2
/dev/ad4q
                             lfs
                                     fs=4.2
                                                  1
             /disks
/dev/ad3q
                             lfs
                                     fs=4.2, fp=1 1
                                                         2
             /cad
```

Also, if you specify an invalid FP number in the fp=n option, the server automatically assigns the file system to FP0 and displays two warning messages. For example, if your server contains fewer than four FPs and you specify **fp=3** for the /export file system in /etc/fstab, the server assigns the file system to FP0 and generates these messages:

```
mount_lfs: /dev/ad3e on /export: Invalid file processor specified
mount_lfs: invalid FP3 specified, mounting /export on FP0
```



**Note:** If no *fp=n* option is specified in */etc/fstab*, the **mount -a** command distributes file systems evenly on the FPs. This distribution, however, does not guarantee that the load is evenly spread out over all FPs because the amount of activity on each file system varies. You might need to assign file systems manually using the *fp=n* option if some file systems are more heavily used than others.

If you want to see the mapping between FPs and file systems while you are mounting the file systems, use the v (verbose) option in the mount command. The following is an example of **mount** -av and its output:

```
mount -av
mount: mounting /export on FP0
mount: mounting /home on FP1
mount: mounting /home/export on FP0
```

You can also see the FP assignment after you have mounted the file systems. Simply enter **mount** without any options. The following is an example of the **mount** command and its output:

#### mount

```
/dev/ad3e on /export type lfs (fs=4.2, fp=0)
/dev/ad3h on /home type lfs (fs=4.2, fp=1)
/dev/ad0e on /home/export type lfs (fs=4.2, fp=0)
```



#### Enabling Write Acceleration in /etc/fstab

Another Auspex modification to /etc/fstab is related to write acceleration. To enable write acceleration for a file system, modify the entry for the file system in /etc/fstab by adding wc. The following is an example showing the wc option for the /home/dvlp file system:

```
/dev/ad30c
              /home/dvlp
                             lfs wc, fs=4.2 0
```

For more information on write acceleration and its requirements, refer to Chapter 7.

## Other Commands Affected by LFS

In addition to the LFS file system type and the *fp=n* option to /etc/fstab, an lfs option has been added to the -t argument of **mount**(8), **df**(8), and **umount**(8) to specify the LFS file system type.

#### File Processor Constraints

The following list shows File Processor constraints:

- Each FP supports a maximum of 256 mounted LFS file systems.
- If the number of operations varies greatly from one file system to another, balance the load among FPs manually by adding the *fp=n* option to the appropriate entries in /etc/fstab.

## File System Formats Supported by UFS

The format of a UFS file system on the HP or an FP can be either of two types:

- Standard UNIX 4.2 BSD. This format is sometimes called the Fast File System (FFS) or Static Table Format.
- Fat Fast File System (FFFS). This format is sometimes called the Dynamic Table Format or the Tahoe File System. It is the default format for LFS file systems.

The primary advantage of the FFFS format is that it supports more inodes per cylinder group and more cylinders per cylinder group than the FFS format. (FFS is limited to a maximum of 2048 inodes and 16 cylinders per cylinder group.)

The **fsck**(8) command has an option that allows you to convert from FFS to FFFS file systems. If you execute fsck(8) with the -c option, the specified file system is converted from FFS to FFFS format.

The output of the **dumpfs**(8) command indicates which format (dynamic or static) is used by the file system being examined, and tunefs(8) handles both formats.

For more information, refer to the **fsck**(8), **dumpfs**(8), and **tunefs**(8) man pages.



## **NFS Version 3 and UFS Limitations**

Client operating systems query the rpc daemon about NFS Version 3 support. With the addition of NFS Version 3 in release 1.10, the client receives a **yes** to the query. However, NFS Version 3 on an Auspex NetServer does not work with UFS file systems. Any UFS file system exported to and mounted by a client must now be explicitly mounted on the client with NFS Version 2. Some client operating systems, such as a PC with NT, do not have the option to explicitly mount NFS versions. Such systems query for NFS Version 3, then appear to mount UFS file systems from the NetServer, but access to the file systems fails. See your client machine's documentation for mount options that specify NFS protocols.



## Specifying a Host Name in /etc/fstab

For a NetServer to access an exported file system with root privilege, the fstab entry for the file system must use the name of the appropriate network interface of the NetServer that exports the file system. The entry must not use the name for the entire NetServer.

For example, you want to export /home/engineering from netserver1 to netserver2 with root permission. If **netserver1-e5** is the Ethernet interface attached to the network that is also attached to netserver2, the fstab entry for /home/engineering on netserver2 must include the following text:

netserver1-e5:/home/engineering /home/engineering

If you specify netserver1 instead of netserver1-e5 in the entry, netserver2 cannot access the file system. The host name used in the entry is the name that maps to the IP address of the network interface on the server that exports the file system (netserver1-e5 in this example). You can obtain the name of an interface from the NIS database or the /etc/hosts file on the server on which you modify *fstab* if NIS is not used at your site.



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.



# Adding or Replacing a Drive

The changing needs of your site may require that you add more disk, tape, or CD-ROM drives. This section describes the procedure used to configure new drives in the system. The procedure includes such functions as:

- ▲ installing a new drive or replacing an existing drive
- ▲ checking the partition table on a new disk drive
- ▲ initializing a new file system on the new disk drive
- mounting and exporting the file system to client workstations



**Note:** See the *DriveGuard Manager's Guide* for adding or replacing drives in a DriveGuard array.

Drives are added into vacant slots in an installed HDDA drawer or drive rack. The slots in an HDDA drawer accept only disk drives. The slots in a drive rack accept the following types of drives:

- ▲ disk drives
- ▲ 4-mm, 8-mm, or 1/4-inch tape drives
- ▲ CD-ROM drives
- ▲ DLT4000 tape drives (supported in a DLT4000 drive rack only)

If you do not have enough vacant slots to accommodate the number of drives you want to add, you may have an additional drive rack or HDDA drive drawer installed (provided your server can accommodate more racks or an expansion cabinet). This upgrade can be arranged through your Auspex representative. For a detailed description of the available drive configuration options, refer to the *hardware manual* for your NetServer.

The instructions in this section assume that your NetServer has enough vacant drive slots to accommodate the drive(s) to be installed.

Two ways exist to install a new drive, depending on whether the NetServer's operating system is booted. The following procedures provide instructions for using both installation methods.



## Installing or Replacing a Drive After Booting

This section explains how to install a new drive or replace an existing drive while the NetServer is powered on and the operating system is booted. This procedure does not apply to the root drive. To replace the root drive, follow the procedure for replacing a disk drive when the operating system is shut down.



**Note:** If you are installing or replacing a CD-ROM drive, by convention Auspex recommends installing the first CD-ROM in slot 1. Install additional CD-ROM drives in any slot except slot 0, which is reserved for the root disk drive, and in an HDDA drawer.

#### **Before You Start the Drive Installation Procedure**

Follow these steps before you start drive installation while the operating system is booted:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 1. Log in as root on the system console.
- 2. If you are replacing an existing drive, make sure no partition on the drive is open for read or write access. For example, verify that:
  - no file system is being dumped.
  - no process such as a database is using a raw partition.
  - no swapping is being done to the disk.
  - no partitions are members of other virtual partitions. Use the ax\_diskconf command to get a report that maps all file systems to the physical partitions or virtual partitions they reside on.

Depending on the type of partition, you may need to do one or more of the following:

- If a virtual partition or a CD-ROM partition is mounted, first unmount it.
- For striped or concatenated virtual partitions that are not part of a mirrored virtual partition, edit /etc/vpartab to comment out the virtual partition, and then run ax\_loadvpar so the system reads the revised table. See "Virtual Partition File" on page 5-9.
- For a mirrored virtual partition, edit /etc/vpartab to comment out the entry for the member on the drive being replaced and redefine the mirror so it has only one member. Finally, run ax\_loadvpar so the system reads the revised table. See "Virtual Partition File" on page 5-9.





**Caution:** When you hot-plug a drive, make sure no active tape drives are on the same SP. Trying to add a drive while a tape drive on the same SP is open for read or write access generates an error message. To determine if a tape drive is busy, use the **ax\_tapestats** command. If you try to run **ax\_tapestats** on a busy device, the message <code>Device busy</code> appears.

#### Using ax\_hot\_plug to Add a Drive to the Server



**Note:** For information on adding, removing, or replacing HDDA drives, refer to your hardware manual.

The **ax\_hot\_plug** command allows drives to be hot-plugged; that is, added or removed without shutting down the NetServer. You can run **ax\_hot\_plug** only on the local console. If you try to run it from a remote login session, the following error message appears:

ax\_hot\_plug: must be on /dev/console to use this command



**Caution:** Never run **ax\_hot\_plug** on an SP that is formatting a drive or running **ax\_mrestore** or **ax\_clonefs**. Doing so may hang the system.

After completing the steps in the preceding section, follow this procedure to add a drive to the server:

1. Enter the **ax\_hot\_plug** command, specifying the slot number of the drive to be added or replaced. For example, if you wish to add or replace a drive in slot 6, enter the following:

```
# ax_hot_plug add 6
```

If a tape drive on the same SP is active, the server displays the following error message:

ax\_hot\_plug: a tape drive is in use on the same SP as the device to be added or changed; cannot replace drive.

This message ensures that no active tape drives are rewound because of the power surge in the drive rack when a drive is added. If the message appears, you must invoke **ax\_hot\_plug** again after all tape drive activity on the same SP has stopped.



**Note:** The **ax\_add\_device** command is also used for adding drives to the NetServer. Refer to the **ax\_add\_device** man page for more information.

When **ax\_hot\_plug** executes successfully, the system suspends SP activity to prevent I/O operations on the SCSI bus and prompts you to insert or replace the specified disk drive. The command then tells you to type a carriage return after inserting the new drive.



**Note:** On a quiescent SP, all I/O operations are halted. NFS requests received from the NetServer's NPs are halted until the SP restarts. This may cause timeouts on client workstations making NFS requests, so it is important to install the new drive as quickly as possible.

- 2. Install the new drive.
- 3. Type a carriage return immediately after installing the new drive.



The system restarts the SP and attempts to attach the drive, which takes approximately 20 seconds. If the drive does not respond (if it is not readable or if it is not properly connected), the system generates an error message.



**Caution:** If you see an error message indicating that the newly installed drive is not found or that a SCSI interface error has occurred, remove the drive from the slot *immediately.* The sooner you remove the drive with the wrong SCSI ID, the less likely the server will crash. Repeat the drive installation procedure only after verifying or correcting the drive's SCSI ID.

Follow the procedure for configuring a new disk drive.

## Installing or Replacing a Drive When the Operating System is Shut Down

- 1. Refer to your NetServer's hardware manual for guidelines on handling disk drives.
- If the operating system is booted, use /etc/shutdown -lh to shut down the system.
- When the NetServer console displays a PROM monitor prompt and if you are replacing an existing drive, remove the existing drive from the slot into which you plan to install the new drive.
- Install the new drive, and reboot the system.
- During the boot process, observe the boot messages to verify that the system detects the presence of the new drive. Unless the drive label has been corrupted, the drive appears in the list of devices during the boot process.

## Configuring a New Disk Drive

1. Determine your partition scheme (refer to "Disk Partitioning" on page 4-7). Also, plan ahead for whatever virtual partitioning scheme you use.

The configuration worksheets in Appendix D may be helpful as you gather disk drive information.



**Note:** Disk drives are labeled at the factory, so a partition table is already assigned to each disk. Unless you are changing the standard partition configuration, you only need to install a UNIX file system using the **newfs**(8) command, as described in step 3.

2. Check to see if the disk drive label is correct (all disk drives are shipped already labeled). To do this, enter the following command:

# dkinfo adn

where n is the disk drive number. This command displays the disk label, which lists the size of the usable UNIX partitions on the disk. For a 4-GB disk in slot 8, the dkinfo command and the resulting display look like this:



#### # dkinfo ad8

```
ad8: Auspex SP at addr 1180, unit # 8
4094 cylinders 16 heads 128 sectors/track
a: 2095104 sectors (1023 cyls: 0-1022)
b: 2095104 sectors (1023 cyls: 1023-2045)
c: 8384512 sectors (4094 cyls: 0-4093)
d: No such device or address
e: 4190208 sectors (2046 cyls: 0-2045)
f: 4190208 sectors (2046 cyls: 2046-4091)
g: 2095104 sectors (1023 cyls: 2046-3068)
h: 2095104 sectors (1023 cyls: 3069-4091)
```

Capacity i	n 100	megabytes
------------	-------	-----------

0	5 10	15	20	25	30	35	40
	e		C 				
	e	l <sub>n</sub>		g		h	
	a   a	b		g		h	

See "Disk Partitioning" on page 4-7 for partition descriptions. If the label has a problem or if an existing label needs to be changed, use **ax\_label** or **format** to correct the problem.

3. Run **newfs**(8) on the desired partition(s) on the disk. Enter the following command:

```
# newfs /dev/radnp
```

In this command, n is the disk drive number and p is the partition letter. The device name ad has an r, prefix, specifying that it is a raw (character) device. newfs can only be run on raw devices. For more details, see the newfs(8) man page.

4. Create the directory that will serve as the mount point for the partition. For example:

```
# mkdir /home/src
```

5. Add an entry to /etc/fstab to create a permanent mount entry for the partition. Using the previous directory as an example, add the following entry:

```
/dev/ad8c /home/src lfs fs=4.2 0 2
```

If your NetServer has multiple FPs, you can specify the FP on which to mount the file system. For example, you can create the following entry:

```
/dev/ad8c /home/src lfs fs=4.2,fp=1 0 2
```

If you want to use write acceleration for this new partition, the entry reads as follows:

```
/dev/ad8c /home/src lfs wc,fs=4.2, fp=1 0 2
```

The server automatically distributes file systems to FPs if you do not specify any FPs in /etc/fstab and if you execute the **mount** -a command. For more information on mapping file systems to FPs, refer to "Mapping LFS File Systems to File Processors" on page 4-20.





**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 6. Mount the file system. For example, to mount the file system created in step 4, enter the following command:
  - # mount /home/src



**Note:** If the new file system is an LFS file system, you must enter the **mount** command with only one argument. This format causes **mount** to read /etc/fstab. If you enter two arguments (that is, if you specify both device and mount point in the command), the file system is mounted as file system type 4.2, and all operations on this file system are controlled by the HP, not the FP.

- 7. Enter the mount command again, without arguments. This form of the command generates a list of all mounted file systems. Check the output to verify the new file system was mounted properly.
- 8. Add an entry to /etc/exports to export the newly created file system. For example: /home/src
- 9. Export the file system by entering the **exportfs** command as follows:
  - # exportfs -a
- 10. Enter the **exportfs** command again, without arguments. This form of the command generates a list of all exported file systems. Check the output to verify that the new file system was exported properly.

The disk drive is now ready for use.



# Removing a Drive

If you need to remove a drive and do not plan to replace it with another drive right away, use the following procedures.

Remove a device in one of two ways, depending on whether the operating system is booted.



**Note:** For information on adding, removing, or replacing HDDA drives, refer to your hardware manual.

## Removing a Drive When the Operating System is Booted

This procedure does not apply to the root drive. To replace the root drive, follow the procedure for replacing a drive while the operating system is shut down on page 4-28.

#### **Before You Remove a Drive**

Follow these steps before you remove a drive from the server:

- 1. Refer to the *hardware manual* for your NetServer for guidelines on handling drives.
- 2. Log in as root on the system console.
- 3. If you are replacing an existing drive, make sure no partition on the drive is open for read or write access. For example, verify that:
  - ▲ no file system is being dumped.
  - ▲ no process such as a database is using a raw partition.
  - ▲ no swapping is being done to the disk.
  - ▲ no partitions are members of other virtual partitions. Use the **ax\_diskconf** command to get a report that maps all file systems to the physical partitions or virtual partitions they reside on.

Depending on the type of partition, you may need to do one or more of the following:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

```
#VP256 ENABLED (do not delete)
```

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- For a mounted physical disk or CD-ROM, unmount the disk or CD-ROM.
- ▲ For striped or concatenated virtual partitions that are not part of a mirrored virtual partition, edit /etc/vpartab to comment out the virtual partition, and then run ax\_loadvpar so the system reads the revised table (see "Virtual Partition File" on page 5-9).
- ▲ For a mirrored virtual partition, edit /etc/vpartab to comment out the entry for the member that is on the drive being replaced, and redefine the mirror so it has only one member. Finally, run ax\_loadvpar so the system reads the revised table (see "Virtual Partition File" on page 5-9).





**Caution:** When you hot-plug a drive, make sure that no active tape drives exist on the same SP. Trying to remove a drive while a tape drive on the same SP is open for read or write access generates an error message. To determine if a tape drive is busy, use the **ax\_tapestats** command. If you try to run ax\_tapestats on a busy device, the message Device busy appears.

### Using ax\_hot\_plug to Remove a Drive

Run ax\_hot\_plug only on the local console. If you try to run it from a remote login session, the following error message appears:

ax hot plug: must be on /dev/console to use this command



**Caution:** Never run **ax\_hot\_plug** on an SP that is formatting a drive or running **ax\_mrestore** or **ax\_clonefs**. Doing so may hang the system.

After completing the preceding procedure, follow these steps to remove a drive from the server:

1. Enter the ax\_hot\_plug command, specifying the slot number of the drive to be removed. For example, if you wish to remove the drive in slot 6, enter the following:

# ax\_hot\_plug remove 6

If a tape drive on the same SP is active, the server displays the following error message:

ax\_hot\_plug: a tape drive is in use on the same SP as the device to be removed; cannot remove device.

This message ensures that no active tape drives are rewound because of the power surge in the drive rack when a drive is removed. If the message appears, invoke **ax\_hot\_plug** again after verifying that no tape drive activity is taking place on the same SP.

The system suspends the activity on the SP so no I/O operations in progress on the SCSI bus, spins down the drive, prompts you to remove the specified drive, and tells you to type a carriage return when you finish.



**Note:** On a quiescent SP, all I/O operations are halted. NFS requests received from the NetServer's NPs are halted until the SP restarts. This may cause timeouts on client workstations making NFS requests, so it is important to remove the drive as quickly as possible.

2. When prompted, remove the drive.



**Caution:** Do not remove a disk while it is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive. Wait until the system console prompts you to remove the device.



3. Press the Return key immediately after you have removed the drive. The system restarts the SP.



**Note:** Once you remove a drive from a given slot, as described in this procedure, the SP does not recognize any drive inserted into that slot until you use the **ax\_hot\_plug** command to inform the SP that a drive is to be added.

- 4. Edit /etc/fstab to comment out entries in the table that refer to the removed drive.
- 5. If you detached a member of a mirrored, striped, or concatenated virtual partition, as described in "Before You Remove a Drive" on page 4-31, you should recreate the member on another drive.



**Note:** You can also use the **ax\_remove\_device** command for removing drives from the NetServer. Refer to the **ax\_remove\_device** man page for more information.

## Removing a Drive When the Operating System is Shut Down

The following procedure describes how to shut down the operating system and remove a drive:

- 1. Refer to the *hardware manual* for your NetServer for guidelines on handling drives.
- 2. Edit /etc/vpartab and /etc/fstab to comment out entries in the table that refer to the drive to be removed.
- 3. Use the /etc/shutdown command to shut down the operating system.
- 4. Remove the drive.
- 5. Reboot your system.

## Using ax\_hot\_plug to Add and Remove Multiple Drives

You can use **ax\_hot\_plug** to add and remove multiple drives with a single command.



**Note:** For information on adding, removing, or replacing HDDA drives, refer to your hardware manual.

Run **ax\_hot\_plug** only on the local console. If you try to run it from a remote login session, the following error message appears:

ax hot plug: must be on /dev/console to use this command



**Caution:** Never run **ax\_hot\_plug** on an SP that is formatting a drive or running **ax\_mrestore** or **ax\_clonefs**. Doing so may hang the system.

1. Enter the **ax\_hot\_plug** command, specifying the slot number(s) of the drives to be added or removed. For example, if you wish to add devices in slots 15 through 17 and remove a device in slot 9, enter the following:

```
# ax hot plug add 15-17 remove 9
```

The system suspends the activity on the SP so no I/O operations are in progress on the SCSI bus, spins down the drives, prompts you to add or remove the first specified



drive (in this example, adding a drive to slot 15), and tells you to type a carriage return when you finish adding or removing the drive.

When prompted at the system console, add or remove the specified drive.



**Caution:** Do not remove a disk while it is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive. Wait until the system console prompts you to remove the device.

Type a carriage return immediately after the specified drive is added or removed.

After typing a carriage return, the system restarts the SP. If a drive was added, the system attempts to attach the drive, which takes approximately 20 seconds. When prompted at the system console, repeat steps step 2 and step 3 for each drive specified in the ax hot plug command. The command addresses each drive in the order that it is entered in the command line.

If the drive or drives do not respond (if they are not readable or if they are not properly connected), the system generates an error message.



**Caution:** If you see an error message indicating that any newly installed drive is not found or a SCSI interface error has occurred, remove the drive from the slot immediately. Such an error message is possibly caused by an incorrect SCSI ID. The sooner you remove the drive with the wrong SCSI ID, the less likely the server will crash. Repeat the drive installation procedure only after you have verified or corrected the drive's SCSI ID.

- 4. Follow the procedure for configuring a new disk drive on page 4-28.
- Edit /etc/fstab to comment out entries in the table referring to the removed drive.

If you detached a member of a mirrored, striped, or concatenated virtual partition in "Before You Remove a Drive" on page 4-31, re-create the member.



# Attaching Drives to the Host Processor **SCSI Port**

The SCSI port on the HP supports the standard SunOS devices listed in Table 4-9, not Auspex devices. The system kernel, by default, allows you to attach up to seven Sunsupported devices to the SCSI port, which include four disk drives, two tape drives, and one CD-ROM drive.



**Note:** Devices attached to the HP SCSI port cannot be used as boot devices. Only Auspex devices attached to an SP are bootable.

You must set the appropriate SCSI ID for each device on the HP, or the system kernel cannot recognize the device. The SCSI IDs that you can use are listed in Table 4-9. Be sure each device has a unique ID.

Table 4-9. SCSI IDs and names for devices on the HP

Device	SCSI ID	Device name
Disk 0	3	sd0
Disk 1	1	sd1
Disk 2	2	sd2
Disk 3	0	sd3
Tape 0	4	rst0
Tape 1	5	rst1
CD-ROM 0	6	sr0

Follow these steps to attach a SCSI device to the HP SCSI port:

- 1. Shut down and halt the NetServer.
- 2. Verify that the device being attached has the correct SCSI ID. Refer to the device manufacturer's documentation for instructions on verifying and changing the SCSI ID.
- 3. Attach the device to the port using the SCSI cable described in the hardware manual for your NetServer.
- 4. Power on the NetServer, and boot it to multiuser mode. During the boot process, observe the messages printed on the console to verify the new device is detected by the system. The following sample messages show a tape device attached to the HP:

```
esp0 at SBus slot 1 0x800000 pri 3
st0 at esp0 target 4 lun 0
st0: <Archive QIC-150>
```

Before you remove a device from the HP SCSI port, be sure to check that the server is halted.



# **Using the Format Command**

This section explains how to use the **format** command, which is a menu-driven command for setting up your disks. Under normal circumstances, such as when installing a new system, you do not need to use **format**. Auspex disks are shipped formatted and labeled with the default partition tables. You generally need to use **format** only if a disk is corrupted or if you want to change the drive partitioning. Also, if you need to partition a drive, Auspex recommends using the **ax\_label** command, which is a more straightforward command for achieving the same purpose as **partition** and **label** in the Format menu. For more information on **ax\_label**, refer to "Using ax\_label to Partition and Label a Disk" on page 4-42.



**Caution:** Low-level formatting of a disk is normally not necessary. Drives are preformatted at the factory before shipment. When formatting a drive, use extreme care, as existing data can be erased from the disk. Be sure to back up any data on the disk before formatting it.

## **Running the Format Command in Single- or Multiuser Mode**

The **format** command is usually run in multiuser mode. However, in multiuser mode, if the disk drive you want to format is **ad0**, you must run **format** after booting from the CD-ROM or an alternative root device.



**Recommendation:** For faster recovery from a disk or media error on the root disk, keep an up-to-date backup of key partitions of the root disk on a separate disk; for example, ad2. Additionally, make tape backups of all key data files, including root file systems. This procedure is described in "Backing Up the Root Disk" on page 8-11.



**Caution:** Do not use the **format** command for DriveGuard arrays. See the *DriveGuard Manager's Guide* for using **ax\_label**(8) with DriveGuard arrays.

## **Accessing the Format Menu**

- 1. Make sure the disk on which you want to run **format** is installed in the system. For installation instructions, see "Adding or Replacing a Drive" on page 4-25.
- 2. Execute **format**, and select the disk you want. In the following example, the NetServer has a total of 8 disk drives; the user selects **ad4**. (Enter the disk number in the form of **adn**, where *n* is the slot number.)

```
# format
```

```
Searching for disks...done

AVAILABLE DISK SELECTIONS:

ad0 ad1 ad2 ad3 ad4 ad5 ad6 ad7

Specify disk number (enter its name): ad4

selecting ad4: <SEAGATE 4GB>
[disk formatted, no defect list found]
```



```
FORMAT MENU:
       disk
                  - select a disk
       type - select (define) a disk type
       partition - select (define) a partition table
       current - describe the current disk
        format
                  - format and analyze the disk
        repair
                  - repair a defective sector
        show - translate a disk address label - write label to the disk
                  - translate a disk address
        analyze - surface analysis
        defect - defect list management
                 - search for backup labels
        backup
        quit
format> t
```

The **format** program usually automatically recognizes the type of the disk selected. (The exception is when format fails to recognize a valid label on the drive.) In this example, ad4 is a 4-GB disk.

For a complete description of the options in the **format** menu, see Sun's *System and Network* Administration. The following sections describe only the options that display menus different from those on a Sun server.



**Caution:** Before using the **format** command to change the size of a physical partition or format a disk, verify the partition or disk is *not* a member of a virtual partition. The **format** command may not issue a warning message when you attempt to format a disk that is part of a virtual partition.

## **Defining a Disk Type**

The **type** option in the **format** menu displays a list of drive types from which you choose one appropriate to the disk to be formatted. While the menu lists disks from various vendors, Auspex only supports Hewlett-Packard drives, Seagate drives, and Micropolis drives. The **type** option displays following list of drives:

```
AVAILABLE DRIVE TYPES:
        0. ....
        ...(drives 0-10 are not supported in the NetServer)
        11. HP 663MB
        12. HP 1GB
        13. HP 1.35GB
        14. MICROP 1.76GB
        15. MICROP 1.76GB NON-ALIGNED
        16. HP 2.0GB
        17. MICROP 2.0GB
        18. HP 2.1GB
        19. MICROP 3GB
        20. Seagate 4GB
        21. Seagate 4GB NON-ALIGNED
        22. MICROP 4GB
        23. MICROP 9GB
```



```
24. Seagate 9GB
25. Seagate 9GB NON-ALIGNED
26. other
Specify disk type (enter its number) [20]:
```

Auspex uses the formatted capacity to define the disk type. Normally, you only need to accept the default drive type because **format** automatically recognizes the type for the selected drive. However, if you plan to use the **dd** command to copy contents between two drives of different capacity, specify the size of the source drive when you use the **type** option on the target drive to avoid confusing the operating system. For example, if you plan to copy from a 1-GB drive to the target drive, which is 1.35-GB, select HP 1GB from the menu for the target drive so it is formatted as a 1-GB drive.

### Formatting a Disk Using the Format Option

Low-level formatting of a disk is normally not necessary. Drives are preformatted at the factory before shipment.



**Caution:** This procedure erases all existing data on the disk. Be sure to back up the data before formatting the drive.

- 1. If you have not already done so, follow the steps outlined in "Accessing the Format Menu" on page 4-36. Do not use the format command for DriveGuard arrays. See the *DriveGuard Manager's Guide* for using ax\_label(8) with DriveGuard arrays.
- 2. From the Format menu, enter **format**. The **format** option applies to the disk that you specified earlier. The following example shows screen output generated by **format**:

```
format> format
Ready to format. Formatting cannot be interrupted and
takes xxx minutes (estimated).
Continue? y
Beginning format. The current time is Tue Feb 15 16:58:05 1995
Formatting ... done
Verifying media ...
  pass 0 - pattern = 0xc6dec6de
  pass 1 - pattern = 0x6db6db6d
Total of 0 defective blocks repaired.
format>
```

xxx (in minutes) varies with the type of drive. The process of formatting and verifying can take up to several hours, depending on the size of the drive. To ensure the integrity of the disk, you should allow this process to complete.



**Caution:** Interrupting the format of a drive causes the drive to enter an irrecoverable state. The drive must be reformatted.

When the process is finished, the disk is formatted. By default, the entire drive is formatted and two passes of surface analysis are run on the disk. If any defects are found in the surface analysis, the system repairs them automatically. Table 4-10 on page 4-39 lists the approximate amount of time it takes to verify the format of each drive type.



Table 4-10. Disk drive format verification time

Drive type	Time to format and verify format (in minutes)
9.0 GB	310
4.0 GB	120
3.0 GB	130
2.0 GB	130
1.76 GB	130
1.35 GB	82
1.0 GB	90
663.0 MB	64

After formatting the disk, partition the drive using the **partition** option as described in the next section. Alternatively, you can use **ax\_label** to set up partitions and write the label to disk, as described in "Using ax\_label to Partition and Label a Disk" on page 4-41.

### Changing the Partitioning on a Disk

The following procedure describes how to partition a disk using the **partition** option in the Format menu:

- 1. Unmount all file partitions on the disk. Make sure no virtual partitions residing on the disk remain mounted.
- 2. If you have not already done so, follow the steps outlined in "Accessing the Format Menu" on page 4-36. Do not use the format command for DriveGuard arrays. See the *DriveGuard Manager's Guide* for using ax\_label(8) with DriveGuard arrays.
- 3. Select the **partition** option on the format menu. The partition menu appears with a set of options relating to disk partitioning:

```
format> partition
PARTITION MENU:
a- change 'a' partition
b- change 'b' partition
c- change 'c' partition
d- change 'd' partition
e- change 'e' partition
f- change 'f' partition
g- change 'g' partition
h- change 'h' partition
select- select a predefined table
name- name the current table
print- display the current table
label- write partition map and label to the disk
quit
partition>
```

To use a predefined partitioning scheme on the disk, enter the **select** option. A list of supported disk partition types appears. If the NetServer identifies a disk drive that has a partitioning scheme not specified in /etc/format.dat, format denotes this scheme



as **original ad***n*, where *n* is the disk number. You can use this scheme to partition another drive of the same size as **adn**. For more information on these partition types, refer to "Disk Partitioning" on page 4-7. The following is the Select menu for a 4-GB disk drive:

```
partition> select
   0. Seagate 4GB
   1. Seagate 4GB STRIPE
   2. Seagate 4GB ROOT
   3. Original ad4
   Specify table (enter its number) [0]:
```

4. Select a disk partition type from the list displayed. The partition table is created.

```
Specify table (enter its number) [0]: 2
partition>
```

5. Use the **label** option to label the current disk.

```
partition> label
Ready to label disk, continue? y
format> q
```

If you want to examine the partition table you have created, enter print. For further information on the print option, see Chapter 6 of Sun's System and Network Administration.



# Using ax\_label to Partition and Label a Disk

To save time, use **ax\_label** instead of **format** to partition and label a disk if you know the exact size of each physical partition.



**Caution:** Use **ax\_label** on Auspex disk drives only. Do not use **ax\_label** for disk drives attached to the SCSI port on the HP.

Labeling a disk means writing the partitioning table and various disk geometry constants to the label sector (sector 0) of the disk and informing the **ad** device driver of these changes.



**Caution:** As with **format**, make sure no file systems are mounted on the drive on which you are going to partition using the **ax\_label** command. Also, labeling a drive can erase all data on the drive; back up the files on the drive before using **ax\_label** if you want to preserve them.

The syntax for **ax\_label** is as follows:

```
ax_label [ -abcdefgh ] size disk
```

The option preceded by a dash is the partition name. Specify the size in MB immediately after each partition name; a size of **all** means that all remaining sectors in the disk volume are assigned to the specified partition. You can define up to seven partitions. Specify the disk number (for example, **ad2**) at the end of the command.



**Note:** In the **ax\_label** command, the partition names are usually entered in alphabetical order. However, if you assign a size of **all** to a partition, specify it as the last partition regardless of its name.

The following is an example for partitioning a 4-GB disk:

```
ax_label -a 400 -b 1580 -d 300 -e 3200 -f 2000 -g 500 -h all ad5
```

This example divides ad5 into seven partitions. After all sectors are assigned to partitions a, b, d, e, f, and g, the remaining space is assigned to partition h.

The following example shows partitioning **ad5** with only the **c** partition:

```
ax_label -c all ad5
```

The following example shows partitioning three disks (**ad7-ad9**) identically with 200-MB **a** partitions and 300-MB **b** partitions, with the remaining sectors assigned to the **h** partition:

ax label -a 200 -b 300 -h all ad7 ad8 ad9



# Displaying a Disk's Partitioning Information

This section describes the ax\_lslabel command, which displays information about physical disk partitions. (If you want to see comprehensive reports about file systems or virtual partitions, use the ax\_diskconf command, which is described in "Displaying Disk Configuration Information" on page 5-23.)

The syntax for **ax\_lslabel** is as follows:

```
ax_lslabel [ -sc ] all|disk
```

The -c option prints out the disk information in a concise format; the -sc option prints out the information in "superconcise" format. The following examples show the ax\_lslabel commands and their outputs:

```
hostname% ax 1slabel ad0
ad0 has 3818 cylinders (16 heads and 64 sectors/track per cylinder)
a: 32768 sectors (32 cyls, 0-31)
b: 204800 sectors (200 cyls, 32-231)
c: 3909632 sectors (3818 cyls, 0-3817)
d: 131072 sectors (128 cyls, 488-615)
e: 624640 sectors (610 cyls, 616-1225)
f: 204800 sectors (200 cyls, 1226-1425)
q: 262144 sectors (256 cyls, 232-487)
h: 2449408 sectors (2392 cyls, 1426-3817)
Capacity in 100 megabytes
  2 4 6 8 10 12 14 16 18
ab| g |d| e | f|
hostname% ax lslabel -c ad0
#device:start cylno:nblks
/dev/rad0a:0:32768
/dev/rad0b:32:204800
/dev/rad0c:0:3909632
/dev/rad0d:488:131072
/dev/rad0e:616:624640
/dev/rad0f:1226:204800
/dev/rad0g:232:262144
/dev/rad0h:1426:2449408
hostname> ax_lslabel -sc ad0
#device:sizeA:sizeB:sizeC:sizeD:sizeE:sizeF:sizeG:sizeH
ad0:32:200:3818:128:610:200:256:2392
```



# **5** Virtual Partitions

# **About This Chapter**

This chapter describes the *Virtual Partition Manager*, which comprises these commands and files that create and manage different types of virtual partitions supported by the NetServer:

- ▲ vp
- **▲** vpartab
- ▲ ax\_diskconf
- ▲ ax\_expand
- ▲ ax\_loadvpar
- ▲ ax\_mattach
- ▲ ax\_mdetach
- $\blacktriangle$  ax\_mconvert
- ▲ ax\_mrestore
- ▲ ax\_vold
- ▲ ax\_vpstat

This chapter also enumerates the advantages and limitations of using virtual partitions on your server.



## **Virtual Partition Overview**

The Auspex NetServers offer the capability to group one or more physical disk partitions from one or more disk drives into a single virtual partition.

Virtual partitions allow you to:

- create disk partitions larger than the maximum size of a single physical disk.
- use all available disk space by combining all unused "runt" partitions into a usable size.
- distribute high-use file systems over multiple disk drives to reduce disk latency because of request queueing.
- define mirrored partitions to protect file systems from disk or media failure and provide high data availability.
- back up file systems without taking the file systems offline.



## Maximum Size of a Virtual Partition

The Auspex virtual partition breaks the 2-GB limit traditionally imposed by UNIX systems. A virtual partition can have up to 16 members with each member partition limited in size only by the size of the disk holding that member. As an example, 16 members taking up 16 9-GB disks give a virtual partition of 144 GB.



**Note:** The size of an indivitual file in an LFS file system is limited only by the size of the file system when accessed through NFS Version 3. Clients accessing such files must have large file capabilities.

# Advantages of Using a Greater-than-2-GB **Virtual Partition**

The benefits of using a virtual partition depend on the type of partition. The various types of virtual partitions are described in the following sections. Regardless of the partition type, defining partitions that exceed 2 GB offers these advantages:

Easier file system management

You now have the flexibility of grouping a large number of files in one file system, which allows you to use fewer **dump** and **restore** commands for system backups. Also, there are fewer file systems to export.

More efficient use of disk space

You can pool more "chunks" of small disk space that are otherwise unusable into one large virtual partition. You do not need to worry about exceeding the file system size limit while adding small physical partitions to a large virtual partition.



**Note:** If you enter the **df** command on a client to display the amount of free disk space on the server, you might see negative numbers in the df output for virtual partitions that are larger than 2 GB. The following example shows the **df** output containing negative numbers:

```
proto:/disks/proto -1046719 -1950330 169422 97%
/disks/proto/sp2vp4
```

However, if you enter the **df** command at the server, the number of kilobytes are displayed correctly. For example:

```
/dev/vp150
            7341889 6437142 170558
                                        97%
/disks/proto/sp2vp4
```



# **Different Types of Virtual Partitions**

Auspex supports three types of virtual partitions: concatenated, striped, and mirrored. Several commands are available for you to define virtual partitions and recover from disk or media failures that affect virtual partitions. These commands are described in "Virtual Partition Commands" on page 5-10.

You can set up virtual partitions on disks only; you cannot use virtual partitions on CD-ROM or tape drives.

#### **Concatenated Partition**

A *concatenated partition* is a concatenation of 1 to 16 physical disk partitions from one or more disk drives, allowing you to accommodate a file or file system larger than a single physical disk drive or assemble several small unused partitions into a single partition large enough to be useful.

For example, in Figure 5-1, the physical partitions ad7c, ad8c, and ad9a are members of a concatenated partition, vp1. (The physical partitions that make up a concatenated partition do not need to be the same size.)

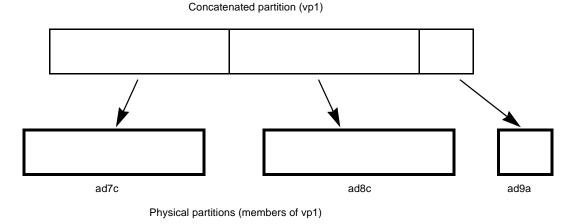


Figure 5-1. Example of a concatenated partition of three physical partitions

## **Striped Partition**

A *striped partition* consists of 2 to 16 physical disk partitions. The striped partition maps its logical address space across multiple physical partitions in fixed-length segments referred to as stripes. For example, in Figure 5-2, a striped partition (**vp2**) consists of four physical partitions of equal size.

Striping distributes access requests on heavily used file systems over multiple drives. Unlike concatenated partitions, adjacent blocks in a striped partition are mapped to blocks on different disks, thus reducing access latency and improving throughput to a given file system.



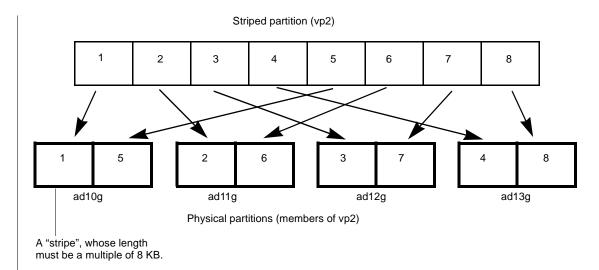


Figure 5-2. Example of a striped partition of four members

#### **Mirrored Partition**

A mirrored partition consists of one or two members. Each member must be of the same size as the mirrored partition. Each member must be a virtual partition, and if there are two members, they should be located on different physical drives to achieve greater reliability. The drives, however, must be supported by the same SP.

The data written to the mirrored partition is queued and written to both members as soon as possible. As a result, using a mirrored partition ensures that critical file systems remain available even if a member of the mirror fails.



**Note:** Using mirrored partitions is not a valid substitute for performing backups. It is still important to back up all data to tape.

When data is read from the mirrored partition, the SP chooses the member from which the read can be more quickly performed, using a short-queue, short-seek algorithm. Thus, although a mirrored write takes slightly longer than an unmirrored write, a mirrored read is faster than an unmirrored read.



Caution: Normally, if one storage component such as a member disk of a mirrored partition fails during a write operation, the data can still be written to the partition, and no data is lost. However, if write caching is used for the file system on a mirrored partition, a failing Write Accelerator might lead to a loss of data because data in the Write Accelerator's NVRAM is not mirrored.

For example, after data has been successfully written to the write cache, a write completion is returned to the application that initiated the write request. However, if the write cache fails *before* it can write the data to both members of the partition, there is no way for you to retrieve the data from the write cache to update the mirrored partition. Therefore, if you want to increase data reliability of a file system, mount it on a mirrored partition without using the write cache.

Figure 5-3 illustrates a mirrored partition (vp3) whose two members are concatenated partitions. Each of the concatenated partitions consists of one physical partition.



(However, the member of a mirrored partition can be a striped partition, and it can be made up of several physical partitions.)

In this example, if ad9 fails, the server can continue to write to and read from the file system on vp3 using ad8.

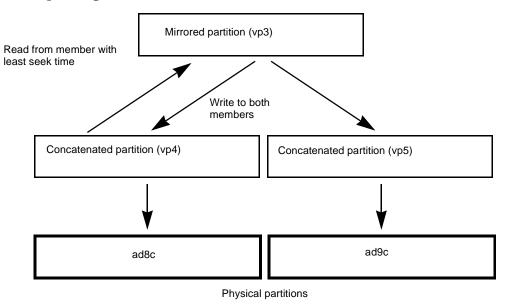


Figure 5-3. Example of a mirrored partition



## Virtual Partition Restrictions

When you set up a virtual partition, remember the restrictions described in the following sections.

#### Restrictions Common to All Kinds of Virtual Partitions

- The root and /usr partitions cannot be members of any virtual partitions.
- A physical disk partition may not be a member of more than one virtual partition.
- Each SP can manage a maximum of 256 virtual partitions. (NetServer software can support up to five SPs and 1280 virtual partitions.)
- All members of a virtual partition must be managed by the same SP.
- If a disk partition is a member of a virtual partition, no disk partition that overlaps that member can be part of another virtual partition.
- Operations that open a drive or a partition with write permission cannot be run on a virtual partition or on a disk partition that is a member of a virtual partition.
  - For example, you cannot run the **format** command on a virtual partition, nor can you format a disk partition that is a member of a virtual partition. You must format and label the disk drive before creating any virtual partitions on it.
  - Similarly, you cannot run newfs, ax\_clonefs, or NSinstall on a disk partition that underlies a virtual partition. For example, if you try to run newfs on ad8c while ad8a is a member of a virtual partition, an error message similar to the following appears:

```
/dev/rad8c: cannot create: Device busy
```

- When a member of a virtual partition starts at the beginning of a disk (that is, if the member is partition a, c, d, or e), the first 8 KB of disk space is not used as part of the virtual partition in order to preserve the disk label.
- The physical partition c (the complete disk) is one cylinder larger than any combination of other physical partitions whose size theoretically takes up the whole disk. Therefore, a copy operation of a c partition from a single partition disk to a multipartition disk of the same size fails.

## Restrictions Applicable to Concatenated and Striped Partitions

- The disk partitions in a concatenated partition need not be of equal size. However, the members of a striped partition must be the same size. Refer to "Disk Partitioning" on page 4-7 for more information.
- You can specify the size of a stripe in a striped partition, but the size you choose must be a multiple of 8 KB (8192 bytes), up to a maximum size equal to the partition size.
  - The recommended size of a stripe is 128 KB.
- The size of each member of a striped partition should be a multiple of the stripe size; if not, the partition size is truncated automatically to the nearest multiple of the stripe size. For example, in Figure 5-4, part of ad7g, ad8g, ad9g, and ad10c is truncated because the length of each of these physical partitions is not a multiple of the stripe size.



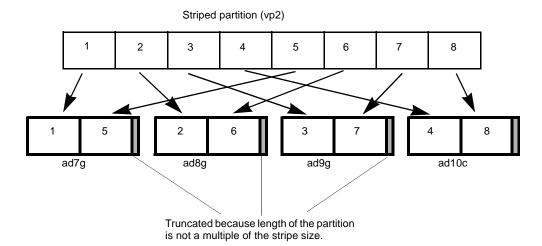


Figure 5-4. Disk space truncated from members of a striped partition

If a member of a virtual partition starts at the beginning of a disk (that is, if the member is partition a, c, d, or e), the first 8 KB of disk space is not used as part of the virtual partition to preserve the disk label. As a result, when deciding whether the size of this kind of member partition is a multiple of the stripe size, remember to consider the 8 KB of disk space. For example, if the stripe size is 32 KB and the member partition (ad1a) is 128 KB, the virtual partition uses only 120 KB on ad1a, and 24 KB of disk space is truncated to make the size of ad1a divisible by 32K.

- The data in a concatenated or striped partition may be spread over a maximum of 16 disk partitions.
- Each member of a concatenated or striped partition must be a valid physical disk partition. A striped partition must have two or more members; a concatenated partition with only one member is permissible.

## **Restrictions Applicable to Mirrored Partitions**

- Each member of a mirrored partition must be a valid virtual partition (either concatenated or striped); a member cannot be a physical disk partition or another mirrored virtual partition.
- A mirrored partition may not have more than two members.
- A virtual partition may not be a member of more than one mirrored partition.
- The members of a mirrored partition should be of equal size.
  - If you create a mirrored partition with members of unequal size, the portion of the larger member that exceeds the size of the smaller member is ignored. For example, if one member contains 50 MB and the other contains 52 MB, only the first 50 MB of the larger member will be recognized; the last 2 MB is never used.
- Each member of a mirrored partition resides on a separate disk drive.
- Both member virtual partitions must be on the same SP, as well as on the same SP as the mirror itself.



## Virtual Partition Driver, File, and Commands

This section describes the driver, file, and commands that manage virtual partitions.

#### **Virtual Partition Driver**

**vp**(4) is the device driver that acts as the interface to virtual partitions constructed from physical disk partitions.

#### Virtual Partition File

**vpartab**(5) is a file containing the table that describes virtual partitions. Edit this file to add or modify virtual partition definitions.

Figure 5-5 show an example of the file /etc/vpartab.

```
# Virtual partitions 1 and 2 - Concatenated
/dev/vp1
                   concat
                                                 ad1h,ad2h,ad4h
/dev/vp2
                                                 ad4d,ad5c
                     concat
# Virtual partition 3 - Striped
/dev/vp3
                     striped, size=128k
                                                 ad7h,ad8h
# Virtual partition 4 - Concatenated
# This is necessary because physical partitions cannot be
# mirrored, and virtual partition 5 is constructed by
# mirroring physical partition ad8c and virtual
# partition 3 (defined above).
                                                 ad9c
/dev/vp4
                     concat
# Virtual partition 5 - Mirrored
                                                 vp3,vp4
/dev/vp5
                     mirrored
```

Figure 5-5. Sample vpartab entries

Comment lines begin with a pound symbol and are ignored. Each virtual partition definition in the table consists of the following tab-separated fields:

1. The first field contains the device name of the virtual partition (for example, /dev/vp1).

You must number the virtual partition according to the SP that controls the partition. Table 5-1 lists the valid partition numbers.

Table 5-1. Numbering virtual partitions

SP that manages the virtual partition	Virtual partition numbers for SP boards
SP0	0 – 255
SP1	256 – 511
SP2	512 – 767
SP3	768 –1023
SP4	1024 –1279

- 2. The second field indicates what kind of virtual partition is being defined. If the partition is a striped partition, it also indicates the stripe size. For example, if you specify striped, size=128K, the first 128 KB of the virtual partition is from the first physical partition, the next 128 KB is from the second physical partition, and so on.
- The third field lists the physical disk partitions making up the virtual partition (for example, ad1h, ad2h, ad4h).

#### **Virtual Partition Commands**

The following list describes the commands and a daemon for creating and managing virtual partitions:

- ax diskconf(8) prints a report showing the current disk configuration of the NetServer. The report shows the physical and virtual partition to which each file system belongs. For more information on ax\_diskconf(8), refer to "Displaying Disk Configuration Information" on page 5-23.
- ax expand(8) enlarges a striped or concatenated virtual partition without first unmounting the file system. For more information on ax\_expand, refer to "Expanding a Concatenated or Striped Partition" on page 5-15.
- ax\_loadvpar(8) reads vpartab and informs vp (the virtual partition driver) and the SPs of the current virtual partitions. ax\_loadvpar checks the validity of each entry in **vpartab**. If errors are found, **ax loadypar** reports the errors and does not load the table.
- ax\_mattach(8) attaches a second member partition to a mirrored virtual partition that currently has one member. It is useful for adding a temporary member to a mirrored partition for backup purposes. The new member is automatically recorded in /etc/vpartab.



**Note:** Before version 1.8 software, the new member was manually entered in /etc/vpartab. Now that it is done automatically, do not manually edit /etc/vpartab. Editing the /etc/vpartab file to add a member partition, such as vp6 to the mirrored partition vp41, and then running ax\_mattach, returns a warning message similar to the following:

WARNING: vpartab edit not done, partition already has members. Please verify state of vp41 vpartab entry using ax\_vpstat.

When this warning occurs, ax\_mattach attaches the new member but does not edit /etc/vpartab.



- ax\_mconvert(8) converts a striped or concatenated virtual partition into a onemember mirrored partition. For more information on ax\_mconvert, refer to "Expanding a One-Membered Mirrored Partition" on page 5-18.
- ax mdetach(8) detaches a member partition from a mirrored virtual partition that currently has two members. It is typically used for removing a member that was added with ax\_mattach for taking a snapshot of the partition. The change is automatically recorded in /etc/vpartab.



**Note:** Before version 1.8 software, the new member was manually removed in /etc/vpartab. Now that it is done automatically, do not manually edit /etc/vpartab. Editing the /etc/vpartab file to remove a member partition, such as **vp6** from the mirrored partition **vp41**, and then running **ax\_mdetach**, returns a warning message similar to the following:

WARNING: /etc/vpartab out of sync: vp6 missing from vp41? verify vp41 state with ax vpstat and update /etc/vpartab.

When this warning occurs, ax\_mdetach detaches the member but does not edit /etc/vpartab.

- ax mrestore(8) updates a member of a mirror after a disk or media failure affected the member. This command runs automatically if the system detects that the two members are not identical at boot time or when ax\_mattach is run. You may occasionally run ax mrestore manually from, either to back up a mirrored partition to another drive or to update a damaged member. Be aware of the following when you use ax\_mrestore:
  - **ax mrestore** may delay the server's responses to other commands. For example, an **ls** or **find** command may take longer if **ax\_mrestore** is in progress. The longer response time is more obvious when multiple mirrored partitions are being restored simultaneously. To reduce the slowdown, use the -m option to assign a "medium" priority to the ax\_mrestore process. After lowering the priority for **ax\_mrestore**, the SP services other system requests faster but takes longer to restore a mirrored partition.
  - You cannot run ax\_mattach or ax\_mdetach while ax\_mrestore is in progress.
  - ax\_mrestore copies data in 192-KB blocks. If there is an NFS request for writing to a block being copied by ax\_mrestore, the SP places the request in a queue and performs the NFS write after **ax\_mrestore** finishes copying the 192-KB block.



Caution: Do not hot-plug devices using ax hot plug, ax remove device or **ax\_add\_device** on the SP that is running **ax\_mrestore**. Doing so may hang the system.

- **ax\_vold**(8) is the daemon that maintains a table showing the current state of all members of all mirrored partitions, based on messages received from the SPs, vp, LFS, and **ax\_mrestore**.
- ax\_vpstat(8) displays information about all virtual partitions. It shows the type and components of a virtual partition and the stripe size of a striped virtual partition. If the partition is a member of a mirror, it also shows the status of the partition. See Figure 5-6 for an example of using the **ax\_vpstat** command.

```
# ax_vpstat
vp65: Concatenated
    ad21c, ad22c
vp68: Mirrored
        vp66: Striped, size=128K, ACTIVE
             ad34c, ad35c, ad36c
        vp67: Striped, size=128K, ACTIVE
             ad37c, ad38c, ad39c
vp77: Mirrored
        vp75: Concatenated, SYNCED
        ad30c
        vp76: Concatenated, SYNCED
        ad31c
vp271: Striped, size=128K
        ad57c, ad58c
```

Figure 5-6. Example from using the ax\_vpstat command

The meaning of each mirrored status is as follows:

ACTIVE	This partition is identical to the other member of the mirror (if the mirror has more than one member), and no unrecovered write error occurred on the partition. The mirror is currently mounted read/write or open for writing.
SYNCED	This status is similar to ACTIVE status, but the partition is not currently mounted read/write or open for writing.
DIRTY	This partition is not identical to the other member of the mirror because the system crashed while the mirrored partition was ACTIVE or RESTORING. Writes to the mirror are not being performed on this member partition.
DAMAGED	An unrecovered write error occurred on this partition and was detected when the SP tried to access the partition. Because of the error, this partition is not identical to the other member of the mirror. Writes to the mirror are not being performed on this member. This status is reset to ACTIVE after you repair or replace the disk and run <b>ax_mrestore</b> to update the data on the partition.
RESTORING	This partition is not identical to the other member of the mirrored partition, but writes to the mirrored partition are being performed on this partition. The ax_vpstat display indicates how much of the restore is complete, expressed as a percentage. The RESTORING condition occurs when the partition is being updated by the ax_mrestore command.

For more information, refer to the man pages for **ax\_diskconf**, **ax\_expand**, **ax\_loadvpar**, **ax\_mattach**, **ax\_mconvert**, **ax\_mdetach**, **ax\_mrestore**, **ax\_vold**, and **ax\_vpstat**.



# **Automatic Error Recovery on Mirrored Partitions**

After a server crashes and reboots, it automatically recovers errors on a mirrored partition to ensure that both members of the partition contain the same data. (However, if disk errors occur on a mirrored partition during normal operations of the server, you need to correct the errors manually using the procedures and commands described in Chapter 6, "Recovering From Disk or File System Failures".)

Figure 5-7 illustrates the sequence of events and changes in a mirrored partition's status before and after a system crash. In this example, vp1 and vp2 are the two members of a mirrored partition.

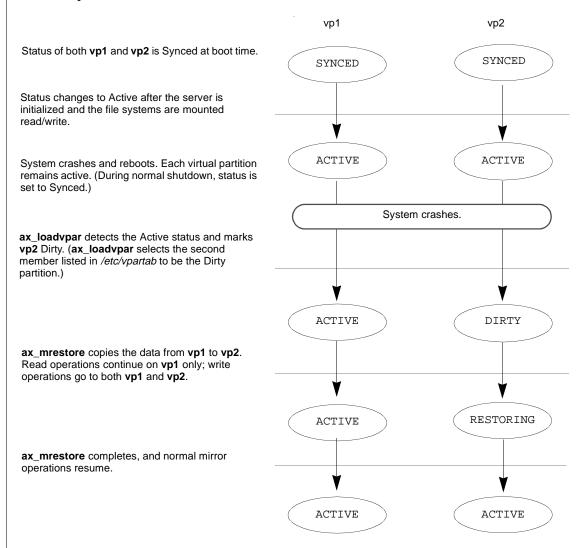


Figure 5-7. Automatic recovery from mirrored partition error



**Caution:** If at any point you want to stop **ax mrestore**(8) and return the state of the partition being restored to DIRTY, use the **kill**(1) command with no signal arguments. That is, use only the process ID of ax\_mrestore in the kill command. Using kill with a signal argument terminates ax mrestore immediately, leaving the partition stuck in the RESTORING state until the NetServer is rebooted.



# **Defining and Reconfiguring Virtual Partitions**

This section describes how to use the virtual partition commands to create and manipulate virtual partitions. The procedures described include the following; most of them require you to log in as root:

- ▲ Defining a virtual partition
- ▲ Expanding a concatenated or striped partition
- Changing a striped or concatenated partition to a mirrored partition
- ▲ Adding a new member to an existing mirrored partition
- Removing a member from an existing mirrored partition



**Recommendation:** Print out a copy of /etc/vpartab each time you modify the file. If one of your NetServer drives fails, having a hard copy record of the virtual partitions (and their members) that reside on the failed drive makes it easier to recover from the failure.

## **Defining a Virtual Partition**

Use this procedure to create a new virtual partition:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 1. If there is an active file system on the physical disk partition you are going to define as a virtual partition, unexport and unmount the partition to prevent access. Change the /etc/fstab and /etc/exports files as necessary.
- 2. Back up the file system to tape or to another drive.
- 3. If necessary, use **format** to repartition the disks.
- 4. Edit /etc/vpartab to add an entry for the new virtual partition. Refer to Table 5-1 on page 5-10 for the list of valid partition numbers and corresponding SPs.
- 5. Run ax\_loadvpar. If ax\_loadvpar detects errors in /etc/vpartab, the command reports the errors and does not load the new table. If this happens, edit /etc/vpartab again to correct the errors and run ax\_loadvpar again.
- 6. Run **newfs**(8) on the new raw virtual partition on which a file system will be mounted (that is, do not run **newfs** on virtual partitions that are members of a mirrored partition). For example, if **vp1** and **vp3** are members of a mirrored partition named **vp2**, enter the following command to create a file system on **vp2**:

/usr/etc/newfs /dev/rvp2





**Caution:** You must run **newfs** on any raw defined virtual partition before attempting to mount the partition.

You do not need to run **newfs** on the underlying physical partitions.

- 7. Use the **mkdir** command to create a directory for the file system.
- 8. Edit /etc/fstab to define a mount point for the file system.



**Note:** When you modify /etc/fstab to define mount points, do not add entries for individual partitions that are members of a mirrored, striped, or concatenated virtual partition. Only the mirrored, striped, or concatenated virtual partition itself should have an entry in /etc/fstab.

For example, if vp1 and vp3 are members of a mirrored virtual partition named vp2, only vp2 has an entry in /etc/fstab; vp1 and vp3 do not. Similarly, if ad5h and ad6h are members of a concatenated virtual partition named vp4, only vp4 has an entry in /etc/fstab; ad5h and ad6h do not.

- 9. Use the **mount** command to mount the file system on the partition.
- 10. Restore the file system that you backed up to tape in step 2.
- 11. Add an entry to /etc/exports to make the file system available for export.
- 12. Use **exportfs** to export the file system so it is accessible to users.

## **Expanding a Concatenated or Striped Partition**

Through the ax\_expand command, you can enlarge an LFS file system on a concatenated or striped partition without unexporting or unmounting the file system. Network clients using the file system on the virtual partition may not even know that the file system is being expanded. Users might, however, notice a pause or an "NFS server not responding" message while trying to access the file system during the expansion.

To expand a concatenated partition, use this syntax:

```
ax_expand virtual_partition physical_partition
```

To expand a striped partition, use this syntax:

```
ax expand -s virtual partition new virtual partition
```

All the partitions involved in one **ax\_expand** command must be controlled by the same SP. You must be superuser to invoke **ax\_expand**.

The following sections describe how to expand a file system on a virtual partition.



**Note:** A command related to **ax\_expand** named **growfs** enlarges a file system to a specified size. Because **ax\_expand** is a front end to **growfs**, always use **ax\_expand** to increase the size of a file system on a virtual partition.

#### **Expanding a Concatenated Partition**

The procedure for expanding a concatenated partition involves adding one or more physical partitions to the virtual partition. The following example shows adding physical partitions **ad8a** and **ad9a** to the virtual partition **vp1**:

```
ax_expand vp1 ad8a ad9a
```

The **ax\_expand** command automatically edits the <code>/etc/vpartab</code> file to reflect the changes to **vp1**. However, if you have comment lines in <code>/etc/fstab</code> that describe the member partitions of the virtual partition being expanded, remember to update those comments.

The **ax\_expand** command generates the following messages. The file system being expanded is mounted on */tmnt* in this example:

```
about to grow vp1. This may take some time...

/dev/rvp1: 683984 sectors in 668 cylinders of 16 tracks, 64 sectors
Initializing cg 41
Isolating /tmnt for superblock adjustments.

/tmnt released.
```

The message Initializing cg number indicates the cylinder group being initialized; each cylinder group takes about one second to initialize. The server prompt appears again when the expansion process completes. (If you interrupt the ax\_expand command, you might end up with additional space initialized for the virtual partition but not used. The additional space is wasted.)

### **Expanding a Striped Partition**

A striped partition cannot be enlarged directly, instead, it is expanded by replacement. After the expansion, the file system on the striped partition is moved to a new virtual partition that is bigger than the original one.

Expanding a striped partition involves two major steps. First, create a new, bigger striped or concatenated partition. Second, run the **ax\_expand** -s command, which changes the existing virtual partition to a single-member mirrored partition with the new partition as its member. Contents of the virtual partition being enlarged are copied to the new partition.



**Note: ax\_mconvert** (used by **ax\_expand**) must create a new virtual partition to be the member of the mirrored partition. If all virtual partition numbers have already been used, **ax\_mconvert** generates this error message:

```
ax_mconvert: all VP devices in use
```

Figure 5-8 illustrates what happens to a striped partition when you enlarge it. The procedure for enlarging a striped partition follows the figure.



#### Before ax\_expand After ax\_expand vp1 (one-member mirrored partition) vp1 (partition to be enlarged) 5 6 vp2 (new virtual partition) 5 8 ad7a ad8a ad9a ad10a ad11a ad12a ad13a

vp3 (unused virtual partition) 2 5 6 ad8a ad9a ad7a

(unused physical partitions)

Figure 5-8. Expanding a striped virtual partition using ax\_expand -s

To enlarge a striped partition, follow these steps:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

#VP256 ENABLED (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

1. Create a new striped or concatenated partition that is bigger than the one you want to enlarge. (This new partition must *not* be a mirrored partition.) Refer to "Defining a Virtual Partition" on page 5-14 for information about creating a virtual partition. In this procedure, the existing partition is named vp1, and the new partition is named vp2.

The following entries are for **vp1** and **vp2** in /etc/vpartab:

vp1 striped, size=128k ad7a,ad8a,ad9a striped, size=128k ad10a,ad11a,ad12a,ad13a vp2



2. Enter the following command to expand **vp1**:

```
ax_expand -s vp1 vp2
```

Remember to use the -s option; otherwise, the following error message appears:

```
ax_expand: can't grow striped partition directly
```

If the command executes successfully, messages similar to the following appear:

```
About to mattach vp2 to vp1. This will take some time...

Attach and sync complete.

about to grow vp1. This may take some time...

/dev/rvp1: 1953776 sectors in 1908 cylinders of 16 tracks,64 sectors

1000.3MB in 120 cyl groups (16 c/g, 8.39MB/g, 3840 i/g)

Initializing cg 119

Isolating /tmnt for superblock adjustments.

/tmnt released.

vp1 now consists of vp2. vp3 is now free.
```

After this procedure, **vp1** becomes a mirrored partition with one member—**vp2**. Although the contents of **vp1** are copied to new physical partitions, access to the file system mounted on **vp1** is unaffected because the name of the virtual partition remains unchanged. The virtual partition expansion process is completely transparent to users of the file system on **vp1**.

The physical partitions used by **vp1** before **ax\_expand** now belong to a new virtual partition. In the previous example, **ax\_expand** chooses the number 3 to name the new virtual partition simply because the number was not used by other virtual partitions. Because the space in this virtual partition is free, you can use this virtual partition for other purposes. For example, you can make a new file system on it. The following entries are for **vp1**, **vp2**, and **vp3** after the expansion process:

```
/dev/vp1 mirrored vp2
/dev/vp2 striped,size=128 ad10a,ad11a,ad12a,ad13a
/dev/vp3 striped,size=128 ad7a,ad8a,ad9a
```



**Note:** The **ax\_expand** command uses the **ax\_mconvert** command to convert the striped partition you want to enlarge to a mirrored partition. If errors occur during the conversion, **ax\_mconvert** displays error messages. For more information on **ax\_mconvert**, refer to the following section, "Expanding a One-Membered Mirrored Partition".

## **Expanding a One-Membered Mirrored Partition**

This section describes how to expand a one-membered mirrored partition.

The procedure for enlarging a one-membered mirrored partition depends on the type of its member partition. If its member is a concatenated partition, you can add one or more physical partitions to the member. If the member is a striped partition, you can expand it by replacing it with a new member.



#### **Expanding a One-Membered Concatenated Partition**

If the member of the mirrored partition is a concatenated partition, add one or more physical partitions to the member by executing **ax\_expand**. For example:

```
ax_expand vp2 ad14c
```

Remember to use the name of the mirrored partition (vp2 in the above example), not the name of its member in the command line.

#### **Expanding a One-Membered Striped Partition**

If the member of the mirrored partition is a striped partition, use ax expand with the -s option to expand it by replacement. For example:

- 1. Create a concatenated or striped partition (for example, vp3) that is larger than the current member of the mirrored partition (for example, vp2).
- 2. Enter the following command to enlarge the mirrored partition (vp2) by replacing its current member with the new member (**vp3**):

```
ax_expand -s vp2 vp3
```

## **Expanding a Two-Membered Mirrored Partition**

Although the ax\_expand command does not support the expansion of two-membered mirrored partitions, you can expand a file system on a two-member mirrored partition with the following procedure:

- 1. Use ax\_mdetach to detach one member from the mirrored partition. The change is automatically recorded in /etc/vpartab.
- 2. Use ax\_expand to enlarge the mirrored partition (see the description provided in "Expanding a One-Membered Mirrored Partition").
- 3. Use ax mattach -f to add a member to the mirrored partition. The -f flag causes **ax\_mattach** to run in the foreground. This new member is automatically recorded in /etc/vpartab.

The new member must be the same size as the existing member of the partition.



**Note:** The data in the virtual partition is not mirrored until this step is complete.

# Changing a Striped or Concatenated Partition to a **Mirrored Partition**

You can convert a striped or concatenated virtual partition to a one-member mirrored partition without affecting the file system mounted on the virtual partition. Because no unexporting or unmounting of the file system is required, users have uninterrupted access to the file system while the conversion takes place.



**Note:** You cannot convert a striped or concatenated partition directly to a twomember mirrored partition. You can, however, use ax\_mattach to attach a second member to the mirrored partition after the conversion process. Also, you cannot convert a partition if there is a UFS file system mounted on it.



To convert a virtual partition to a mirrored partition, use the **ax\_mconvert** command. Its syntax is as follows:

```
ax_mconvert partition
```

The partition in the command must be a striped or concatenated partition currently defined in /etc/vpartab. Applying the ax\_mconvert to a mirrored partition returns an error message.

The **ax\_mconvert** command creates a new virtual partition and makes it the only member of the mirrored partition. The mirrored partition retains the name of the virtual partition being converted. The following example shows how to convert a striped partition (**vp1**) to a one-member mirrored partition:

```
ax_mconvert vp1
```

The <code>ax\_mconvert</code> command modifies <code>/etc/vpartab</code> to create a new virtual partition (for example, <code>vp2</code>) to be used as the only member of the newly formed mirrored partition, <code>vp1</code>. In this example, <code>vp2</code> is mapped to the physical partitions that <code>vp1</code> used before the conversion. After modifying <code>/etc/vpartab</code>, <code>ax\_mconvert</code> invokes <code>ax\_loadvpar</code> to load the virtual partition table. If errors occur during the loading of the table, messages generated by <code>ax\_loadvpar</code> appear. For more information on <code>ax\_loadvpar</code> errors, refer to the <code>ax\_loadvpar</code> man page.

The following is the *vpartab* entry for **vp1** *before* the conversion:

/dev/vp1 striped, size=128k ad3a, ad4a, ad5a

The following are the *vpartab* entries for **vp1** and its member *after* the conversion:

/dev/vp1 mirrored vp2

/dev/vp2 striped, size=128k ad3a, ad4a, ad5a

Figure 5-9 illustrates the conversion process.

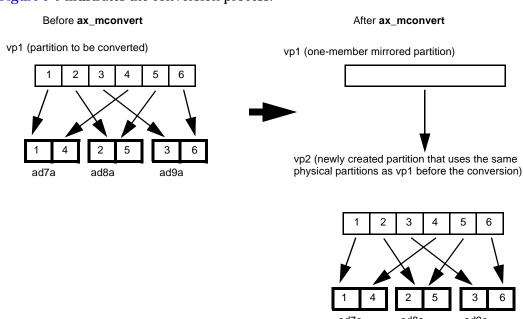


Figure 5-9. Using ax mconvert to change a partition from striped to mirrored





**Note:** Because **ax\_mconvert** must create a new virtual partition to be the member of the mirrored partition, if all virtual partition numbers have already been used, **ax\_mconvert** generates this error message:

ax\_mconvert: all VP devices in use

# Adding a New Member to an Existing Mirrored Partition

If a mirrored partition with a single member already exists, you can add another member to the partition by entering the following command:

ax\_mattach mirrored\_partition new\_member

For example, if you want to add **vp4** as the second member to **vp5**, enter the following:

ax mattach vp5 vp4

The new member is automatically recorded in /etc/vpartab.

If the mirrored partition (in the example, vp5) is mounted, ax\_mrestore starts to copy data from the existing member to the new member of the mirrored partition. Now the mirrored partition consists of two members with the same data.

# Removing a Member from an Existing Mirrored Partition

If a two-membered mirrored partition already exists and you want to remove a member from it, enter the following command:

ax\_mdetach mirrored\_partition old\_member

For example, if you want to remove **vp4** from **vp5**, enter the following:

ax\_mdetach vp5 vp4

The change is automatically recorded in /etc/vpartab.



**Note:** If you are planning to remove the drive containing the detached virtual partition, edit /etc/vpartab to comment out references to the partition (vp4 in the previous example).

Now the mirrored partition consists of only one member.

# Moving a Striped or Concatenated Partition

This section describes how to move a striped or concatenated partition from one group of drives to another. The drives must still be managed by the same SP. The following procedure uses **vp1** as the name of the partition to be moved:

- 1. Use ax mconvert to convert vp1 to a one-membered mirrored partition. The **ax\_mconvert** command creates a new virtual partition (for example, **vp2**) to be the only member of **vp1**. **vp2** comprises the physical partitions that were in **vp1**.
- 2. Create a new virtual partition (for example, **vp3**) consisting of the group of physical partitions to which you want to move the old striped or concatenated partition.



- 3. Use ax\_mattach -f to attach vp3 to the mirrored partition created in step 1. This step starts the ax\_mrestore command, which copies the contents from vp2 to vp3. Go to the next step only after **ax\_mrestore** is finished.
- 4. Use **ax\_mdetach** to detach **vp2** from the mirrored partition.

After this procedure, vp1 becomes a one-membered mirrored partition, with its contents stored on physical disk partitions that make up **vp3**.

# Moving a One-Membered Mirrored Partition

This section describes how to move a one-membered mirrored partition from one group of disks to another. In the following procedure, the mirrored partition is **vp1**; the member of **vp1** is **vp2**:

- 1. Create a new virtual partition (for example, **vp3**) consisting of the group of physical partitions to which you want to move the contents of **vp1**.
- 2. Use **ax\_mattach** -**f** to attach **vp3** to the mirrored partition.
- 3. Use ax\_mdetach to detach the original member, vp2, from the mirrored partition.

The contents of **vp1** are now moved to **vp3**.

## Moving a Two-Membered Mirrored Partition With No Unmount

This section provides a procedure for moving a two-membered mirrored partition. In the following procedure, the mirrored partition is vp1 and its members are vp2 and vp3:

- 1. Create two new virtual partitions (for example, **vp4** and **vp5**), each of which map to the group of physical partitions to which you want to move the contents of vp1.
- 2. Use ax\_mdetach to detach vp2 from vp1.
- 3. Use ax\_mattach -f to attach vp4 to vp1.
- Use **ax\_mdetach** to detach **vp3** from **vp1**.
- Use ax\_mattach -f to attach vp5 to vp1.



**Note:** After each ax mdetach command in the previous procedure, the mirrored partition has only one member. As a result, no mirroring function can take place before the subsequent **ax\_mattach** command is complete.



# Displaying Disk Configuration Information

You can display information about the current configuration of all the disks in the NetServer using the ax\_diskconf command. This command lets you see at a glance how your file systems are arranged on the available partitions and which drives have unused partitions. (If you are only interested in the physical partitions, use the ax\_lslabel command to display the information. Refer to "Displaying a Disk's Partitioning Information" on page 4-42 for more information on ax\_lslabel.)

The ax diskconf report consists of two tables, each displaying disk information in different ways.

The first table is organized by file system. For each file system, the following information is provided:

- Name of the mirrored virtual partition if the file system is a mirrored virtual partition.
- Names of the members of the mirrored virtual partition if the file system is a mirrored virtual partition. Name of the virtual partition, if the file system is a concatenated or striped partition.
- Name of the DriveGuard array if applicable.
- Names of the physical disk partitions on which the file system resides. If the file system is a virtual partition, this column also indicates whether the physical partitions are concatenated or striped. In the list, concatenated partitions are separated by a plus sign (+), and striped partitions are separated by a vertical bar (|).

The second table is organized by drive. For each drive, the following information is provided in graphic form:

- The layout of partitions on the drive, indicated by the partition identifiers.
- The location of unused partitions, indicated by decimal points.

In addition, if you enter the **ax\_diskconf** command with the -**p** option, the second table is expanded to include the following information:

- A list of the file systems on each drive.
- For each file system mounted on a virtual partition, the report includes the identifier of the virtual partition and of its members (if any).

Figure 5-10 shows an edited output of the ax\_diskconf command; Figure 5-11 shows edited output of the second **ax\_diskconf** table with the -**p** option.

```
۷P
                                                    Physical Partitions
File System
                                          RAID
                           Mirror
                                                    ad0a
/usn
                                                    ad0g
/var
                                                    ad0f
                                                    ad8c
/openv
/usr/openwin
                                                    ad0d
/var/crash
                                                    ad0e
/home
                                                    ad0h
/home/docs
                                  vp11
                                                    ad6c+ad7c
                                                    ad30c1ad40c
/home/atria
                                  vp12
/home/sapphire
                                                    ad12c
/home/attic4
                                                    ad29c
/home/dvlp1
                                          ard28
                                                    ad48c+ad54c+ad55c
/home/dvlp2
                                          ard28
                                                    ad48c+ad54c+ad55c
/disks/usr.local
                                                    ad3e
                           vp4
                                  vp5
                                                    ad3f
                                  vp6
    Capacity in 1000 megabytes
        1 2
                                                      7
    la....ggdeeefffffffhhhhl
ad0
ad1
ad2
    Lecceel
    leeeeeeeeeeffffffffffl
ad3
ad5
    ad6
"reid10" 32 lines, 1422 characters
```

Figure 5-10. Example ax\_diskconf display

The following list explains some information in the **ax\_diskconf** display in Figure 5-10:

- ▲ The file system /home/docs uses **vp11** as a concatenated virtual partition made up of **ad6c** and **ad7c**.
- ▲ The file system /home/atria uses a striped virtual partition made up of two members (ad30c and ad40c).
- ▲ The file system /disks.usr.local uses a mirrored virtual partition (**vp4**) whose members are **vp5** and **vp6**. Each of the member partitions consists of a single physical disk partition (**ad3e** for **vp5** and **ad3f** for **vp6**).
- ▲ Drives **ad0** and **ad1** both contain unmounted partitions (indicated by rows of decimal points).

Although all this information is available from other sources, such as **fstab**(5), **vpartab**(5), and **dkinfo**(8), no other utility makes it available in a single report.



```
Capacity in 1000 megabytes
0 1 2 3
   la....ggdeeefffffffhhhhl
ad0
    а
                  /usr
    9
d
                  /usr/openwin
                  /var/chash
                  /van
                  /home
ad1
ad2
    Tecceel
                  /export/mfg
ad3
   leeeeeeeeeefffffffffffl
                /disks/usr.local
       vp4:vp5
    e
f
                  /disks/opt.local
       vp6:vp7
ad5
   /home/attic2
ad6
   Lececeel
    c vp11#1
                  /home/docs
```

Figure 5-11. Example ax\_diskconf second table with -p option



# 6 Recovering From Disk or File System Failures

# **About This Chapter**

This chapter describes server problems that cause a file system to be inaccessible. It also suggests recovery procedures for making the file system available to clients again with minimum interruption of service.

This chapter also describes error recovery on a root disk using the NSinstall command.

The procedures provided in this chapter assume that you are familiar with basic file structure concepts (for example, the functions of inodes, superblocks, and so on). Also, familiarize yourself with virtual partitions by reading Chapter 5 carefully before recovering a disk that is part of a virtual partition.



Note: Several of the procedures involve unmounting or unexporting file systems on the failed drives. You can use the **ax\_diskconf** command to get a report that maps all file systems to the physical partitions or virtual partitions.

Also, some procedures require copying data from the failing drive to a spare drive. Be sure the spare drive is at least the same size as the failing drive, and both drives are formatted with the same formatting scheme.



# When Does a File System Become Unavailable?

A file system may become unavailable for one of the following reasons:

- ▲ Hardware problem, which can be a disk error or a media error:
  - Disk errors: The entire disk becomes invisible to the system. Error messages are generated by the SP, vmunix, or the FP when the system tries to read from or write to this disk. Disk errors can be either permanent or intermittent.
  - Media errors: Only a sector of a disk is damaged. Error messages are generated by the SP or vmunix when the system tries to read from or write to the failing sector. The sector number is reported in the SP or vmunix error message.
- ▲ File system problem, which can be a metadata corruption within the file system.

The symptom of an unavailable file system and the error recovery procedure depend on these factors:

- ▲ Cause of the problem.
- ▲ When the problem occurs (during system reboot or when the system is up and running in multiuser mode).
- ▲ Whether the file system is isolated as a result of the problem (that is, whether the FP takes the file system offline until the problem is corrected).



# Disk Problems at System Reboot

When the system is booting, it detects all the disk drives that have been installed. The power-up messages list disks that are visible to the system. For example:

server name vmunix: ad20: <Seagate 4GB cyl 3605 alt 1 hd 21 sec 110> server\_name vmunix: ad22: <Seagate 4GB cyl 3605 alt 1 hd 21 sec 110> server\_name vmunix: ad23: <Seagate 4GB cyl 3605 alt 1 hd 21 sec 110>

In this example, a disk installed in slot 21 is not visible.

The following describes how the system reacts to a disk that is invisible at boot time:

- If no file system is mounted on the disk, the system can successfully reboot.
- If the disk is a member of a virtual partition, reboot stops and the system displays the following error messages:

```
Virtual partition download failed...help!
WARNING - file systems have NOT been checked.
After fixing the virtual partition download problem,
run "fsck -p" to check the file systems before
going multi-user.
```

Then the system enters single-user mode.

If file systems are mounted on the disk, fsck fails to check the file systems and lists them as file systems that have unexpected inconsistency. The system enters singleuser mode after displaying the following error message:

```
Reboot failed...help!
```



**Note:** When the system cannot see a drive during reboot, the SP does *not* generate an error message. This is the only case in which the SP does not report a disk error.

- If the file systems are mounted on a physical partition of the disk, **fsck** notes the missing device, but the system will reboot successfully.
- If the disk is part of a RAID array, ax\_write\_cache may display messages related to the RAID array changing state, and fsck notes the missing device, but the system will reboot successfully.



# Is the Problem Caused by Poor Disk Installation?

Sometimes the system cannot detect a disk because it is not properly installed. The problem can be easily detected and corrected with this procedure:

- 1. In single-user mode, enter the **ax\_hot\_plug** or **ax\_add\_device** command.
- 2. When the system prompts you to insert or replace the specified disk drive, remove the drive.
- Reinsert the drive in the drive slot, and press the Return key.
   If the SP can successfully attach the drive, the problem was caused by a poorly installed disk.
- 4. Run **fsck** -**p** to check all the file systems, and then press Ctrl-D to go to multiuser mode.

Go to the next section if the SP fails to attach the drive.

# Is the Problem Caused by a Malfunctioning Drive Slot?

If the procedure in the preceding section does not succeed (that is, the SP still cannot detect the drive after you reinstall it), a SCSI interface error message appears. Follow these steps to install the drive in another slot:

1. Remove the drive from its current location. It is not necessary to enter the **ax\_hot\_plug** command because the drive is not recognized by the system software.



**Caution:** Do not remove a disk while it is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive.

- 2. Select a new slot for this drive.
- 3. Use the **ax\_hot\_plug** or **ax\_add\_device** command to add the drive in the new slot. If you need additional information on these commands, refer to "Adding or Replacing a Drive" on page 4-25.

If the drive is detected in the new slot, continue using the disk at the new location. If file systems are mounted on the disk, edit the <code>/etc/fstab</code> file to reflect the new disk location. Also, if the disk is a member of a virtual partition, edit the <code>/etc/vpartab</code> file and then run <code>ax\_loadvpar</code> to load the revised virtual partition table.

4. Run **fsck** -**p** to check all the file systems and then press Ctrl-D to go to multiuser mode.



**Note:** The slot in which the drive was originally installed probably is not working; report the problem to Auspex.

If the drive is still not detected in the new drive slot location, go to the next section.

# Is the Problem Caused by a Disk Without a Disk Label?

If the disk cannot be detected in another slot or if the system still cannot reboot to multiuser mode after the disk has been detected, the problem may be caused by a missing disk label. Although the SP can recognize a disk without a disk label during an **ax\_hot\_plug** or **ax\_add\_device** command, **ax\_loadvpar** and **fsck** fail if they cannot see the label. If an error



message during reboot indicates that there is no disk label on a disk, use the **format** command in single-user mode to relabel it. Refer to "If Sector 0 (Disk Label) Is Unreadable" on page 6-24 for instructions on fixing the problem.

If **format** does not list the disk, you can conclude that the problem arises from a malfunctioning disk. You must remove or replace this bad disk before you can boot the system to multiuser mode.

# Replacing a Malfunctioning Disk

To replace a drive, refer to "Replacing a Drive in Single-User Mode" or "Entering Multiuser Mode and Replacing the Drive". Replacing the disk allows users to have continuous access to the file systems on the drive, although the data on the drive may not be up-to-date. Follow the instructions in "Using the System Without the Failing Drive" on page 6-7 only if you do not have a spare drive and need to wait for a replacement.

#### Replacing a Drive in Single-User Mode

This section describes how to replace a drive while the system is in single-user mode:



**Caution:** The following line appears at the top of the /etc/partab and /etc/fstab files on systems running Version 1.10:

```
#VP256 ENABLED (do not delete)
```

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 1. Comment out the entries in /etc/fstab and /etc/vpartab that reference the bad drive. (To comment out an entry, type a # sign in the first column of the entry.)
- 2. Run **ax\_loadvpar** to load the revised /etc/vpartab.
- 3. Use ax\_hot\_plug to replace the drive. For example, if the drive is installed in slot 4, enter the following:

```
ax hot plug add 4
```

(If the drive is not recognized by the SP, simply remove the drive and insert a new one in the slot at the prompt.)



**Caution:** Do not remove a disk from a slot while the disk is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive.

- 4. Remove the # signs in /etc/fstab and /etc/vpartab that you added in step 1.
- 5. Run **ax\_loadvpar** to load the virtual partition table.
- 6. Use **newfs** to create the file systems that used to exist on the replaced drive.

For example, if **ad4c** is a member of **vp2**, enter the following command:

```
newfs /dev/rvp2
```



- 7. Run **fsck** to check the newly created file system.
- 8. Restore data to the new drive from the most recent backup tapes.
- 9. Run fsck -p to check all the file systems and then press Ctrl-D to go to multiuser mode.

#### **Entering Multiuser Mode and Replacing the Drive**

If you want the server to be in multiuser mode as soon as possible, follow this procedure to replace the drive in multiuser mode:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 1. In single-user mode, comment out the entries in /etc/fstab and /etc/vpartab that reference the drive to be replaced. If you change /etc/vpartab, run ax\_loadvpar to load the revised virtual partition table.
- 2. Run fsck -p to check all the file systems, and then press Ctrl-D to go to multiuser mode.
- 3. After the system boots to multiuser mode, use ax\_hot\_plug to replace the drive. (If the drive is not recognized by the SP, use ax hot plug with the add option. At the prompt, remove the problem drive and insert the new one.)



**Caution:** Do not remove a disk from a slot while the disk is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive.

- 4. If you commented out entries in /etc/vpartab in step 1, remove the # signs for those entries.
- 5. Run ax\_loadvpar if you edited /etc/vpartab.
- 6. Use **dkinfo** to verify that the disk has the correct disk label. If it does not, use **format** to repartition the disk.
- 7. Use **newfs** and **fsck** to create and check the new file systems to be restored.
- Restore data to the new drive from the most recent backup tapes.
- Remove the # signs you added in /etc/fstab in step 1 so file systems can be mounted on the new drive.
- 10. Mount and export the file systems existing on the new drive so network users can access them again.



#### Using the System Without the Failing Drive

The following procedure describes how to make the system available to network users as soon as possible without replacing the failing drive:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

**#VP256 ENABLED** (do not delete)

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

- 1. Comment out the entries in /etc/fstab and /etc/vpartab that reference the bad drive.
- 2. Use ax\_hot\_plug or ax\_remove\_device to remove the drive. (If the drive is not recognized by the SP, simply remove the drive without using ax\_hot\_plug or ax\_remove\_device.)



**Caution:** Do not remove a disk while it is spinning down. Removing a disk drive while it is still spinning can cause permanent damage to the drive.

- 3. If you have modified /etc/vpartab in step 1, run ax\_loadvpar to load the virtual partition table.
- 4. Run **fsck** -**p** to check all the file systems, and then press Ctrl-D to go to multiuser mode. The system boots successfully without the failing drive, but file systems on that drive are no longer available to users.



# Recoverable Disk Problems in Multiuser Mode

A disk may be visible when the system reboots but inaccessible when the system tries to read from or write to it in multiuser mode. When this happens, the SP tries to reach the disk up to 10 times. If one of the retries succeeds, the error is considered recoverable, and the SP generates a message stating that an error was recovered. For example:

Recovered: write drive fault, slot 3, drive 1.

The SP *always* generates an error message after each failed retry. The following example shows an SP error message for a failed attempt to access the drive:

1193:Drive Fault slot 10, drive 1, sense key 4 sense code 44.



**Note:** The slot number, not the drive number, in the SP error message shows the location of the drive. The drive number shows which drive in a particular slot is causing the error. It is always 1 because each drive slot contains only one drive.

It is not necessary to replace a drive that has experienced recoverable errors. However, if recovered errors occur often, or if you believe the read and write retries are slowing down the system, use **ax\_clonefs** or **ax\_sputil copy** to copy the disk to another disk. When you finish copying, replace the disk with the newly created disk.



# **Unrecoverable Disk or File System Problems** in Multiuser Mode

When a system is running in multiuser mode, a file system can become unavailable because of a hardware or software problem. The following subsections discuss how you can resume the use of a file system after an unrecoverable disk or file system error.

- "Error Messages Indicating Disk or File System Problems" discusses the symptoms of an unrecoverable disk or file system error.
- "File System Isolation" describes a NetServer feature that takes an inaccessible or corrupted file system off-line without affecting the operation of the entire server. This section discusses the types of errors that cause file system isolation and outlines the major steps to follow to add the file system back online.
- "Examples of Recovery Procedures After File System Isolation" provides examples in which file systems are isolated under different circumstances. In each example, a detailed procedure shows how to determine the cause of the problem and how to correct it.
- "Recovering from Permanent Disk Errors Without File System Isolation" and "Recovering from Intermittent Disk Errors Without File System Isolation" give instructions on how to recover from a disk error that does not trigger file system isolation.
- "Repairing Disk Sectors After an Unrecovered Media Error" describes how to fix errors on different types of disk sectors. The procedures are applicable whether or not the disk sector causes file system isolation.

# Error Messages Indicating Disk or File System Problems

If the error is not recoverable, error messages appear on the console as well as in /var/adm/messages. The error messages are generated by these sources:

The SP. Note that an SP error message gives the exact location of the problem disk sector. The following is an example of an SP error message:

```
Nov 17 17:47:36 server name SP1: 1196:Disk Media Error, slot 36,
drive 1, sense key 3, sense code 12, sector 3884624
```



**Note:** If the problem is caused by the file system and not by the disk hardware, the SP does not display any error messages.

- The application program.
- The FP or vmunix.

Because the FP resides on an I/O Processor (IOP) board, the message refers to the FP as "IOP*n*/FP*n* CPU B." For example:

```
Nov 17 17:47:36 server_name IOPO/FPO 0 CPU B: vp2: fatal error e
(Permanent drive error) on read, block 9382416
```

The following example shows a vmunix error message:

```
Nov 17 17:47:36 server name vmunix: ad10: fatal error e (Permanent
drive error) on write
```





**Note:** If an unrecovered error occurs on a drive that is mirrored, the SP can complete the disk I/O request on the other member of the mirrored partition. In this case, no error messages are generated from the application program, the FP, or vmunix. The SP, however, prints the following warning message:

```
Warning, a mirrored partition has a dirty or damaged component, vp#
```

The SP also displays other error messages to indicate why the virtual partition is dirty.

If you see an error message from an application program, the FP, or vmunix, but *not* the SP, it is likely that a file system problem, not a disk error, occurred.

If an unrecoverable error happens to an LFS-mounted file system, the FP takes the file system offline so you can fix the problem. The file system isolation feature is discussed in the next section.

## File System Isolation

A file system is considered isolated when the FP takes it offline after detecting hardware or software problems. Isolating an unreachable or corrupted file system gives you time to repair it while allowing clients of other file systems to continue using the server unaffected.

Only an LFS-mounted file system (a file system managed by the FP, not HP) can be isolated. The error conditions that trigger file system isolation are described in the next section.

## Types of Errors that Cause File System Isolation

Two major types of error conditions cause the FP to isolate a file system:

- ▲ Errors detected by LFS
- ▲ Disk problems

#### Errors detected by LFS

Errors detected by LFS are those file system corruption problems that can be fixed by **fsck**. The following lists all file system corruption conditions that cause a file system to be isolated:

```
1018: realloccg: bad optim
1019: ialloc: dup alloc
1038: alloccgblk: cyl groups corrupted
1039: alloccgblk: can't find blk in cyl
1040: ialloccg: map corrupted
1041: ialloccg: block not in map
1043: free: freeing free block
1044: free: freeing free frag
1046: ifree: freeing free inode
1047: alloccg: map corrupted
1048: alloccg: block not in map
1070: isblock
```



```
1071: clrblock
1072: setblock
1073: getmp: bad magic
1074: ufs_statfs
1077: rwip type
1078: ufs readdir: dir reclen == 0!
1079: ufs_readdir: dir reclen > 512!
1094: ufs readdir nfs: dir reclen == 0!
1095: ufs_readdir_nfs: dir reclen > 512!
```

The following examples of error messages in /var/adm/messages are caused by a file system corruption problem:

```
Mar 15 14:04:19 host1 FPO: 1095 ufs_readdir_nfs: dir reclen > 512!
Mar 15 14:04:19 host1 FPO: isolating filesystem mounted on
/disks/home
Mar 15 14:04:43 host1 mountd[270]: rpc.mountd: encountered isolated
filesystem, dropping request to mount /home/host1/steve
Mar 15 14:05:04 host1 ax_isolated[295]: Filesystem /dev/vp77 mounted
on /disks/home is isolated
```

#### Disk problems

When the FP tries to write to a disk that the SP cannot reach, the FP also isolates the file system so you can correct the disk or media error. The following is a list of errors caused by hardware:

PSA\_ERR\_DRIVE\_OFFLINE

This error occurs when the disk is spun down and not on line.

PSA\_ERR\_PERM

This error is any disk error other than PSA\_ERR\_DRIVE\_OFFLINE, which occurs when the SP cannot send commands to the drive.

Unrecovered media errors

The FP does not isolate a file system if a media error is recovered. If the error is not recovered, however, the FP isolates the file system so you can repair the hardware.

The following are sample messages in /var/adm/messages caused by a permanent drive error:

```
Mar 14 07:14:12 host1 FPO: ad52c: fatal error e (Permanent drive
error) on read, block 1983184
Mar 14 07:14:12 host1 FP0: bp->b_un.b_addr = 21e22000, bp->b_attr = 1
Mar 14 07:14:12 host1 FPO: isolating filesystem mounted on /disks/eng
Mar 14 07:15:21 host1 rquotad[368]: rpc.rquotad: Skipping isolated
filesystem /disks/eng
Mar 14 07:15:58 host1 ax isolated[215]: Filesystem /dev/ad52c mounted
on /disks/eng is isolated
```



#### What Does the FP Do When an Attempted I/O Fails?

This section describes what happens when the FP encounters a file system problem.

When you try to delete or create files in an inconsistent file system or a file system on an unreachable disk, the FP cannot complete the required I/O operation. It can, however, take the file system offline so you can do the following:

- Investigate the cause of the problem by examining the SP and FP error messages logged in /var/adm/messages
- Follow the appropriate procedure to repair the file system or disk and then bring the file system back online.

An isolated file system remains mounted but any process that tries to access it hangs.



**Note:** After a file system is isolated, a user's attempt to log in to a server hangs if the .login or any shell startup file contains references to the isolated file system. For example, if /usr/local is LFS-mounted and /usr/local/bin is included in the PATH of .login, you cannot log in as root once /usr/local is isolated. To ensure that you can log in to a server as root after a file system is isolated, do not include commands that try to access an LFS file system in the root's .login file or any shell startup file such as .cshrc and .profile.

The ax isolated daemon is started by /etc/rc.local to monitor file system isolation.

The syntax of the **ax\_isolated** command is as follows:

```
ax_isolated [ -m user(s) ] [ -n ]
```

By default, ax\_isolated sends email to root if a mounted file system is isolated. If you want messages sent to additional users, enter user names after the -m flag. (Use a comma to separate multiple user names.) For example:

```
ax isolated -m bsmith, lchung, smueller
```

When a file system is isolated, the ax isolated daemon automatically calls fsck to check the file system. If fsck is successful, ax\_isolated then releases the file system. If you wish to disable automatic calling of fsck as well as ax fsutil, invoke ax isolated with the -n flag.

An additional option for **ax\_isolated** is **-f**, which is automatically placed in /var/spool/cron/crontabs/root. Do not invoke this flag from the command line.



Note: Upon system startup, ax isolated is invoked without the -m, -n, or -f flag. To start ax isolated with the -m or -n flag, you must first kill the currently running ax\_isolated daemon. Refer to the ax\_isolated man page for more information.

If you suspect file system isolation, use the following command to determine which file system is isolated:

```
ax fsutil
```



Running **ax\_fsutil** without any arguments prints a list of isolated file systems. **ax\_isolated**(8) also places file system isolation messages in /var/adm/messages. The ax\_isolated daemon generates the following message showing which file system was isolated:

```
Mar 14 07:15:58 host1 ax_isolated[215]: Filesystem /dev/ad52c mounted
on /disks/eng is isolated
```

The procedure for recovering from an isolated file system varies depending on the cause of the problem. (See the examples in "Examples of Recovery Procedures After File System Isolation" on page 6-15.) You must first determine whether a disk problem or an LFS problem has occurred. If it is a disk problem, decide whether to fix or replace the disk. Try to salvage the disk because replacing a disk and restoring data from a backup tape takes more time.

The following example shows a file system that is isolated because of an LFS error. In this case, the file system can be repaired by fsck before it becomes available to network users again.

In the /mnt file system, the metadata pertaining to the inode is corrupted. If you try to use **rm** to delete a file within /mnt, the command hangs and the following messages appear on the console:

```
May 19 15:34:42 Host1 FP0: dev = 0x0, ino = 8, fs = /mnt
May 19 15:34:42 Host1 FP0: isolating filesystem mounted on /mnt
May 19 15:34:42 Host1 FPO: 1046: ifree: freeing free inode
```

In this example, because of a file system inconsistency, the FP tries to free an inode that was already marked "free." To remedy the situation, the FP takes /mnt offline, which causes all activity on the file system to stop and all LFS requests to the file system to return an error message. This error message is transparent to user programs but is handled by the HP or NPs. Other file systems on the server, however, continue to function as usual.

If network users try to access an isolated file system, they receive no responses from the server. To them, the server seems to have hung, but in fact it continues to deliver file service for other file systems. NFS I/O operations continue on other file systems that are not isolated.

#### Repairing an Isolated File System with fsck

If you have determined that an LFS error, not a faulty disk, caused the isolation, you can repair the file system by running **fsck** interactively while the file system remains mounted.

While the NetServer is running, fsck keeps track of dirty cylinder groups in each file system. If a file system contains dirty cylinder groups, fsck checks only those cylinder groups at reboot, and the file system name is entered in /etc/cgfscklog. File systems listed in /etc/cgfscklog are checked later on by ax\_isolated -f, which is normally invoked from a cron script.

If the Write Accelerator contains unwritten data for the isolated file system (that is, write cache state is DIRTY), the fsck command may fail, and an error message appears. The following is an example of the error message:

```
ad4c: fatal error 10 (Accessing Dirty Write Cache) on read, drive 0,
block 64, absolute block 64
```



To flush the Write Accelerator, use the **ax\_write\_cache** command. For example, if the Write Accelerator is installed on SP0, enter the following:

```
ax_write_cache -s 0 flush
```

After the state of the Write Accelerator changes to OFF, you can retry the **fsck** command. If the flushing is not successful, there may be a malfunctioning disk in the system that prevents data from being written. Check the SP error messages in <code>/var/adm/messages</code> to determine which sector caused the problem. Use **ax\_sputil reassign** to reassign the bad sector, and try flushing again. If it still fails, purge the data from the cache.

For information on the arguments to **fsck**, refer to the **fsck** man page. If you want to know more about how **fsck** repairs corrupted file systems, refer to Sun's *System and Network Administration*.

If the file system is successfully repaired by **fsck**, go to "Releasing a File System After Repairing It by fsck" for information on making the file system available again. If the file system cannot be repaired because of a bad disk, go to "Restoring a File System that Cannot Be Repaired" on page 6-15 for information on killing processes sleeping on the file system.

#### Releasing a File System After Repairing It by fsck

After fixing the inconsistency of an isolated file system, release the file system to make it available again with the **ax\_fsutil** command. Use the partition name or the mount point as the argument in the command. For example, to release the file system with the mount point /mnt, enter the **ax\_fsutil** command with the **release** option as follows:

```
ax fsutil release /mnt
```

If /dev/vp3 is mounted on /mnt, you can also enter the command in this form:

```
ax fsutil release /dev/vp3
```

**ax\_fsutil** checks /etc/mtab to verify the file system you try to release is indeed mounted. If /etc/mtab has no entry for the file system, **ax\_fsutil** prints out this message:

```
ax fsutil release mountpoint/filesystem
```

If **ax\_fsutil** is successful, activity on the file system resumes, and the following message is generated:

```
ax_fsutil: Successfully released filesystem mounted on /mnt
```



**Note:** Invoke **ax\_fsutil** without an argument to list any file systems that have been isolated. You are not required to be root to run **ax\_fsutil** in this form.

**ax\_fsutil** works only on a clean file system. Therefore, be sure to run **fsck** on an isolated file system *before* using **ax\_fsutil** to release it. If you try to release an unclean file system, messages similar to the following appear on the console:

```
Dec 20 15:34:42 cabot IOPO/FPO (CPU B): fc_unfence: File system
/test/ad2f isn't clean and can't be released
Dec 20 15:34:42 cabot IOPO/FPO (CPU B): Can't release, got error ==
16
```



On the terminal where you enter the command, the following error message appears:

```
ax_fsutil: test/ad2f isn't clean. Cannot release.
```

If a disk problem caused an isolated file system, using **fsck** to repair the file system does not solve the problem. Note that the **fsck** command may not alert you to the disk problem. For example, if the problem is intermittent, **fsck** may mark the file system clean after the file system check, which allows you to release the file system successfully. However, the file system may be isolated again immediately with the same error. In this scenario, fix the disk problem before putting the file system online again.

#### Restoring a File System that Cannot Be Repaired

If a permanent disk error causes a file system to be isolated, follow these steps to make the file system available again:

1. Enter **ax\_kill** to terminate all the processes sleeping on the isolated file system.



**Note:** Sometimes **ax\_kill** fails to terminate all processes. In this case, a message appears to ask you to kill the processes manually.

- 2. Unexport the file system, and unmount the isolated file system.
- 3. Replace the failing disk. Refer to "Adding or Replacing a Drive" on page 4-25 for information on how to replace a drive. Run **newfs** and **fsck** to create and check a new file system on the disk, and then restore the file system from the backup tape.
- 4. Run **fsck** against the restored file system.
- 5. Mount the file system and re-export it to make it available again.

# **Examples of Recovery Procedures After File System Isolation**

This section provides three scenarios in which file systems are isolated for different reasons. For each scenario, the symptoms are described first, followed by the solution. The purpose of each of these recovery procedures is to bring the file system back online as fast as possible while avoiding system reboots. The media errors described in the examples are *unrecoverable* errors because only unrecovered errors cause file system isolation.

## File System Isolation Caused by Media Errors That Cannot Be Fixed

The following sample procedure shows how to recover from a file system that was isolated because of multiple media errors:

1. Enter the following command to determine which file system has been isolated:

```
grep isolate /var/adm/messages
Nov 19 17:05:02 host1 syslog: ax_isolated: Filesystem /dev/vp80
mounted on /export/root is isolated
```

2. Determine which SP handles the file system with the error. In this example, the error occurred in **vp80**, which is supported by the first SP (that is, SP0). Examine the SP error messages in /var/adm/messages by entering the following command:

```
grep SP0 /var/adm/messages
Nov 19 17:04:50 host1 SP0: 1196:Disk Media Error, slot 36, drive 1,
sense key 3, sense code 12, sector 3884624.
```

```
Nov 19 17:04:51 host1 SP0: 1196:Disk Media Error, slot 36, drive 1, sense key 3, sense code 12, sector 3884625.

Nov 19 17:05:51 host1 SP0: 1196:Disk Media Error, slot 36, drive 1, sense key 3, sense code 12, sector 3884632.

Nov 19 17:05:52 host1 SP0: 1196:Disk Media Error, slot 36, drive 1, sense key 3, sense code 12, sector 3884633.
```

These messages show that multiple media errors occurred in sectors that are close to one another, which may be the result of a faulty read/write head. To confirm, use the **ax\_sputil verify** command to see if the damaged sector can be verified. For example:

```
ax_sputil verify 36 3884625
```

If the damaged sector can be verified, see "File System Isolation After Media Errors That Can Be Fixed" on page 6-17.

If the **ax\_sputil verify** command hangs or returns error messages, the sector cannot be verified. In this case, you cannot map out the problem sector. Instead, replace the disk and restore the file system using the following steps.

3. Unexport the file system so clients cannot start processes that try to access the file system. For example:

```
exportfs -u /export/root
```

Diskless clients trying to access the file system receive a stale file handle error message after the file system has been unexported. As a result, the clients crash. This is acceptable in this example, however, because clients cannot access the bad disk anyway. In this example, unexporting /export/root affects only diskless clients.

- 4. Enter **ax\_kill** to kill all processes that are sleeping on the isolated file system.
- 5. If the SP controlling the isolated file system uses write acceleration, flush all unwritten data in the write cache to all disks and then purge the data that can not be written. For example:

```
ax_write_cache -s 0 flush
ax write cache -s 0 purge
```

The write accelerator is automatically turned off after these steps. Enable it again by entering the following command so other file systems on SP0 can continue to use write acceleration:

```
ax_write_cache -s 0 enable
```

- 6. Unmount the file system so you can replace the disk.
- 7. Use **ax\_hot\_plug** to replace the drive. For example:

```
ax_hot_plug add 36
```

Refer to "Installing or Replacing a Drive After Booting" on page 4-26 for more information on how to replace a disk.

- 8. After replacing the disk, run **newfs** and **fsck** to create and check the new file system.
- 9. Restore the file system from the most recent backup. Run **fsck** to check the file system after the restore.



- 10. Mount and export the file system so it is available to clients again.
- 11. Reboot diskless clients that crashed when the disk was replaced. (This step is required for this example. For other file systems, the clients would not have crashed and would not require a reboot.)

The file system recovery procedure is now complete.

#### File System Isolation After Media Errors That Can Be Fixed

The example in this section shows how to recover from an isolated file system by repairing a bad sector that caused file system isolation:

1. Enter the following command to determine which file system was isolated:

#### grep isolate /var/adm/messages

```
Nov 19 17:05:02 host2 syslog: ax_isolated: Filesystem /dev/vp277
mounted on /home is isolated
```

2. Determine which disk underlies **vp277** by entering the following command:

```
grep vp277 /etc/vpartab
```

```
/dev/vp277
               concat
                              ad47c
```

The screen output indicates that if disk errors occurred, they occurred on ad47.

Examine the SP error messages in /var/adm/messages by entering the following command:

#### grep SP1 /var/adm/messages

```
Nov 20 16:06:37 host1 SP1: 1233:Recovered, SCSI bad bus phase at
data xfer, slot 47, drive 1.
Nov 20 16:06:50 host1 SP1: 1196:Disk Media Error, slot 47, drive 1,
sense key 3, sense code 12, sector 196224.
```

Ignore the first message because it describes a recovered error. The second message indicates that an unrecoverable error happened in sector 196224 of ad47.

4. Use the **ax\_sputil verify** command to see if the disk can be verified:

#### ax sputil verify 47 196224

```
ax_sputil: Read write sector error, status = 02030000.
sense data = F0000500 3BB1EC14 00000000 21000000 00000000 00000
```

Because the ax\_sputil reassign command can return the verification result, try mapping out the bad block.

5. Try reassigning the sector to a new location as follows:

#### ax\_sputil reassign 47 196224

```
ax_sputil: Read write sector error, status = 02030000.
sense data = F0000500 3BB1EC14 00000000 21000000 00000000 00000
ax sputil: restoring data to sector 196224 is needed.
ax_sputil: Confirm reassign sector (y/n)? y
ax_sputil: Read write sector error, status = 0e03ffe0.
sense data = 70000500 00000014 00000000 21000000 00000000 00000
000
ax_sputil: sector 196224 reassigned and data restored.
```

6. If the SP controlling the isolated file system uses write acceleration, flush all data from the write accelerator:

```
ax_write_cache -s 1 flush
ax_write_cache -s 1 purge
```

The write accelerator is automatically turned off after these commands.

7. Use **fsck** to check the isolated file system, which fixes any damage to the file system caused by the bad disk sector. For example:

```
fsck /dev/rvp277
```

```
Start fsck of /dev/rvp277: CONTINUE? [yn] y
** /dev/rvp277

** Last Mounted on /home

** Phase 1 - Check Blocks and Sizes

** Phase 2 - Check Pathnames

DIRECTORY CORRUPTED

SALVAGE? [yn] y

** Phase 3 - Check Connectivity

** Phase 4 - Check Reference Counts

** Phase 5 - Check Cyl groups

FREE BLK COUNT(S) WRONG IN SUPERBLK

SALVAGE? [yn] y

158427 files, 4389754 used, 1154922 free (2114 frags, 144101 blocks, 0.0% fragmentation)
```

8. Add the file system back online. For example:

```
ax fsutil release /home
```

Successfully released filesystem mounted on /home

9. Reenable the write accelerator if one is used on the SP. For example:

```
ax_write_cache -s 1 enable
```

The file system recovery procedure is now complete.



**Note:** Whether you unexport the file system while fixing the disk sector depends on how many clients are affected by the file system and how long the repair takes. In this example, */home* is used by all clients, which all hang if you do not unexport */home*. However, because reassigning the bad block takes only a short time, clients can resume access to */home* without too much delay.

## File System Isolation Caused by Corruption

This section provides an example showing how to recover from a file system that was isolated because it was corrupted. The disk on which the file system resides is not at fault.

1. Enter the following command to determine which file system has been isolated:

```
grep isolated /var/adm/messages
Nov 21 18:10:33 host3 syslog: ax_isolated: Filesystem /dev/ad6
mounted on /wrld/apps/frame_3.0 is isolated
```

2. Search /var/adm/messages to determine if file system isolation was caused by the SP's failure to reach the disk. File system corruption is the cause if /var/adm/messages contains no SP error messages or FP error messages indicating a permanent disk error. In this example, the FP messages logged are as follows:



```
Nov 21 18:10:05 host3 FPO: 1012: /wrld/apps/frame_3.0: bad block
Nov 21 18:10:18 host3 FP0: bad block -872415232, ino 53765
Nov 21 18:10:23 host3 FPO: bad block 805306819,
```

3. If the SP controlling the file system uses write acceleration, flush all unwritten data from the write cache:

```
ax_write_cache -s 0 flush
ax write cache -s 0 purge
```

The write accelerator is automatically turned off after these commands.

4. Unexport the file system so clients trying to mount it do not hang. For example:

```
exportfs -u /wrld/apps/frame_3.0
```

Because the isolated file system is for a specific application, unexporting it only aborts the application on clients that have mounted it. Clients trying to mount it are denied the use of this application, but they do not hang.

- 5. Use **fsck** to check the file system.
- 6. Add the file system back online with the following command:

```
ax fsutil release /wrld/apps/frame 3.0
Successfully released filesystem mounted on /wrld/apps/frame_3.0
```

7. Re-enable the write accelerator if one is used:

```
ax_write_cache -s 0 enable
```

- 8. If there is lost data, restore it from the most recent backup.
- 9. Export the file system again so clients can mount it.

The file system recovery procedure is now complete.

# Recovering from Permanent Disk Errors Without File System Isolation

When the server is running in multiuser mode, it is possible for a disk to become invisible without triggering file system isolation. If the file system on the disk is managed by the HP or if the disk error is not one of those causing file system isolation, replace the disk as soon as possible to prevent a system crash.

#### Replacing a Disk Containing Unmirrored Partitions

If the disk experiencing problems contains file systems that are not mirrored, follow these steps to replace the disk:

1. Unexport and unmount the file systems on the inaccessible drive. For example, if **ad3** is the inaccessible disk and the file system mounted on ad3c is /home/src, enter the following commands:

```
exportfs -u /home/src
umount /home/src
```



- 2. If the disk is a member of a concatenated or striped virtual partition, follow these steps:
  - a. Edit out the entries for this drive in /etc/vpartab.
  - b. Run **ax\_loadvpar** to load the new virtual partition table, which no longer includes the bad disk.
- 3. Use **ax\_hot\_plug** to replace the failing drive with a spare drive.
- 4. Verify the new drive has the correct disk label. The new disk should have the same partitions as the replaced disk. Obtain the partitioning information about the replaced disk from your records or worksheets. Use **ax\_lslabel** to check the partitioning scheme for the new disk, and use **format** to repartition it if necessary. All partitions referenced in /etc/vpartab must exist in the disk label, and their size must be greater than 0 MB.
- 5. If the disk is part of a concatenated or striped virtual partition, follow these steps:
  - a. Open /etc/vpartab, and add entries for this drive.
  - b. Run **ax\_loadvpar** to load the new virtual partition table.
- 6. Run **newfs** to create the file systems on the new disk.
- 7. Restore data to the file systems on the new drive from the most recent backup tapes.
- 8. Mount and export the file systems you unmounted and unexported in step 1.

#### **Replacing a Disk Containing Mirrored Partitions**

If the disk experiencing problems contains mirrored file systems, follow these steps to replace it:



**Caution:** The following line appears at the top of the /etc/vpartab and /etc/fstab files on systems running Version 1.10:

```
#VP256 ENABLED (do not delete)
```

Do not delete this line. Also, it must always be the first line of the file. If using RCS control, add any text below the first line. Deleting or moving the first line causes problems with virtual partition numbering in subsequent upgrades.

1. If a mirrored partition has a damaged member, the following message is sent to /var/adm/messages:

Warning: a mirrored partition has a dirty or damaged component, vp#



**Note:** You can also use **ax\_vpstat** to determine which mirrored partition has a bad component.

2. Run **ax\_mdetach** to detach the bad disk from the mirrored partition. This step is required for each mirrored partition that has a member on the damaged disk.

For example, if the mirrored partition is **vp3** and the member partition on the bad disk is **vp4**, enter the following:

```
ax_mdetach vp3 vp4
```



This change is automatically recorded in /etc/vpartab. ax\_mdetach appends a comment to the end of the file for the edited entry. For example:

```
# ax mdetach: vp4 removed from vp3 on Fri May 5 22:18:50 1995
```

The mirrored partition, now with one member, never stops performing read and write operations even when a member of the virtual partition is being detached.

3. Edit /etc/vpartab to comment out any remaining references to the virtual partition or the failed disk drive. The following example shows the virtual partition vp4 commented out:

```
% vi /etc/vpartab
# Virtual partition 3- Mirrored
/dev/vp3 mirrored
                                     vp2
#/dev/vp4 concat
/dev/vp2 concat
                                     ad6e
                                     ad2e
```

- 4. Run **ax\_loadvpar** to install the new partition table.
- 5. Use **ax\_hot\_plug** to replace the problem drive with a spare drive.
- 6. Verify that the replacement drive has the correct label. If necessary, repartition the drive using format.
- 7. Edit /etc/vpartab to add the member partition from the replacement drive and to comment in any additional references to the added member partition or replacement drive.

The following example shows the member partition, vp4, from the replacement drive added to the mirrored partition, vp3. It also shows that the reference to vp4 is no longer commented out.



**Caution:** To avoid erasing data, the virtual partition you are adding must be listed last. This allows the data from the existing member partition (vp2 in the example) to be copied to the new member partition (vp4).

```
% vi /etc/vpartab
# Virtual partition 3- Mirrored
/dev/vp3
                                              vp2, vp4
                     mirrored
/dev/vp4
                      concat
                                              ad6e
/dev/vp2
                                              ad2e
                      concat
```

8. Run **ax\_loadvpar** to install the new partition table.

After ax\_loadvpar runs, ax\_mrestore automatically runs to update the mirrors.

9. Verify the completion of the mirror restoration process with **ax\_vpstat**.

# **Recovering from Intermittent Disk Errors Without** File System Isolation

It is common for a drive to return to normal operation after being temporarily inaccessible. Intermittent disk errors are harmless unless they happen frequently (for example, several times a week). Replace a disk with a high rate of intermittent errors by using one of the procedures described in the next two sections.





**Note:** Both procedures involve copying data from the drive with errors to a spare drive. However, if the drive with errors becomes invisible to the system before data copy is complete, follow the procedure in "Recovering from Permanent Disk Errors Without File System Isolation" on page 6-19.

#### Replacing a Drive that Is a One-Member Mirrored Partition

If the drive that experiences intermittent errors is a one-member mirrored partition, use commands in the Virtual Partition Manager to copy data from the bad disk to the spare disk.

The following example assumes the bad disk contains a single virtual partition, **vp3**, and **vp3** is the only member of a mirrored partition, **vp2**. The entry in /etc/vpartab for **vp3** is as follows:

/dev/vp2 mirrored vp3

- 1. Obtain and install a spare drive (the new drive) that is the same size as the one to be replaced (the old drive).
- 2. Choose an available drive slot for the new drive.
- 3. Use **ax\_hot\_plug** to install the new drive.
- 4. Insert the new drive, and then type a carriage return.
- 5. Edit /etc/vpartab to create a virtual partition on the new disk. For example, if the spare drive is in slot 8, you can add the following /etc/vpartab entry to create **vp4**:

/dev/vp4 concat ad8

- 6. Run ax loadypar to load the virtual partition table.
- 7. Use **ax\_clonefs** to copy data from the bad drive to the new drive. For example:

ax clonefs /dev/vp3 /dev/vp4

8. Edit /etc/vpartab to change the definition of **vp2** and to comment out the reference to **vp3**. For example, if the old drive is in slot 6, the new entry for **vp2** and **vp3** should read as follows:

/dev/vp2 mirrored vp4 #/dev/vp3 concat ad6c

- 9. Run **ax\_loadvpar** to load the new /etc/vpartab.
- 10. Use **ax\_hot\_plug** or **ax\_remove\_device** to remove the old drive.

#### Replacing a Drive that Has Multiple Partitions

If the drive generating errors consists of multiple partitions instead of a one-member mirrored partition, follow this procedure to replace it:

1. Obtain a spare drive (the new drive) that is at least the same size as the one to be replaced (the old drive).



2. Choose an available drive slot for the new drive.



**Note:** If you use **ax\_sputil** in step 6, the new drive must be supported by the same SP as the old drive. If you use **ax\_clonefs** in step 6, the drive can be in any slot.

- 3. Use **ax\_hot\_plug** to install the new drive.
- 4. Insert the new drive, and then type a carriage return.
- 5. Unexport and unmount all the file systems on the old drive so they cannot be modified while you copy data from the old drive to the new drive.
- 6. Use **ax\_sputil** to copy data from the old drive to the new drive. For example, if you are copying data from slot 4 to slot 9, enter the following:

```
ax_sputil copy 4 9
```

The copy operation copies the disk label as well as the contents of the disk. For more information, refer to the **ax\_sputil** man page.

Alternatively, you can use **ax\_clonefs** to copy each partition from the old drive to the new drive.

7. Use **ax\_hot\_plug** to remove the new drive from slot 9 and replace the old drive with the new drive in slot 4:

```
ax_hot_plug remove 9 add 4
```

For more information, refer to the **ax\_hot\_plug** man page.

8. Mount and export the file systems on the new drive.

# Repairing Disk Sectors After an Unrecovered Media Error

A media error occurs when a disk sector rather than the entire disk is damaged. The following two examples of SP error messages indicate that media errors occurred. In an actual message, *nnn* is the number of the bad sector.

```
Nov 17 17:47:36 host_name SP1: 1211: Recovered, correctable ECC error, slot 21, drive 1, sector nnn.

Nov 17 17:47:36 host_name SP1: 1196: Disk Media Error, slot 22, drive 1, sense key 3 sense code 11, sector nnn.
```

How to recover from a media error depends on which disk sector is bad. Because only a particular sector is damaged, the recovery procedure involves remapping the bad sector and restoring the file affected by the bad sector instead of replacing the entire drive. The following sections describe the procedures for recovering from a media error in various situations. After repairing the bad sector, add the file system back online if a media error caused file system isolation.

#### If Sector 0 (Disk Label) Is Unreadable

Sector 0, which is the disk label, is the most important disk sector because it contains drive partitioning information. If the label is missing on a disk, the server cannot check the file system on this disk during reboot, nor can it reboot to multiuser mode. To make the disk label readable again, follow these steps:

- 1. If the SP does not generate an error message, the label probably has been erased accidentally. Follow these steps to relabel the disk:
  - a. Invoke the **format** command. If the disk is not listed by **format**, go to step b. If the disk is listed, use the **backup** command under **format** to recover the spare disk label and copy it to the primary label. If this problem happens more than once, reassign sector 0 as described in step 2.
  - b. If the disk is not listed by **format**, that means the disk cannot be accessed by the system at all. Go to "Replacing a Drive in Single-User Mode" on page 6-5 for more information about replacing an inaccessible disk.
- 2. If an SP error message indicates that a media error occurs in sector 0, use the **ax\_sputil** command to reassign the label to another location. For example, if sector 0 of disk 2 is bad, enter the following command:

```
ax sputil reassign 2 0
```

#### If Superblock Is Unreadable

If a file system superblock cannot be read during system reboot, **fsck** stops and the system cannot boot to multiuser mode. If the superblock problem happens during normal server operation, an error message reports that the superblock is unreadable. In either case, the SP generates an error message showing the number of the sector that caused the media error.

Follow this procedure to reassign the superblock:

1. Use the **ax\_sputil** command to reassign the superblock. The following is an example: Suppose the SP generates the following media error message:

```
Nov 17 17:47:36 host_name SP1: 1196: Disk Media Error, slot 22, drive 1, sense key 3 sense code 11, sector 16.
```

Enter the **ax\_sputil** as follows to reassign the superblock:

```
ax_sputil reassign 22 16
```

This command reassigns sector 16 of disk 22 to a new location. However, because the superblock is larger than one sector, you may need to reassign more than one sector.

2. Use the **fsck** command to restore information from the alternate superblock to the new superblock:

```
fsck -b 32 rad22a
```

To see a complete list of backup superblocks before running **fsck**, use the **newfs -N** command. (Block 32 is always the first alternate superblock.) For example, to list the superblocks for **ad0a**, enter the following command:

```
newfs -N /dev/rad0a
```

3. If you are already in multiuser mode, mount and export the file system.

If you are in single-user mode, use **fsck** -**p** to check the file systems. Then press Ctrl-D to enter multiuser mode.



#### If Other Disk Blocks Are Unreadable

If an unrecovered media error occurs on a sector that is neither a disk label nor a file system superblock, you see error messages from these sources besides the SP error message:

The FP (if the media error occurs in an LFS file system).

The following is an example of an FP message for an error on a virtual partition:

```
Nov 17 17:47:36 server_name FPO: vp2: fatal error e (Permanent
drive error) on read, block 9382416
```

The following is an example of an FP message for an error on a physical partition:

```
Nov 17 17:47:36 server name FPO: ad2c: fatal error e (Permanent
drive error) on read, block 9382416
```



**Note:** The block number printed by the FP is relative to the beginning of the partition, whether the error occurs in a virtual or physical partition. That is, the block number printed by the FP always matches the block number in the corresponding SP error message.

The UNIX device driver (if the media error occurs when the server tries to access a UFS file system).

In addition, you may see an error message from the application program that tried to access the bad sector.



**Note:** If the SP error message indicates that the error has been recovered, you can ignore the message unless it happens frequently. Alternatively, you can use the ax sputil command to reassign the sector to a new location. For example, if the SP reports a media error on sector 2718 of disk 5, enter the following:

```
ax sputil reassign 5 2718
```

Always note the result of the reassign operation as a success or a failure.

If the error occurred in a sector other than the disk label or a file system superblock, the recovery procedure is as follows:

- 1. Use **ax\_sputil reassign** to reassign the bad block to a new location. If the reassign operation is successful, the procedure is complete. If not, go to the next step.
- 2. Unexport and unmount the file system affected by the bad block.
- 3. Run **fsck** on the file system.
- 4. Mount and export the file system again.
- 5. Restore from the most recent backup tapes any files that were erased by **fsck**.

# **Recovering a Damaged Root Disk**

This section outlines procedures for recovering from a failure on the root disk. The procedures range in complexity from booting from a backup root disk (the easiest and quickest means of recovery) to reinstalling the operating system from CD-ROM. The choice of procedure depends on whether you have a backup root disk or a recent level 0 backup of the root partitions on tape, and on the nature of the root disk damage.

Auspex provides the **NSinstall** command for installing the operating system from the distribution CD-ROM to a formatted disk in any slot. However, installing the operating system could be a time-consuming process and can result in losing changes to many site-specific configuration files. Before using **NSinstall** to install the operating system, refer to the procedures outlined as follows. In most cases, you can use one of the less drastic methods and avoid running **NSinstall**.

# **Booting From a Backup Root Disk**

Use this procedure if you have a backup root disk from which you can boot the server:

- 1. Power off the server.
- 2. Remove the backup root disk from its slot if it is installed in the server.
- 3. Remove the damaged root disk from slot 0, and replace it with the backup root disk.
- 4. Select an unused slot for the damaged root disk.
- 5. Power up and boot the NetServer in single-user mode (refer to "Booting the Server at the Monitor Prompt" on page 3-4).
- 6. Determine whether the damaged root disk can be repaired using **fsck** or **format**.
- 7. If the entire disk is damaged, go to step 8. If only the root partition was damaged, not the entire disk, follow these steps:
  - a. Re-create the root partition using one of the following methods:
  - ▲ Run **newfs** on the root partition, and restore the root partition from the most recent level 0 backup.



**Note:** After you restore from tape, the restored files may be on different inodes. If you attempt to run an incremental backup on the restored files, the backup may not select the files you expect. Thus, the first backup you run on the restored root disk should be a full level 0 backup.

- ▲ Use the **dd**(1) command to copy the root partition from the backup root disk onto the repaired root disk.
- b. Run **installboot** on the restored root partition to make the disk bootable. For example, if the disk you are repairing is in slot 1, enter the following commands to install the bootblocks in **ad1a**:

```
# mkdir /a
# mount /dev/adla /a
# cd /usr/mdec
# ./installboot /a/boot bootad /dev/radla
```



c. Use halt to halt the processor. When the system is in monitor mode, replace the backup root drive with the restored root drive, and boot the system to multiuser mode.

The root drive recovery procedure for a damaged root partition is now complete.

- 8. If the entire root disk was damaged, and you are replacing it with a new disk, use ax\_sputil to copy the entire contents of the backup root disk onto the new root disk.
- Use **halt** to halt the processor. When the system is in monitor mode, replace the backup root drive with the restored root drive, and boot the system to multiuser mode. The root-drive recovery procedure is now complete.

# **Booting From the CD-ROM in Single-User Mode**

Use this procedure if you have no backup root disk, or if your backup disk fails to boot:

- 1. Use the **halt** command to bring the system down to monitor mode.
- 2. Install the distribution CD-ROM in slot 1.
- 3. Power up the NetServer. Then boot the server in single-user mode by entering the following command:

```
HP> b ad(0,1,1)
```

4. After the system boots, the following messages appear:

```
Remounting and replenishing /etc
Remounting and replenishing /var
Remounting and replenishing /tmp
AUSPEX CD-ROM MAINTENANCE MENU
OPTION:
    1) Format/label the root drive
    2) NSinstall
    3) Maintenance shell
Select an option? [1/2/3] (3)
```

Select 3 to enter the maintenance (single-user mode) shell. The single-user mode prompt (#) appears.

5. Check the label on the root drive using the **dkinfo** command:

```
# dkinfo ad0
```

- 6. If the label is incorrect, apply the correct label using **format**.
- 7. Determine whether the damaged disk can be repaired using **fsck**:

```
# fsck /dev/rad0a
# fsck /dev/rad0g
# fsck /dev/rad0f
```

- 8. If the root partition is severely damaged, go to step 9. If it can be mounted, follow these steps:
  - a. Mount it and examine the root partition to see what damage would cause the system not to boot. For example:

```
# mount /dev/ad0a /a
# cd /a
# ls -l vmunix boot sbin
...
# ls -lt etc | more
```

The output of these commands helps you determine whether the files crucial for booting are present and whether they have been changed.

b. If you notice the files are modified or missing, restore them from the most recent backup.

If a backup is not available, go to "Using NSinstall to Install Software From CD-ROM" on page 6-29 to restore the root partition from the CD-ROM. After restoring the partition, modify the files in /etc to include site-specific information. Refer to Table 2-1 on page 2-5 for a list of site-specific files.

c. If you cannot determine what is wrong with the files, restore the root partition from the most recent level-0 backup. For example, enter the following if the tape in slot 9 is the backup tape:

```
# mount /dev/ad0a /a
# cd /a
# restore -if /dev/nrast9
```

After restoring the files, go to step 10.

- 9. If the root partition is severely damaged, do either of the following:
  - A Run **newfs** on the root partition of the damaged disk, and restore the root partition from the most recent level 0 backup. For example:

```
# mount /dev/ad0a /a
# cd /a
# restore -if /dev/nrast9
```

- ▲ If a backup is not available, go to "Using NSinstall to Install Software From CD-ROM" on page 6-29 to restore the root partition from the CD-ROM. After restoring the partition, modify the files in /etc to include site-specific information. Refer to Table 2-1 on page 2-5 for a list of site-specific files.
- 10. If you restore the root partition from a backup, you must run **installboot** on the root partition to make the disk bootable. For example, use the following commands to install the bootblocks in **ad0a**:

```
# mount /dev/ad0a /a
# cd /usr/mdec
# ./installboot /a/boot bootad /dev/rad0a
```

11. To restore other damaged partitions that contain site-specific information that can be easily edited or contain no site-specific information, follow the instructions in "Using NSinstall to Install Software From CD-ROM" on page 6-29.



# Using NSinstall to Install Software From CD-ROM

Normally, installing the operating system is not required because each server is shipped with the NetServer software installed. However, there are situations where you want to copy the operating system from the distribution CD-ROM to a disk:

- When the root disk or a file system on the root disk is damaged and must be re-created.
- When you want to create a spare root drive.
- When you want to update to a new operating system release. See the *Software Release Note* included with a your new release package for the use of **NSupdate**(8).

NSinstall installs file systems from the distribution CD-ROM to a specified disk. Its syntax is as follows:

```
NSinstall [ -cdrom slot# ]
```

The *slot#* is the slot where the CD is inserted. Slot numbering starts from 0. Without an argument, NSinstall uses the CD in slot 1. NSinstall can run in either single-user mode or multiuser mode.



**Note:** NSinstall assumes that the drive to which the file systems are installed is formatted. For information on how to format a drive, refer to "Using the Format Command" on page 4-36. Formatting a disk erases all data that existed on the disk before the format operation.

The following subsections describe the steps for installing file systems from the CD-ROM.

# Starting NSinstall

If you are installing file systems to the current root drive, boot the system from the CD-ROM as described in "Booting the server from the CD" on page 3-7.

If you are installing file systems to a spare disk, you can run NSinstall in multiuser mode. For example, you can invoke NSinstall to install file systems from a CD-ROM in slot 5 by entering the following command:

NSinstall -cdrom 5



Note: If you want to run NSinstall in multiuser mode from a root disk running a system software release earlier than Version 1.7, first mount the disk before running the installation program.

After entering the **NSinstall** command, the following message appears:

AUSPEX CD-ROM MAINTENANCE MENU OPTION:

- 1) Format/label the root drive
- 2) NSinstall
- 3) Maintenance shell

Select an option? [1/2/3]

The target drive for the **NSinstall** must have the correct root partitioning scheme, which was updated with Release 1.7. If your target drive has not been properly formatted, select



1 from the Maintenance menu and follow the instructions. Refer to "Using the Format Command" on page 4-36 for additional information.

If your target drive is properly formatted, select **2** from the Maintenance menu to begin **NSinstall**.

If the TERM variable for your terminal is not set, the system displays a numbered list of terminal types and prompts you to enter the terminal type. Select the number from the menu that corresponds to the terminal type you are using:

- 1) NCD X terminal
- 2) ANSI-compatible terminal (including VT220 and Link MC5)
- 3) Wyse Model 30/50
- 4) Televideo 925
- 5) Sun Shell Tool
- 6) Sun Command Tool
- 7) DEC VT510
- 8) Other

Select a number for your terminal type >>

For example, select 7 for a DEC VT510 terminal or 2 for an ANSI-compatible terminal.

If you select **8** (Other), you are prompted to enter the terminal type, as shown in the following example. The type you enter must be in /etc/termcap and the terminfo database.

```
Enter terminal type (must be in both /etc/termcap and terminfo):
>>
```

After selecting the **NSinstall** option and entering your terminal type (if necessary), a message similar to the following appears:

```
*** WARNING ***
All data will be destroyed on the target disk!
NSinstall from the boot CD-ROM? ([y]/n)
```

After selecting **y**, the **NSinstall** form appears. Refer to the next section for information about the fields in the form.

#### **NSinstall Form**

**NSinstall** displays the form shown in Figure 6-1. Table 6-1 explains the meaning of each field. Some fields in the form appear only if you specify **Yes** for a related field. For example, the prompts for the partition and **newfs** appear only if you want to install or set up the related file system. These fields are indented in the table. Be sure to read the form carefully before executing it; some default values in the form may not be applicable and need to be changed.

Use the Arrow keys and Tab key to move the cursor within the form. Type a value in an input field, which is enclosed with square brackets ([]). Also, press the space bar to display the possible values for a select field, which is preceded by an angle bracket (>). For more information on cursor control, refer to "Using Configuration Command Forms" on page 2-9.

The log file for NSinstall, *NSinstall.log*, is in the /var/log directory of the target disk. A copy of the log also appears in the current root drive's /tmp directory.





Note: If the server was booted from the root disk, not a CD-ROM, NSinstall will not allow you to install file systems to the current root disk. In this case, you can only install file systems to a target disk in a slot other than slot 0.

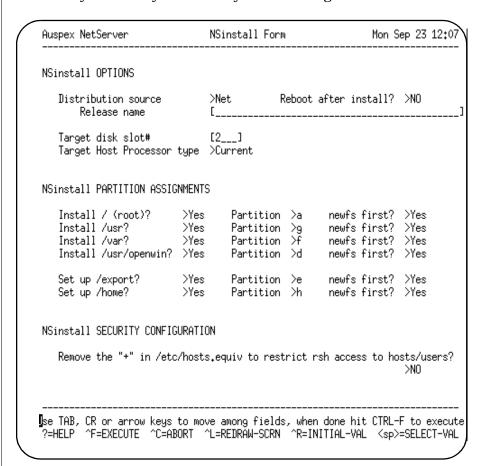


Figure 6-1. The NSinstall form

Table 6-1. The NSinstall form fields

Field	Description	Possible values
CD-ROM slot#	The number of the slot where the Auspex NetServer Software CD-ROM is inserted.	Default is 1. It can be any number appropriate to your server. (Slot numbering starts from 0.)
Target disk slot#	The number of the slot where the target drive is inserted. For example, to repair a damaged root drive, specify <b>0</b> . To install file systems to a spare root drive, enter the number of the slot where the spare drive is inserted.	The target drive can be in any slot so it can be any number appropriate to your server. Slot 0 is allowed only if your system was booted from the CD-ROM. Default is <b>3</b> .
Target Host Processor Type	Type of HP with which the newly installed software will be used.	<b>Current</b> (the HP currently installed on the current server), and <b>HP VIII</b> . Default is <b>Current</b> .



Table 6-1. The NSinstall form fields (Continued)

Field	Description	Possible values
Reboot after install?	Whether the system automatically reboots in single-user mode after the installation.	Default is <b>No</b> . Consider <b>Yes</b> only if you install the software to drive 0 (the current root drive). Use the default if you install to a slot other than slot 0.
Install / (root)?	Whether to install the <i>root</i> file system.	Yes when you do normal installations; No when you use NSinstall to repair a partition other than <i>root</i> .
Partition	The physical partition on which to build the root. This field appears only if you respond <b>Yes</b> to the "Install / (root)?" prompt.	Default is <b>a</b> .
newfs first?	Whether to initialize the root file system before installing the software. This field appears only if you respond <b>Yes</b> to the "Install / (root)?" prompt.	<b>Yes</b> if you want to initialize the file system in case it is damaged; <b>No</b> if you don't want to initialize the file system.
Install /usr?	Whether to install the /usr files.	Yes when you do normal installations; No when you use NSinstall to repair a partition other than /usr.
Partition	The physical partition on which to build /usr, usually, the <b>g</b> partition. This field appears only if you respond <b>Yes</b> to the "Install /usr?" prompt.	Default is <b>g</b> .
newfs first?	Whether to initialize the /usr file system before installing the software. This field appears only if you respond <b>Yes</b> to the "Install /usr?" prompt.	<b>Yes</b> if you want to initialize the file system in case it is damaged; <b>No</b> if you do not want to initialize the file system.
Install /var?	Whether to install the /var file system.	Yes when you do normal installations; No when you use NSinstall to repair a partition other than /var.
Partition	The physical partition on which to build /var. This field appears only if you respond <b>Yes</b> to the "Install /var?" prompt.	Default is f.
newfs first?	Specifies whether to initialize the /var file system before installing the software. This field appears only if you respond <b>Yes</b> to the "Install /var?" prompt.	<b>Yes</b> if you want to initialize the file system in case it is damaged; <b>No</b> if you do not want to initialize the file system.
Install /usr/openwin?	Whether to install the /usr/openwin files.	Yes when you do normal installations; No when you use NSinstall to repair a partition other than /usr/openwin.
Partition	The physical partition on which to build /usr/openwin. This field appears only if you respond <b>Yes</b> to the "Install /usr/openwin?" prompt.	Default is <b>d</b> .
newfs first?	Whether to initialize the /usr/openwin file system before installing the software. This field appears only if you respond <b>Yes</b> to the "Install /usr/openwin?" prompt.	<b>Yes</b> if you want to initialize the file system in case it is damaged; <b>No</b> if you do not want to initialize the file system.



Table 6-1. The NSinstall form fields (Continued)

Field	Description	Possible values
Set up /export?	Whether to set up the /export file system.	<b>Yes</b> when you install software for the first time; <b>No</b> when you reinstall.
Remove the "+" in /etc/hosts.equiv?	Whether to remove the single "+" in the file /etc/hosts.equiv.	<b>Yes</b> to restrict rsh access to the server and thereby make the server more secure.
Partition	The physical partition that you want to reserve for /export.	Default is <b>e.</b>
Set up /home?	Whether to set up the /home file system.	<b>Yes</b> when you install software for the first time; <b>NO</b> when you reinstall software. This option is automatically disabled if the target disk doesn't have the <b>h</b> partition.
partition	The physical partition that you want to reserve for /home.	Default is <b>h</b> .

### **Executing the NSinstall Form**

After filling out the NSinstall form, execute it by following these steps:

1. Execute the form by typing Ctrl-F. **NSinstall** displays a number of messages describing the operations performed for the installation. When the installation is finished, the following message is displayed:

Scratch install DONE.

If this is a partial install, please check the /etc/fstab file on the target disk to make sure that is has the desired information. Partial install is used for partition repair only, so the fstab file might not have the desired information.

You might want to check the log file /tmp/NSinstall.log on current root disk or /NSinstall.log on the target disk.

NSinstall session ended on date

Partial installation means that some partitions were not selected. The messages are also saved to /NSinstall.log on the target drive and /tmp/NSinstall.log on the current root drive.

If you used **NSinstall** to copy software from the CD to a spare disk, no system reboots are necessary after the installation; the NSinstall procedure is now complete. Remove the spare disk from the slot, and save it in a safe place.

The system may or may not reboot after **NSinstall**, depending on whether you enabled automatic reboot in the NSinstall form. If automatic reboot occurs, go to the next step. (If the automatic reboot brings the server up in multiuser mode, use the **shutdown** command to enter single-user mode before going to the next step.)

If you reinstalled files to the current root disk and specified no automatic reboot, enter the system maintenance commands that are appropriate to your server. After you finish with the commands, reboot the system in single-user mode.





**Note:** If optional software products were installed on your server before installing system software from the CD, you must reinstall the optional products before rebooting the server. NSinstall does not preserve optional products during installation, because older versions may not be compatible.

The key and license are preserved during the **NSinstall** procedure.

3. After the system boots in single-user mode, run NSconfig. Follow the procedures in "NSconfig" on page 2-10 to complete the configuration.

When **NSconfig** is finished, type **exit** to go to multiuser mode.

4. If you enabled **newfs** in the **NSinstall** form, you may have lost some of the server configuration information. To reconfigure your NetServer, run SetupTty, SetupExec, and SetupClient.



**Note:** If your server has been using a customized kernel, remember to reconfigure the kernel after running NSinstall. This ensures that the kernel contains the devices and software options appropriate for your environment.



# **7** Write Acceleration

# **About This Chapter**

This chapter covers the following topics:

- **▲** Overview of write acceleration
- ▲ Restrictions that apply to the write cache
- ▲ A description of the write cache states and the commands used to manage those states
- **▲** Instructions for enabling the write cache
- ▲ Instructions for disabling the write cache
- ▲ Instructions for managing the write cache, including error recovery procedures



## **Write Acceleration Overview**

Auspex NetServers offer an optional Write Accelerator to accelerate client NFS write operations. The Write Accelerator is a daughter board with nonvolatile cache memory, which fits onto the SP board. The terms "write cache" and "Write Accelerator" are used interchangeably in the NetServer's screen messages and in this guide.

The Write Accelerator offers these advantages that make it easy to manage:

- Write acceleration is available on an individual file system basis.
- The Auspex Performance Monitor statistics show write cache activities.
- Various commands are available for you to manage the Write Accelerator.



# How the Write Accelerator Improves Throughput

The following list explains how a NetServer equipped with a Write Accelerator achieves higher throughput:

▲ Faster response for an NFS write operation

Without write acceleration, a client NFS write operation is completed only after the data is successfully written to the disk drive on the server. With write acceleration, the NetServer puts the data into the cache memory on the Write Accelerator board first and immediately informs the NFS client that the write operation is complete. The client sees the faster response for NFS write operations.

More buffered I/O daemons (BIODs) available for concurrent NFS writes

A typical NFS client starts a number of BIODs in order to start several concurrent NFS write operations. For example, if an NFS client uses four BIODs to complete four conventional NFS writes, each BIOD must wait until its write to disk is completed before the BIODs are available for the client to use again. With the faster response time provided by the write cache, the BIODs do not have to wait for the write to disk to be completed. For the NFS client, the net result is more available BIODs, which translates to improved client throughput. The actual performance benefits for the client depend on what applications the client is running.

Reduction of disk I/Os

Given the large I/O memory cache provided by an NFS file server, most client NFS read operations get their data from the I/O memory cache. As a result, the majority of disk I/Os are write operations. The write cache design helps NetServer throughput in two ways. First, it discards redundant writes to the same disk location, as happens with inode and indirect block updates. Second, it coalesces sequential data blocks that remain in the cache and have not been written to disk. Each of these design features helps reduce the number of write operations to disk.

The Write Accelerator III on the SP V has 8 MB of NVRAM. The Write Accelerator includes a battery backup.



**Note:** Because the data written to the cache is not written immediately to disk, a disk or power failure can occur before the data is permanently stored. When this happens, the contents of the write cache are preserved until the error is corrected.

While the write cache is disabled, NFS writes are written to disk in the conventional manner.



**Caution:** Although the NVRAM in the Write Accelerator can prevent data loss during a power outage, the integrity of data in the cache is not guaranteed. Before writing cached data to disk, the SP checks to see if the control data in the cache is valid. If not, it changes the state of the Write Accelerator to BADCHECKSUM. Purge the data from the cache when this happens, because the SP cannot determine the disk location to which to write the data. Purging removes the data permanently from the cache. For more information on write cache states, refer to "Write Cache States and Commands" on page 7-6.



# **How the SP Maintains Drive Information**

The SP function ensures that the cached data, if any, is written to disk when the system is booted. The SP maintains a table of disk drive serial numbers and other cache control information that is checked at power up to determine if the disk drive configuration has changed. If it has not changed, the SP uses checksums to verify the integrity of the cache control data and then writes the data to disk.

When a new disk is added to the NetServer, the ax\_hot\_plug or ax\_add\_device command automatically updates the SP with the drive serial number to support the write cache function. (Every disk drive shipped with a NetServer has a unique serial number.) This information is used by the write cache to verify that information is written to the correct location.

Before a disk drive is removed from the NetServer, the ax\_hot\_plug or ax\_remove\_device command instructs the SP to flush the write cache.



# Write Accelerator Restrictions

The following restrictions apply to write acceleration:

- Only writes with 8 KB or less of data are supported by the Write Accelerator. Writes with more than 8 KB of data to transfer are treated as conventional writes.
- The backup battery has a minimum shelf life of two years at 70°F. When the battery power is low, the NetServer displays an error message to the console and attempts to flush data to disk before the write cache becomes unusable.
- The Write Accelerator III provides approximately 3 months of actual backup.
- In the NetServer, each SP that uses write caching must have a Write Accelerator board installed on it. The Write Accelerator cache memory is not shared among SP.
- Physical partitions must start on an 8-K boundary.
- An SP V must have the latest PROM/Flash code to access the full 8-MB Write Accelerator. Look at the Write Accelerator notes in your system's hardware manual, or check with your Auspex field representative to determine the PROM/Flash code level of your SP V.



# Write Cache States and Commands

Understanding the state of the write cache is important for managing it. The SP keeps track of seven cache states:

If the write cache is present on the SP, it powers up in the UNINITIALIZED

> UNINITIALIZED state. The write cache is initialized automatically at boot time from an entry in /etc/rc.boot. During initialization, the write cache changes states several times, as described in the

following paragraphs and Figure 7-1.

The state is set to OFF in either of these situations: OFF

- If no unwritten data is found in the cache at initialization, the

write cache state is set to OFF.

- If unwritten data is found, checksums validate the integrity of the cache control data. If the control data is valid, the contents of the cache are written to the appropriate disk. After all the data in the

cache is written successfully, the state switches to OFF.

If unwritten data is found in the write cache at initialization but the BADCHECKSUM

cache control data is invalid, the state switches to BADCHECKSUM.

If the data in the cache cannot be written to disk because of disk DIRTY

errors or disk changes, the state is set to DIRTY.

The ON state indicates the write cache was successfully enabled. ON

The voltage of the battery is low, and the write cache is disabled. BATTERY\_LOW

This state occurs when no NVRAM is installed on the SP. NONE

If the write cache state is BADCHECKSUM or DIRTY after initialization, the server enters single-user mode, and you can use the ax\_write\_cache command to change the state of the write cache as described in "Managing the Write Accelerator" on page 7-9.



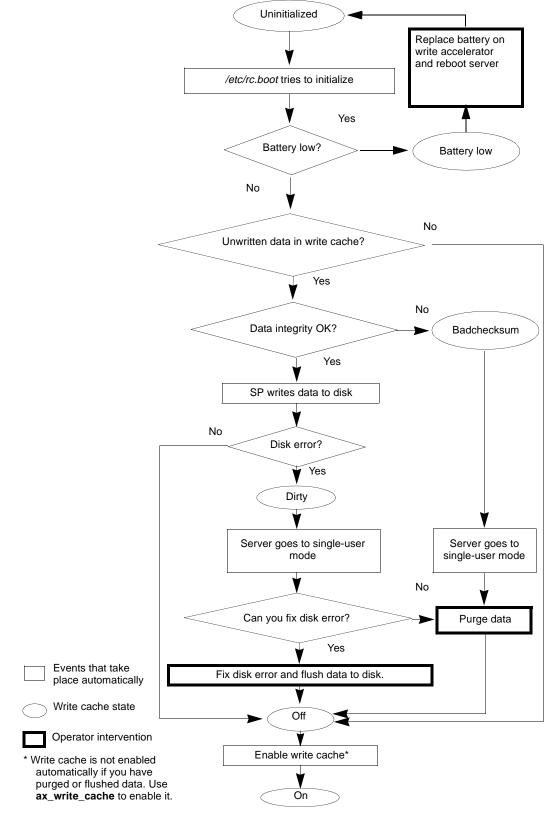


Figure 7-1. Write cache changing states after server power on





**Note:** Because write caching is not supported under UFS, all write messages to root, /var, and /usr file systems bypass the write cache. Consequently, the state of the write cache does not affect the system boot files.

A utility command, **ax\_write\_cache**(8), manages the state of the write cache. You must be root to invoke the command. The syntax for the command is as follows:

```
ax_write_cache [ -v ] [ -s SP_number ] option...
```

The -v argument executes the command in verbose mode. If no SP numbers are specified, the command applies to all SPs in the system. (Processor numbering starts from 0.) The command options, which are described in Table 7-1, are for changing the state of the write cache or printing out information about the write cache. You can use more than one option per command.

Table 7-1. ax\_write\_cache command options

Option	Description
init	Initializes the write cache and, if all data successfully writes to disk, switches it to the OFF state.
enable	Puts the write cache in the ${\tt ON}$ state. You can set the write cache to ${\tt ON}$ only from the ${\tt OFF}$ state.
disable	Puts the write cache in the OFF state. You can set the write cache to OFF only from the ON state. This option flushes all the cached blocks. If an error occurs while flushing these blocks, the write cache enters the DIRTY state instead of the OFF state.
purge	Discards all unwritten data in the write cache, and then switches the state from DIRTY or BADCHECKSUM to OFF.
flush	Disables the write cache, and writes all data in the write cache to disk. This only applies if the write cache state is DIRTY. If the flush is successful, the write cache state becomes OFF. If the flush is unsuccessful, the write cache state remains DIRTY.
state	Displays the write cache states: UNINITIALIZED, OFF, DIRTY, ON, BADCHECKSUM, or NONE.
errors	Prints a list of cache pages that cannot be written to disk. This only applies if the write cache state is DIRTY.

To examine write cache performance statistics, use the **ax\_perfmon** command.

# **Enabling the Write Accelerator**

Enabling the Write Accelerator involves invoking the **ax\_write\_cache** command and modifying /etc/fstab.

The cache memory is automatically initialized and switched to the ON state (enabled) at boot time.

Write acceleration is enabled only after the cache is turned on and an entry added to <code>/etc/fstab</code> specifying the file system you want to enable. For example, to enable write caching for the file system <code>/home/dvlp</code>, create the following entry:

```
/\text{dev}/\text{ad30c} /\text{home}/\text{dvlp} lfs fs=4.2,wc 0 2
```

In this example, wc (write cache) enables write acceleration for the file system, /home/dvlp.





**Note:** The wc option is ignored if no Write Accelerator is installed on the SP, in which case NFS write operations are handled conventionally.

To enable write acceleration on SP0 from the OFF state, enter the following command:

```
ax_write_cache -s 0 enable
```

The SP automatically disables write acceleration when the cache state switches to BADCHECKSUM or DIRTY. Refer to "Managing the Write Accelerator" on page 7-9 for error recovery procedures.

## Disabling the Write Accelerator

The write cache is automatically initialized and switched to the ON state (enabled) at boot time.

Disable the write cache using any of the following methods:

Use the ax write cache command to switch the cache state to OFF. For example, when the write cache state on the SP0 is ON, enter the following command at the root prompt to change it to OFF:

```
ax_write_cache -s 0 disable
```

To disable the write cache function for a specific file system, remove wc from the specific file system entry in the /etc/fstab file. Then unmount and mount that file system.

Additionally, the SP automatically disables the write acceleration function when the cache state changes from UNINITIALIZED or ON, to BADCHECKSUM or DIRTY. Refer to "Managing the Write Accelerator" on page 7-9 for error recovery procedures.

# Managing the Write Accelerator

The ax write cache command is used to manage the write cache in a number of ways. This section describes the following procedures:

- Recovering from a write cache (BADCHECKSUM or DIRTY) error
- Recovering from a disk media error
- Recovering from a disk hardware error
- Recovering from a system configuration change

Each of these procedures requires that you log in as root.

### Recovering From Write Cache (BADCHECKSUM or DIRTY) Errors

If the SP indicates that the write cache is in a DIRTY or BADCHECKSUM state after initialization, follow this procedure:

1. Enter the ax\_write\_cache command to verify the cache state. For example, if the write cache is on SP0. enter:

```
ax write cache -s 0 state
```

2. Follow either of these steps depending on the write cache state:



a. If the state is BADCHECKSUM, enter the following command, and then go to step 3:

```
ax_write_cache -s 0 purge
```

b. If the state is DIRTY, flush the unwritten data to disk:

```
ax_write_cache -s 0 flush
```

This command tries to write all data currently in the write cache to the file systems that use write caching. If all data is successfully written to disk, the write cache switches to OFF, and you can go to step 3.

If a message indicates that the state of the write cache is still DIRTY, you may have a disk problem that prevents some data from being flushed. Go to "Recovering From a Disk Media Error (DIRTY State)".

3. Re-enable the write cache by entering the following:

```
ax_write_cache -s 0 enable
```

#### Recovering From a Disk Media Error (DIRTY State)

If the write cache state is DIRTY, you may have a malfunctioning disk that prevents the data from being written. This section provides an example for recovering a disk media error through the **ax\_write\_cache** command on SP0:

1. If the ax\_write\_cache flush command returns a message indicating the write cache is still dirty, look for the SP error message logged in the /var/adm/messages file that reports the disk error. Use ax\_sputil reassign to reassign the sector that caused the problem, and then try to flush the data again.

If the flush succeeds, go to step 2. If it fails and you want to continue using the write cache, follow these steps:

a. Enter the following command to purge all data from the cache:

```
ax_write_cache -s 0 purge
```

- b. If you cannot flush data to disk, it is likely that the disk is bad. You should replace the disk, use **newfs** to create a file system on the new disk and restore the contents of the file system from the backup tape. Then go to the next step.
- 2. Re-enable the write cache by entering the following:

```
ax write cache -s 0 enable
```

#### Recovering From a Disk Hardware Error

This section provides an example for recovering a disk hardware error through the **ax\_write\_cache** command on SP0:

1. Display the write cache error pages to verify the hardware error:

```
ax_write_cache -s 0 errors
```

- 2. Run ax\_kill(8) to kill all processes sleeping on file systems that may have become isolated because of disk failure.
- 3. Unmount the affected file system.
- 4. Run **ax\_hot\_plug** to remove and reseat the drive that failed. (Reseating recycles power to the drive, which sometimes corrects drive problems.)



5. Check the drive LED to see if the drive is working. If the LED remains on or flashes continuously, go to step 6. If the LED flashes briefly and then goes out, reseating the drive may have corrected the error. If so, enter the following command:

```
ax_write_cache flush -s 0
```

If the flush succeeds, reseating the drive corrected the problem, and the recovery procedure is complete. If the state of the cache is still DIRTY, go to step 6.

- 6. Run **ax\_hot\_plug** to replace the problem drive.
- 7. After replacing the drive, purge the write cache by entering the following command:

```
ax_write_cache -s 0 purge
```

8. Verify the cache state by entering the following command:

```
ax_write_cache -s 0 state
```

If the state is OFF, the purge was successful. If the state is not OFF, contact Auspex Technical Support for assistance.

9. Enable the write cache by entering the following command:

```
ax write cache -s 0 enable
```

- 10. Partition and label the new disk in the same way you prepared the old disk.
- 11. Use **newfs** to create file systems on the disk, and restore the data from tape backups.
- 12. Remount the file system.

#### Recovering From a Drive Configuration Change

This section provides an example for recovering from a drive configuration change on SP0:

1. Enter the **ax\_write\_cache** command to verify the cache state:

```
ax_write_cache -s 0 state
```

2. If the cache state is OFF, go to step 4. Otherwise, flush the unwritten data to disk:

```
ax_write_cache -s 0 flush
```

3. Purge all other data from the cache:

```
ax write cache -s 0 purge
```

4. Re-enable the write cache by entering the following:

```
ax_write_cache -s 0 enable
```



# Backing Up the NetServer

# **About This Chapter**

A good backup strategy is vital to the protection of your data. A program of regularly scheduled backups protects your data from corruption because of a system crash or other cause, and protects user files against accidental deletion.



**Note:** Familiarize yourself with the Auspex tape drive naming conventions before attempting to back up or restore file systems. Refer to "Tape Drives" on page 4-2 for more information on how to name the drives.

This chapter, in three major parts, summarizes how to back up and restore file systems on the NetServer:

Backing up a physical disk partition to tape, which must be performed when the partition is offline.

The information includes:

- using the **dump** and **restore** commands
- calculating the amount of tape to use for a backup
- a suggested scheme for backing up a system with 40 drives
- backing up the root disk
- Backing up a file system while it is online (that is, cloning an active file system).
- Restoring a file with a damaged inode.

Remember that no matter how effective your backup strategy is, if you are not following the recommended procedures for cleaning and maintaining your tape drives and tape media, you risk losing valuable data. For more information, refer to "Preventive Maintenance" in the hardware manual for your NetServer.

For a more detailed description of backups and how to perform them, see Sun's System and Network Administration.



**Recommendation:** Although a mirrored partition protects data on the partition from loss because of disk or media errors, mirroring cannot protect the data from loss because of other causes such as accidental deletion. Follow a regular backup routine for mirrored partitions as for other partitions.



# **Using the Dump and Restore Commands**

The **dump** command is a flexible and effective command for performing backups at several levels of completeness. A full system backup copies everything on the file system, while an incremental one backs up only selected parts of the system. The dump command allows you to specify a dump level from 0 to 9. Dump level 0 is a full system backup, and each subsequent dump level (1 to 9) backs up only those files that have changed since the last dump of a lower level.

The counterpart of the **dump** command is **restore**, which recovers file systems that were backed up by **dump**.

## **Dump Command**

The **dump** command has the following syntax:

/usr/etc/dump options tape\_device\_name file\_system\_to\_dump

Table 8-1 describes the command arguments. Refer to the **dump**(8) man page for more information.

Table 8-1. Arguments for the dump command

Argument	Meaning
options	The most commonly used options are listed as follows:
0 – 9	Dump level you have chosen
b	Blocking factor (number of blocks written at a time)
d	Tape density
f	Name of the device to which dump backs up the file system
s	Size of the tape in feet
u	Writes the date of the dump and partition name to /etc/dumpdates
T	Date string you want dump to insert in /etc/dumpdates. By default, the current time (at which dump starts) is supplied. Use this option only when you use a script to dump a file system that has been cloned by ax_clonefs. In the script, create a timestamp, such as d1, using a date string to specify the time when ax_clonefs finishes copying the file system. This ensures that the next incremental dump backs up all the changes since the last time the file system was cloned.  The timestamp should use the following date string: 'date '+%a %h %d %T 19%y''  The following is an example of a dump command with the T option: dump OTdsbfu "\$d1" 141000 11500 126 /dev/rast4c /dev/ad0c
	where <b>d1</b> is the timestamp.



Table 8-1. Arguments for the dump command (Continued)

Argument	Meaning
m	Name of the partition to be entered in /etc/dumpdates. By default, dump writes to /etc/dumpdates the name of the physical partition being backed up. Give a name only if you are dumping a clone partition. The name should be that of the source partition in ax_clonefs so future incremental dumps can obtain the appropriate information about the partition in /etc/dumpdates. The following is an example:  dump Odsbmfu 65000 6700 126 /dev/rvp10 /dev/rast4 /dev/rvp11  In this example, the partition name specified with the m option is /dev/rvp10. It is recorded in /etc/dumpdates. The name of the clone partition, /dev/rvp11 is not recorded.
V	Verifies the contents of the tape.
tape_device_name	The device name of the tape device to be used for the dump. This must be specified if the ${\bf f}$ option is used.
/dev/nrastn, /dev/nrastnlo, /dev/nrastnc, /dev/nrastnloc	The no-rewind-on-close option must be used if you copy more than one file system onto the tape during the dump procedure.  The <b>n</b> in the device name is the slot number in the range of available slots. Append <b>lo</b> , <b>c</b> , or <b>loc</b> to the device name to indicate whether compression is used. Refer to Table 4-2 and Table 4-3 on page 4-3 for more information on the relationship between device names and data compression.
/dev/rastn, /dev/rastnlo, /dev/rastnc, /dev/rastnloc	The rewind-on-close option rewinds the tape after a file is written to the tape This option is not used when dumping more than one file. The <b>n</b> in the device name is the slot number in the range of available slots. Append <b>Io</b> , <b>c</b> , or <b>Ioc</b> to the device name to indicate whether compression is used. Refer to Table 4-2 and Table 4-3 on 85-0515-001 for more information on the relationship between device names and data compression.
file_system_to_dump	Name of the file system to be backed up. You can specify the file system name or the physical partition name on which the file system is mounted.

For more information for the **b**, **d**, and **s** options for the **dump** command, refer to "The Size, Density, and Blocking Factor Options for Dump" on page 8-7.



**Caution:** If you use 4-mm or 8-mm tapes to back up file systems, use only digital quality tapes to prevent data loss and avoid tape drive damage.

#### **Restore Command**

If you want to restore dump files back onto disk, use the **restore**(8) command. You can restore a dump file in noninteractive mode, or you can use interactive mode to select specific files on the tape.

When using **restore** to replace dump files onto disk, be aware of the following:

- Before restoring the files, first mount the target partition.
- Files are restored in a relative path from the point of execution.
- **restore** overwrites existing files.
- If the system reboots, the tape automatically rewinds.

The **restore** command has the following syntax:

/usr/etc/restore option [ filename...]



Option is a character that can be followed by a modifier or argument. It determines how the system restores the files (for example, whether to restore files interactively, whether to list files on a tape using a table of contents, and so on). For more information on available options, see the **restore**(8) man page. For more information on the strategy to restore a file system, refer to Sun's System and Network Administration.



**Recommendation:** For HP-mounted file systems, you can speed up restores by using the -F modifier with the restore command. This modifier enables delayed writes of file system information to the disk, allowing restore to run about three times faster. You can also query and enable delay writes with the fastfs command. For more information on available options for delaying file system writes, see the **fastfs**(8) man page.

System backups and restores with **dump** and **restore** can be run with the system in one of three different modes:

- On a quiescent file system in multiuser mode
- On a quiescent file system in single-user mode
- On an active file system (although this is not recommended)

The dump and restore commands used for each mode are similar. See Sun's System and *Network Administration* for **dump** and **restore** examples for each of these operating modes. Also, refer to "Restoring a File With a Damaged Inode" on page 8-17 for an example of restoring a file whose inode is damaged.



# **Tape Drive Capacity**

This section describes the capacity of each type of tape drive supported by Auspex. Use the information in this section to estimate the number of tapes needed to back up your server.

The maximum capacity of an Auspex tape drive depends on the following factors:

Type of tape drive (8-mm, DLT4000, and so on)



**Note:** The DLT4000 tape drive requires a DLT4000 drive rack.

- Length of tape used in the drive (60-meter, 90-meter, 112-meter, and so on)
- Whether data compression is enabled, which is determined by the following:
  - For 4-mm drives, the switch 2 setting on the drive, the device name used in software commands (for example, rast5 and rast5c), and the prerecorded data format on the tape
  - For 8-mm and DLT4000 tape drives, the device name used in software commands

### Data Format on 4-mm Tapes

Switch 2 on the 4-mm tape drives affects the data format (DDS or compression) used on the tape. When the switch is OFF, the drive can be used in low-density or high-density mode; when it is ON, the drive can be used in high-density mode only. For the switch 2 location and information on how the switch setting works with device naming to determine the data format, refer to your NetServer's hardware manual. This section only describes how different models of the 4-mm tape drives define "DDS format."

The NetServer supports two models of 4-mm tape drives: WangDAT 2600 and WangDAT 2000. The term "low density" has different meanings for these models.

The WangDAT 2000 uses industry-standard DDS when switch 2 is set to OFF and when you use rastnlo to name the drive. On the WangDAT 2600, you can also set switch 2 to OFF and use **rastnlo** to name the drive, but the drive operates in a low-density mode that is different than the standard DDS. A tape with data recorded in standard DDS format cannot be read by a WangDAT 2600 drive operating in low-density mode. Similarly, data recorded in low-density format on a WangDAT 2600 drive cannot be read by another tape drive that uses standard DDS.

To determine which model of 4-mm tape drive is installed in a particular slot, use one of the following methods:

- If a sticker on the drive carrier shows that the drive accepts both 60-m or 90-m tapes, the model is WangDAT 2000.
- If no sticker is attached to your drive carrier, remove the drive and check the bar code number on the back of the drive. The number under the bar code is 3105 for WangDAT 2000 and 3103 for WangDAT 2600.

# **Tape Drive Capacities**

Table 8-2 lists the approximate capacity of each type of tape drive. The device name indicates the mode of operation. For example, rastnc indicates that data compression in



Exabyte 8500 format is used. For more information on the relationship between device names and data compression, refer to Table 4-2 and Table 4-3 on page 4-3.



**Note:** When using data compression, you may or may not be able to store to the tape the amount of data indicated in Table 8-2, depending on how compressible your data is.

The table lists the capacity of the 112-meter tape in an Exabyte 8200 drive if the device name is rast*n*lo. Because this drive does not use data compression, do not include "lo" in the device name for an Exabyte 8200 tape drive.

Table 8-2. Tape drive data capacities

Tape and drive				
type	rast <i>n</i> lo	rast <i>n</i> loc	rast <i>n</i>	rast <i>n</i> c
4-mm (60-meter), WangDAT 2600	1.3 GB (WangDAT DDS)	Not applicable	2.6 GB (WangDAT group compression) or 1.3 GB (WangDAT DDS)	2.6 GB
4-mm (90-meter) WangDAT 2000	1.95 GB (Standard DDS)	Not applicable	3.9 GB (WangDAT group compression) or 1.95 GB (Standard DDS)	3.9 GB
8-mm (112-meter) Exabyte 8200	2.3 GB	Not applicable	2.3 GB	Not applicable
8-mm (112-meter) Exabyte 8500	2.3 GB	Not applicable	4.7 GB	Not applicable
8-mm (112-meter) Exabyte 8505	2.3 GB	4.7 GB	5 GB	10 GB
8-mm (112 or 160-meter) Exabyte 8505XL	2.3 GB 3.5 GB	4.7 GB 7 GB	5 GB 7 GB	10 GB 14 GB
Quantum CompacTape IV (1800-feet) DLT4000	10 GB	20 GB	20 GB	40 GB
1/4-inch (600-feet) QIC-150	150 MB	Not applicable	150 MB	Not applicable
1/2-inch (2400-feet)	40 MB (1,600 BPI)	Not applicable	150 MB (6,250 BPI)	Not applicable

Auspex drives are set at the factory to the highest capacity listed in Table 8-2. To operate your drive at the lower capacity, name your drive following the naming conventions described in Table 4-2 and Table 4-3 on page 4-3.



**Note:** If you want to dump data to a 4-mm tape without data compression (for example, if you specify /dev/rast1lo in dump), make sure there is no compressed data prerecorded on the tape. Trying to back up uncompressed data to a 4-mm tape containing compressed data generates a tape write error. Always check the LED on the tape drive before issuing the **dump** command. If the tape inside the drive contains compressed data, the LED is green; if the tape contains low-density data, the LED is amber.



To erase prerecorded data on the tape, use the **mt** command. For example:

mt -f /dev/rast1lo erase

## The Size, Density, and Blocking Factor Options for Dump

Table 8-3 lists suggested arguments to use with the **dump** and **restore** commands for the Auspex tapes listed in Table 8-2.



**Caution:** In the **dump** command, do not use a blocking factor greater than 126. Doing so may cause data to be unreadable when you try to restore it. For example, suppose you specify the blocking factor to be 512, which means that you try to write blocks of 262144 bytes (512 bytes x 512) to tape. Before the data is written, each block of 262,144 bytes is broken up into smaller blocks, because the value 262,144 exceeds the maximum block size supported by the server. The actual size of data blocks written to tape is equal to the maximum block size supported by the tape driver, which depends on the HP model. In this example, if you use **restore** with the same blocking factor (512) to restore data that has been backed up on an HP IV-based server, the data cannot be read because the block size **restore** expects to see (65,532 bytes) is different from the actual block size on the tape (64,512 bytes).

Table 8-3. Suggested dump and restore arguments

Drive Type	Tape capacity	Option	Argument
4-mm (60-meter)	2.6 GB	size density blocking factor	6300 110400 100
	1.3 GB	size density blocking factor	4100 55000 100
4-mm (90-meter)	3.9 GB	size density blocking factor	9600 110400 100
	1.95 GB	size density blocking factor	6100 55000 100
8-mm (112-meter) Exabyte 8200	2.3 GB	size density blocking factor	6700 65000 126
8-mm (112-meter) Exabyte 8500	4.7 GB	size density blocking factor	11500 141000 126
	2.3 GB	size density blocking factor	6700 65000 126

Table 8-3. Suggested dump and restore arguments (Continued)

Drive Type	Tape capacity	Option	Argument
8-mm (112-meter) Exabyte 8505	9.8 GB	size density blocking factor	11500 141000 126
	4.9 GB	size density blocking factor	11500 141000 126
	2.3 GB	size density blocking factor	6700 65000 126
8-mm (160-meter) Exabyte 8505XL (A 112-meter tape in this drive	14.0 GB	size density blocking factor	16282 141000 126
has the same capacities as with the 8505 drive above.)	7.0 GB	size density blocking factor	8141 141000 126
Quantum CompacTape IV (1800-feet) DLT4000	40.0 GB	size density blocking factor	115000 81633 126
	20.0 GB (native)	size density blocking factor	115000 81633 126
	20.0 GB (compressed)	size density blocking factor	76800 62500 126
	10.0 GB	size density blocking factor	76800 62500 126
1/4-inch (600 feet)	150.0 MB	size density blocking factor	14400 1000 126
1/2-inch (2400 feet)	150.0 MB	size density blocking factor	2400 6250 126
	40.0 MB	size density blocking factor	2400 1600 126



**Note:** For the 4-mm and 8-mm tapes listed in Table 8-3, *size* arguments compensate for the inter-record gap used in the **dump** program. Using *size* to compensate for this gap forces **dump** to use the full capacity of the tape. For 1/4- and 1/2-inch tapes, **dump** uses an inter-record gap of 0.12 inch; however, there is no inter-record gap for 4-mm and 8-mm tapes. Without compensating for this gap, **dump** uses only part of the tape.

Also, the suggested value for *size* has made allowance for rewriting bad blocks.



For tape lengths not listed in Table 8-2, you can estimate tape capacity based on the length of the tape. For example, assuming you use the same density and blocking factor, the capacity of a 54-meter tape in an Exabyte 8200 drive is about half the capacity of a 112meter tape in the same drive.

Table 8-4 lists dump and restore command examples for some tape drives. These examples assume that you install a 4-mm tape drive as rast1, an 8-mm tape drive as rast2, a DLT4000 tape drive as rast7, a 1/4-inch tape drive as rast3, and a 1/2-inch tape drive as rast19.

Table 8-4. Dump and restore command examples

Drive type	Tape size	Command
		dump
4 mm	60 meter	dump 0fusdb /dev/rast1 6300 110400 126 /dev/rad0a
8 mm	60 meter (Exabyte 8200)	dump 0fusdb /dev/rast2 6700 65000 126 /dev/rad0a
DLT4000	1800 feet	dump 0fusdb /dev/rast7 115000 81633 126 /dev/rad0a
1/4-inch	600 feet	dump 0fusdb /dev/rast3 600 1000 126 /dev/rad0g
1/2-inch	2400 feet	dump 0fudb /dev/rast19 6250 126 /dev/rad0h
		restore*
4 mm	60 meter	restore /dev/rast1
8 mm	90 meter	restore /dev/rast2
DLT4000	1800 feet	restore /dev/rast7
1/4-inch	600 feet	restore /dev/rast3
1/2-inch	2400 feet	restore /dev/rast19

<sup>\*</sup> If you omit the b option in the restore command, restore attempts to figure out the block size for the tape.



**Note:** When using **dump** to back up a partition larger than 2 GB, remember to specify the raw partition name, not the special block device name. This applies to both physical and virtual partitions. For example, use /dev/rad2c and /dev/rvp2 to dump ad2c and vp2, respectively. If you specify /dev/ad2c or /dev/vp2 in the dump command, this error message appears:

bread: lseek fails



# **Backing Up With a Large Number of Disks**

The procedures for automated backups for a NetServer with fewer than 50 GB of storage are the same as the procedures for other servers with a comparable amount of disk storage. However, for a server with more than 50 GB of storage, implementing a scheme for unattended, automated backups requires planning. This section provides some recommendations for backing up large-capacity NetServers. You may wish to modify the scheme to suit your environment and needs. You need to create the cron(8) scripts that execute the backups; the scripts are not supplied by Auspex.

The scheme for unattended backups shown in Figure 8-1 on page 8-10 makes the following assumptions:

- The NetServer has 40 4-GB disk drives (arranged in 80 2-GB file systems), and one or more tape backup systems attached to the HP SCSI.
- File systems average less than 10% change per week.
- Backups are performed five nights per week. In Figure 8-1 below, 0 represents a level 0 dump, and the numbers 1, 5, and 9 represent different levels of incremental dump.
- Level 0 dumps are performed biweekly on each file system; level 0 dumps are distributed over five nights, not performed all on the same night.
- Each night's level 0 dump of a single file system fits easily on one tape.
- Incremental dumps (alternating levels 1, 5, and 9) are performed on each file system on all nights when level 0 dumps are not performed.

Week 1 File System	Sun	Mon	Tue	Wed	Thu	Fri	Sat
1–16	_	0	5	9	5	9	_
16–32	_	5	0	5	9	5	_
33–48	_	9	5	0	5	9	_
49–64	_	5	9	5	0	5	_
65–80	_	9	5	9	5	0	_
Week 2	C	Man	Torre	VA/a al	The	F-:	Cot
File System	Sun	Mon	Tue	Wed	Thu	Fri	Sat
1–16	_	1	5	9	5	9	_
16–32	_	5	1	5	9	5	_
33–48	_	9	5	1	5	9	_
49–64	_	5	9	5	1	5	_
65–80		9	5	9	5	1	

Figure 8-1. Backup scheme for 42 drives



### **Backing Up the Root Disk**

Because NetServer software prevents the root and /usr partition from becoming members of virtual partitions, and because the NetServer requires swap space on the root disk, the root disk cannot be mirrored or hot-plugged. Auspex recommends that you back up the a and g partitions of the root disk onto tape.

Additionally, you can replicate the root partitions onto another disk drive for use as a backup boot source in case the root disk fails. You can then quickly replace a damaged root disk by moving the backup drive into slot 0. Auspex recommends that you use the online backup procedure described in "Online Backup" on page 8-12 to copy a root disk.



**Recommendation:** If you changed the root drive configuration, use online backup to copy the drive only after you have used it to boot the system successfully. Then use **dump** to back up daily incremental changes to files on the backup drive. If you perform online backups for the entire drive every day without first verifying that the modified drive is bootable, you risk copying corrupted data or configuration errors to the backup drive, which may not be able to boot the system if the current root drive fails.



# **Online Backup**

The backup procedures in the previous sections require that the file system be inactive when being copied to tape. This section describes how to back up a file system while it is still active. The online backup procedure involves these two major steps:

- Create a virtual partition as a clone of an active file system to be backed up.
- Back up the clone partition to tape. (Details about **dump** are not repeated in the following sections. See "Using the Dump and Restore Commands" on page 8-2 for information.)

# Cloning a Virtual or Physical Partition

A clone is a physical or virtual (striped or concatenated) partition that duplicates the data in another partition, which can also be physical or virtual. The partition being duplicated is called the source partition. Think of a clone as a snapshot of the source partition, which can be resident on another SP.

Because the clone and the source partition contain the same data immediately after the "snapshot" is taken, the partitions look like the members of a mirrored partition. However, there are significant differences between cloning and mirroring, as described in Table 8-5.

Table 8-5. Differences between cloning and mirroring a partition

Cloning	Mirroring
As data is copied from the source partition to the clone, NFS read operations are performed only on the source partition, not on its clone. (Write operations are performed on both partitions.)	A read request can be processed by either member of a mirrored partition.
The clone and the source partition can reside on different SPs.	Both members of a mirrored partition must reside on the same SP.
Both the clone and source partitions can be either virtual or physical.	Both members of a mirrored partition must be virtual partitions.

The purpose of creating a clone is to back up a large partition without taking the partition offline. Backing up file systems by cloning offers these advantages:

- You can back up a large virtual partition to another partition on another SP. Without cloning, online backup requires a mirrored partition to be created on the same SP as the virtual partition that needs to be backed up. You may not be able to find enough space on disks that are on the same SP.
- You can create a clone to take a snapshot of a mirrored partition. When you back up the files from the clone partition to tape, the members of the mirrored partition continue to function uninterrupted. Without cloning, you must detach a member from the mirrored partition to back up files to tape and, as a result, temporarily lose the advantages of mirroring.

The command used for cloning a partition is **ax\_clonefs**. Its syntax is as follows:

```
ax_clonefs [ -b ] [ -p ] source_partition clone_partition
```



The -b option logs errors to syslogd. The -p option shows the percentage of the file system that has been copied when **ax\_clonefs** is in progress. You can specify the file system name or partition name for the source partition (for example, */usr/openwin* or **vp3**). Specify the partition name for the clone partition (for example, **vp2** or **ad4a**).

### Important Guidelines for Creating a Clone Partition

Read and remember the following guidelines before you attempt to create a clone partition:

- A clone must be at least the same size as the partition that it duplicates. Figure 8-2 shows a sample clone of a mirrored partition on SP0. The clone consists of six concatenated physical disk partitions on SP1, which duplicate all write operations on the mirrored partition.
- ▲ Before selecting a clone partition, *always* use **ax\_diskconf** to make sure the clone partition is not a member of a two-membered mirrored partition or an underlying physical partition of any virtual partition. For example, to back up **vp1** to **vp3** using **ax\_clonefs**, make sure **vp3** is not a member of a mirrored partition. If **vp3** is a member of **vp2**, **ax\_clonefs** overwrites the data currently in **vp3**, and the members of the mirror no longer contain the same data.
- ▲ The clone partition is for backup purposes only. Do not mount the clone partition or its underlying physical partitions and modify its contents after **ax\_clonefs** has started. Auspex recommends you record the correspondence between each pair of source and clone partitions to avoid accidentally mounting a clone partition.
- ▲ Cloning a partition that spans other partitions (such as c on a root drive) is not supported.



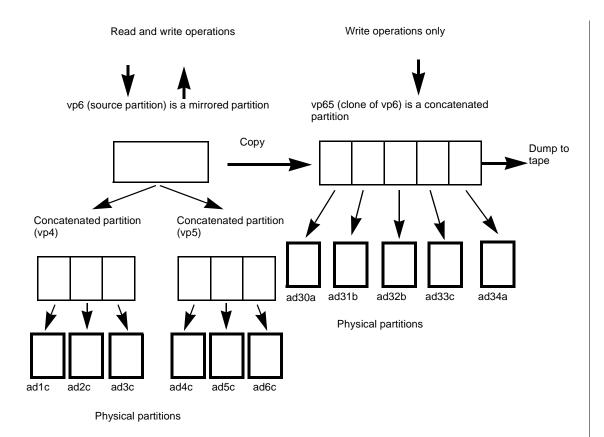


Figure 8-2. Example of a clone of a mirrored partition

# **Procedure for Cloning a Partition**

This section describes cloning a partition. You can use the procedure to back up the entire root drive to another drive. Before you start, use the **dkinfo** command to verify that each partition on the target drive is at least the same size as each partition on the root drive.

Follow these steps to clone a partition:

- The clone partition must be at least the same size as the source partition. If you need a
  larger clone partition, create the clone as a concatenated or striped virtual partition.
  The clone and source partitions can be on different SPs. Refer to "Defining and
  Reconfiguring Virtual Partitions" on page 5-14 for information on creating a virtual
  partition.
- 2. Attach the clone partition to the source partition using **ax\_clonefs** as in this example:

#### ax\_clonefs vp6 vp265

In this example, **vp6** is the source partition, and **vp265** is the clone partition. For the source partition, you can also use the file system name. For example, if the */usr/openwin* file system is mounted on **vp6**, enter the following:

#### ax clonefs /usr/openwin vp265

To back up your root disk, use **ax\_clonefs** for each partition on the root disk. For example, if you are copying the root disk to a disk in slot 4, enter the following command to copy the **a** partition:

ax clonefs ad0a ad4a



The **ax\_clonefs** command causes the NetServer to perform the following:

- ▲ Copy data from the source partition to the clone partition. During the copy, all NFS write operations are performed on both the clone and source partitions, and read operations are performed on the source partition alone.
- ▲ Lock the file system and sync the meta and user data to ensure consistency between the source and clone partitions.
- ▲ The clone partition is detached from the source partition.



**Caution:** When the SP is running **ax\_clonefs**, do not hot-plug devices using **ax\_hot\_plug**, **ax\_remove\_device**, or **ax\_add\_device** on the SP. Doing so hangs the system.

Also, never run **ax\_clonefs** and **ax\_mrestore** on the same mirrored partition simultaneously. The system might crash if you try to clone a mirrored partition and restore the contents from one member to another at the same time.

- 3. Run **fsck** on the cloned partitions.
- 4. The server prompt appears when the data copy procedure is finished. You can now dump the contents of the clone partition to tape. If you plan to do incremental backups on the partition, we recommend you issue the **dump** command on the same command line as **ax\_clonefs**. Refer to "Dumping a Clone Partition to Tape" for information on using **ax\_clonefs** and **dump**.



**Caution:** Never use the same clone partition for backing up different source partitions simultaneously. For example, if you execute **ax\_clonefs vp3 vp2** before **ax\_clonefs vp1 vp2** is finished, the second **ax\_clonefs** generates an error message, and the data backed up in the first **ax\_clonefs** may be corrupted.

# **Dumping a Clone Partition to Tape**

If you use **ax\_clonefs** to back up a file system online, remember the following regarding the subsequent **dump** command that copies the clone partition to tape:

▲ Always issue the **dump** command immediately after **ax\_clonefs** unless you use **dump** with the **T** option (see the next bullet item). For example:

ax\_clonefs ad1c ad2c;dump 0dsbfu 141000 11500 126 /dev/rast4c /dev/ad2c

- In this example, the date string that **dump** inserts in /etc/dumpdates is also the time when **ax\_clonefs** is finished. When you do an incremental dump next time, all changes after the snapshot was taken are backed up. A time gap between **ax\_clonefs** and **dump** causes a loss of some changes at the next incremental dump.
- ▲ Normally you do not need to supply a date because the current time at which **dump** starts is automatically added in /etc/dumpdates. Use the **T** option only when you use a script to dump a file system that was cloned by **ax\_clonefs** (for example, you might want to clone several partitions before dumping them to a tape).

In the script, create a timestamp using a date string in order to specify the time when **ax\_clonefs** finishes copying the file system. Include the timestamp along with the **T** 



option in the dump command. Doing so ensures that dump enters the proper date string in /etc/dumpdates for each partition.

The timestamp should use the following date string:

```
'date '+%a %h %d %T 19%y''
```

The following is an example of a **dump** command with the **T** option, where **date1** is the timestamp created in the script:

dump 0Tdsbfu "\$date1" 141000 11500 126 /dev/rast4c /dev/ad0c

Use the m option together with the name of the source partition in ax\_clonefs. This option forces **dump** to record in /etc/dumpdates the partition name you specified instead of the name of the partition being dumped (that is, the clone partition). Using the source partition name ensures that the next incremental **dump** finds the information from /etc/dumpdates about the source partition that has been backed up. The following is an example:

dump 0dsbmfu 65000 6700 126 /dev/rvp10 /dev/rast4 /dev/rvp11 In this example, the partition name specified with the **m** option is /dev/rvp10. It is recorded in /etc/dumpdates. The name of the clone partition, /dev/rvp11, is not recorded.

For more information on the dump command syntax, refer to "Using the Dump and Restore Commands" on page 8-2.



# Restoring a File With a Damaged Inode

This section describes how to restore a file when you suspect data is corrupted and fsck shows errors in an inode. The following is a sample **fsck** output:

```
# fsck /dev/ad9a
** /dev/rad9a
** Last Mounted on /export/root
** Phase 1 - Check Blocks and Sizes
BAD/DUP FILE I=26 OWNER=root MODE=644
SIZE=404 MTIME=Sep 18 16:13 1995
CLEAR? [yn] n
```

Before you replace the corrupted file with the backup copy, determine the filename associated with the incorrect inode from the fsck output. In the previous example, the inode number is 26. The following procedure restores the file associated with inode 26:

1. Use **ncheck** to determine which file is associated with the inode:

```
# ncheck -i 26 /dev/rad9a
/dev/rad9a:
        /.cshrc
```

In this example, the file /.cshrc is associated with inode 26.

2. Locate the backup tape that contains /.cshrc, and install it on the server. Then invoke the **restore** command in interactive mode and verbose mode. For example, if the backup tape is in slot 2, enter the following:

```
# restore imvf /dev/nrast2
```

The restore command displays files by name and inode.

3. Select the file to be restored, making sure the file is associated with the inode reported by fsck. In this example, the file with inode number 26 is /.cshrc, so enter the following to add the file to the list of files to restore:

```
restore> add .cshrc
.cshrc added
```

4. Enter the **extract** command, and specify the volume from which to extract the file:

```
restore> extract
Which volume? 1
./26 extracted
```

The restore program uses the inode number as the name of the file extracted. You can change the filename back to /.cshrc.



# Restoring a File from 1.5.1-produced Tapes

This section describes how to restore from 1.5.1-produced dump tape. Software Release 1.5.1 contains a bug that generates an additional End Of File (EOF) marker at the beginning of the tape and at the beginning of each file on the tape. This bug was corrected in later releases but can create a compatibility problem for 1.5.1-produced tapes.

To perform a restore of a particular file system on a Release 1.5.1-produced dump tape, advance the tape past the extra EOF mark at the beginning of each file, as well as past the normally occurring EOF at the end of each file. For example, to reach the third file on the tape, you must advance through five EOF markers with the following command:

#mt fsf 5

This command advances the tape beyond the extra EOF markers to start the restore from the correct position. At this point, you can invoke the restore command. For example:

#restore imvf /dev/nrast4c

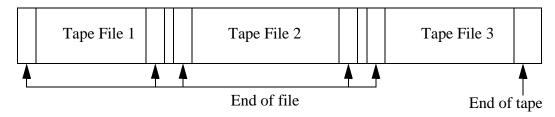


Figure 8-3. 1.5.1 tape encoding

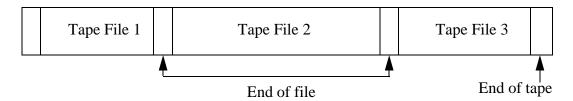


Figure 8-4. 1.9 tape encoding



# Common Tape Drive Errors

This section describes common errors you might encounter when using a tape drive to back up or restore a file system. The errors are categorized into the following types:

- ▲ Operator errors
- Media errors
- Tape drive errors

This section gives either a sample command that causes the error or describes the scenario in which a tape drive problem is detected. It also lists the error messages or symptoms and provides suggestions for fixing the problem.

# **Operator Errors**

This section discusses errors caused by the person using the tape.

## I/O Error

Command tar cvf /dev/rast2 /vmunix

Error messages ast2: no tape loaded or drive offline

tar: /dev/rast2: I/O error

Analysis Tape drive does not exist, or the tape is not loaded.

Solution Specify the correct tape drive, or load tape in the drive.

#### **Write Error**

Command tar cvf /dev/rast2 /vmunix

Error messages ast2: tape is write protected

tar: /dev/rast2: Permission denied

**Analysis** Tape is write-protected.

Solution Eject the tape, and open the write-protection slot.

#### Blank Tape Error

Command tar xvf /dev/rast2 /vmunix

Error messages ast2: error: sense key(0x8): blank check

vmunix: sense = f0 0 48 0 0 fc 0 12 0 0 0 0 0 0 0

0 0 0 1 20 0 0 22 fc 21

**Analysis** Tape is not formatted, or tape is formatted for the 8500 drive but

read in an 8200 drive.

Solution Use the tape on the correct tape drive, or use the correct tape on the

current drive.



## **Memory Error**

Command tar tvf /dev/rast13

Error messages tar: read error: Not enough memory

**Analysis** Data on the tape is not in **tar** format, or an incorrect block size

is specified.

Solution Load the correct tape that contains the **tar** file.

## **End-of-tape Error**

Command dd if=/dev/ad0e of=/dev/nrast3 bs=16384

Error messages dd: write: Error 0

vmunix: EOM on forward-spacing command

**Analysis** Use mt -f /dev/nrast3 status to see if it returns the

following message:

sense key(0x13) = EOT

If so, the tape has reached the end, and the tape drive is unable

to write.

Solution Use a higher-capacity tape.

# Tape Media Errors

This section discusses errors caused by bad spots on a tape. These errors are called Error Checking and Correcting (ECC) errors and are correctable. Tapes often contain unusable spots, which the tape drive simply skips over during read or write operations. If the number of ECC errors exceeds 10 per megabyte of data transferred, vmunix generates a warning message.

Tape media errors are harmless, but they slow down write operations. They also reduce the tape capacity. Whether or not you need to take corrective action when ECC errors occur depends on the number of errors indicated in the vmunix message. If fewer than 20 errors occur per megabyte of data transferred, you can ignore the error message. If more errors occur and other error messages appear, we recommend using a new tape or cleaning the drive heads. If you continue to use the tape, remember that the actual tape capacity is reduced because of ECC errors, and you probably need to change the backup script to ensure that the tape does not reach the end before the backup completes.



**Recommendation:** Because tape media errors are more common when the tape drive heads are dirty, clean the 8-mm drive heads once a month or every 30 hours of use, the 4-mm drive heads every 8 to 10 hours of use (once a week at most), and the DLT4000 when the Use Cleaning Tape LED is lit. For the 1/4-inch drives, clean the heads every 8 hours of use. Also, keep the tape drive door closed when not in use to keep out dust. Refer to the NetServer's hardware manual for more information on the type of cleaning cartridge for your drive.



#### Write Error

Command dump Oucbsdf 100 6700 65000 /dev/rast2 /dev/ad2c

Error messages vmunix: ast2: 22 recoverable errors per MB

transferred

vmunix: ast2: the tape may be wearing out or the head

may need cleaning

Analysis Write operations are retried because the drive needs to skip the bad

spots on the tape.

Solution If the number of errors per megabyte of data transferred exceeds

20, clean the drive heads or switch tape brands.

## **Read Error**

Command tar xvf /dev/rst2

Error messages vmunix: ast2: 14 recoverable errors per MB

transferred

vmunix: ast2: the tape may be wearing out or the head

may need cleaning

**Analysis** Read operations detect bad spots on the tape.

Solution Ignore the error message unless, because of the error, the drive

cannot read data from the tape.

#### Write Failure Error

Command dump Oucbsdf 100 6700 65000 /dev/rast2 /dev/ad2c

Error messages DUMP: Tape write error 5098 feet into tape 1

vmunix: ast2: file mark write failed

vmunix: ast2: error: sense key(0x3): media error vmunix: sense = f 0 0 43 0 0 0 1 12 0 0 0 0 0 0 0

0 0 11 0 0 0 23 1 21

The tape has a very large bad spot that it fails to skip. Sometimes Analysis

this occurs because the tape drive heads are dirty.

Solution Clean the heads and try again using the same tape. If the error

happens again, use a new tape.

#### **Tape Jam Error**

Tape cannot be ejected from the drive after you press the release Scenario

button on the drive.

**Analysis** Tape tension is incorrect, confusing the tape drive logic.

Solution Remove the tape drive and replace it in the drive slot to reset the

> tape drive logic. When the drive executes the self test, repeatedly press the release button until the tape is released. Do not reuse

the tape.



# **Tape Drive Errors**

This section discusses the errors caused by confused tape drive logic or malfunctioning drive read/write heads. In some cases, you need to replace the drive and send it to Auspex for repair.

#### Write Error

Command tar cvf /dev/rast2 /vmunix

Error messages ast0: fatal error d (SP data transfer error)

SPO: 1198: SP Data Transfer Error, slot 2, drive 1,

Sector 0

**Analysis** The tape drive cannot write to the tape. Use the mt -f /dev/nrast0

> **status** command to display status information about the drive. If the status is sense key(0x10) = fatal, the drive logic is confused. This happens usually when the drive is not properly seated in the drive rack or the drive hardware is damaged.

Solution Remove the tape drive, and replace it in the drive slot to reset the

> tape drive logic. After the drive executes the self test, repeat the write operation. If the error happens more than three times, the drive hardware probably is damaged, and you need to replace

the unit.

#### **Online Error**

Scenario The tape drive fails to return to the online status.

**Symptoms** The tape drive LED does not indicate that the drive is online.

(When online, the 8200 drive has only the right LED on, and the

8500 drive has only the top LED on.)

The mt -f /dev/nrastn status command returns sense key(0x0)

= no sense.

The drive logic is confused because the drive is not properly seated Analysis

in the drive rack or the drive hardware is damaged.

Solution Remove the tape drive, and replace it in the drive slot to reset the

> tape drive logic. After the drive executes the self test, repeat the write operation. If the error happens more than three times, the drive hardware probably is damaged, and you need to replace

the unit.



# 9 Measuring the NetServer's Performance

# **About This Chapter**

This chapter describes the Auspex Performance Monitor, which gathers performance statistics about your server. You can display the statistics in real time on the screen of any workstation connected to the server, or you can save the statistics and view them later. When playing back the statistics, you can display them in histograms or graphs for easy interpretation.

The following commands collect, display, and convert server statistics:

ax\_perfmon Collects server statistics, displays them in real time, or saves the data

to a file.

Displays statistics collected by **ax\_perfmon** in histograms or graphs. ax\_perfhist

A variety of processor names appear on screens generated by the Auspex Performance Monitor. You may not have all the processor boards mentioned in this chapter or on the screens. The following list describes some of the ax\_perfmon processor naming conventions:

▲ NP refers to a dual-CPU processor board. One CPU is for network processing, and one CPU is for file processing. With SBus adapters, each NP board can have three FDDI interfaces, three 100Base-T Ethernet interfaces, high-speed interfaces, or three ATM interfaces.



**Note:** ATM LANE is available as an optional product. Contact your authorized Auspex representative for more information.

*FP* refers to the File Processor on an NP.



# Capturing and Displaying Real Time **Performance Data**

This section describes ax\_perfmon, an Auspex Performance Monitor command for collecting and displaying NetServer performance data. From a single workstation, you can use **ax\_perfmon** to display performance data from several Auspex servers.

The concise display of NetServer performance data is a valuable tool for system management decisions, enabling detailed investigation of site- and application-specific NetServer performance.



**Note:** If you upgraded your NetServer software from any previous version, notice that ax\_perfmon rearranged data in some of its screen displays to accommodate hardware features introduced with this software release. The information you are looking for may be in a different location on the screen.

# How ax perfmon Works

ax\_perfmon gathers data from each processor in the system and provides both real-time display and fast-motion replays of significant NetServer performance events. The statistics gathered include:

- CPU utilization for each processor
- network and disk I/O rates and I/O types
- file system operation loads, mixes, and rates
- SP loads, write accelerator usage, and write operation rates
- cache usage and age distributions for network, file, and disk subsystems

ax\_perfmon gathers data over user-selected intervals and integrates the data over that interval. That is, if the interval is five minutes, ax\_perfmon measures total activity for each parameter for five minutes and then divides the results by 300 seconds to give the average rate for the parameter. Gathered this way, the data reflects average demand performance. To obtain data that reflects peak demand performance, select a very short interval (seconds or a fraction of a second).



The data displays in multiple screens, each containing different types of performancerelated data:

System summary	Summarizes CPU utilization for each processor, traffic over the network interfaces, LFS file system operations, data cache activity, and disk activity for disks supported by SP0, which are the drives in the NetServer base cabinet. If more than one SP is installed in your system, pressing the "S" key displays a System Summary screen with disk statistics from the next SP.
NP statistics	Displays CPU utilization information about the network processor on an NP. Also displays the statistics for frames received and sent by the network interfaces.
FP statistics	Displays CPU utilization for an FP, statistics on file system operations per second, and statistics on cache utilization. Cache utilization helps determine when to add more primary memory.
SP statistics	Displays the CPU utilization for each of the SPs, statistics for disk operations per second, and statistics for cache utilization.
Virtual partition statistics	Displays the number of I/O operations for all the virtual partitions on each SP. For virtual partitions composed of multiple members, the statistics indicate which member performed the operation.

To see a graphical representation of the data gathered by **ax\_perfmon**, use **ax\_perfhist**, which is another Performance Monitor command. For more information on displaying statistics in histograms, refer to "Displaying Performance Data in Histograms" on page 9-18. The following provides an outline of the procedure for displaying performance data in histograms:

1. Store the **ax\_perfmon** statistics in a file, as shown:

```
ax_perfmon -o outfile
```

2. Execute the ax perfhist command, which uses the file saved in step 1 as input. The following is an example:

```
ax_perfhist -df outfile
```

The next subsections describe the following:

- How to start and control the Performance Monitor
- Various Performance Monitor screens (each section includes a sample screen captured from a NetServer and a description of the screen)
- How to analyze Performance Monitor data

# Starting the Performance Monitor

To start collecting server performance data, execute **ax\_perfmon**(8). The syntax of the **ax\_perfmon** command is shown as follows:

```
ax_perfmon [-s] [-o file [-f filterfile]] [-i file ] [-p period] [-t
time]
```

The optional arguments perform the following functions:

-S	Executes <b>ax_perfmon</b> in silent mode. Does not display data as it is gathered. (This option cannot be used with the - <b>i</b> option.)
-o file	Stores gathered data in a specified output file. The data in this file can be displayed using the -i argument. This option is required to save the

statistics for display later using the **ax\_perfhist** command, which organizes the statistics in histogram format for easy viewing. For further information on **ax\_perfhist**, refer to "Displaying Performance Data in Histograms" on page 9-18.



**Note:** The - $\mathbf{o}$  option uses a lot of space. A file can grow to a few megabytes in size in as little as 30 seconds if the update period (see the - $\mathbf{p}$  option) is left at the one second default. Be sure that the partition receiving the output file has adequate space and use the - $\mathbf{p}$  option to control the number of updates that go to the output file.

-f filterfile

Specifies the name of the file containing a list of variable names, which determine the types of performance data (for example, statistics concerning network interfaces) to save in the output file. Specify the filter file only if you are writing performance statistics out to a file using the -o option. Saving statistics only for items that interest you reduces the size of statistics files. For more information on creating a filter file, refer to "Using a Filter File with ax\_perfmon" on page 9-16.

-i file

Displays data from the specified input file instead of gathering it from an active system.

-p period

Specifies the interval in seconds between screen updates (default is one second).

If used in combination with the -i argument, the -p argument controls the screen update rate but does not change the rate at which data was originally gathered. In this way, you can replay saved data quickly.

If used without the -i argument, the -p argument controls the interval between data retrievals.

-t time

Specifies the duration in seconds that **ax\_perfmon** runs. Without the **-t** option, **ax\_perfmon** runs continuously either on the screen or to a file.

Some of the most commonly used **ax\_perfmon** commands are listed as follows:

▲ To display real-time data about the local host at one-second intervals:

ax\_perfmon

▲ To display real-time data about host *netserver1*:

rlogin netserver1; ax\_perfmon

▲ To silently gather data about the local host at five-minute intervals and store the data in the file /sysadm/perf.data:

ax\_perfmon -s -p 300 -o /sysadm/perf.data >& /tmp/myerrfile &
The file myerrfile stores any error output messages gathered by ax\_perfmon.

▲ To play back the data collected in the previous example:

```
ax_perfmon -i /sysadm/perf.data
```



**Note:** Although the data was gathered at five-minute intervals, the screens display at the default one-second intervals, thus speeding up data playback.



## Interactive Performance Monitor Commands

After starting the Performance Monitor, use the following single-character commands to control the display. Remember that the commands are case-sensitive.

- Displays the **ax\_perfmon** help screen.
- n Displays the next data screen. For example, if the current display is System Summary, the next screen is NP Statistics.
- Displays the previous data screen. p
- ^L Refreshes the display.
- Quits ax\_perfmon. q
- Pauses the currently displayed screen (useful when replaying saved data).
- Continues displaying updated screens (used only after a pause command).
- Does one screen update in pause mode. >
- ſ Speeds up the display by reducing the interval between screen updates by 1/10 of one second.
- 1 Slows down the display by increasing the interval between screen updates by 1/10 of one second.
- { Speeds up the display by reducing the interval between screen updates by one second.
- } Slows down the display by increasing the interval between screen updates by one second.
- On an NP with more than eight net interfaces, the NP screen initially displays C information for the first group of interfaces. This command steps through information for the remaining SBus interfaces.
- f On a system with more than one FP, the FP Statistics screen initially display information for the first FP (FP0) in the system. This command displays information for subsequent FPs.
- On a system having more than one SP, the System Summary screen, the SP S Statistics screen, and the Virtual Partition Statistics screen initially display information for the first SP (SP0) in the system. This command displays information for subsequent SPs. From the System Summary screen, it also displays disk activity statistics for disks on the subsequent SP.
- Displays information for the next group of virtual partitions. v
- N On a system having more than one NP, the NP Statistics screen, Summary Screen, and NP Protocol Stats screen initially display network information for the first NP. This command displays information for the subsequent NP. From the System Summary screen, it also displays disk activity statistics for subsequent NPs.

The **s** and **N** commands have no effects on systems with only one SP or NP.



# **Interpreting Statistics Screens**

This section describes the fields in each Perfomance Monitor screen.

#### Screen Header

At the top of each Performance Monitor screen is a screen header similar to the following:

```
1.0 System Summary(NS7000-500)
                                          host1
                                                           Thu Aug 21
14:03:50 1997
```

The / is the spinner that moves each time the screen is updated. The number following the spinner is the time interval between screen updates. In the previous example, the screen is updated every second.

The title of the screen follows the time interval. The title can be one of the following:

- System Summary
- NP Stats
- FP Stats
- SP Stats
- **Virtual Partition Stats**

The header also includes the name of the host from which statistics are gathered and the current date and time. If you are playing back statistics, the host name and time information show where and when the data were collected.

The header for the System Summary screen shows additional information on the server type. The server type, which is displayed before the host name, can be one of the following:

- NS7000-200 for an NS 7000/200 Series NetServer
- NS7000-600 for an NS 7000/600 Series NetServer IS THIS SUPPORTED??
- NS7000-700 for an NS 7000/700 Series NetServer
- NS7000-800 for an NS 7000/800 Series NetServer

The following sections include screen examples generated by different server types.

# System Summary Screen

The format of the System Summary screen is different among server types. Figure 9-1 is a sample System Summary screen for the NS 7000/200 Series NetServer. Figure 9-2 is a sample System Summary screen for the NS 7000/600 Series NetServer. The NS 7000/800, NS 7000/700, NS 7000/500, and NS 6000 System Summary screens are similar to the NS 7000/600. The statistics on the screens are described after Figure 9-2.



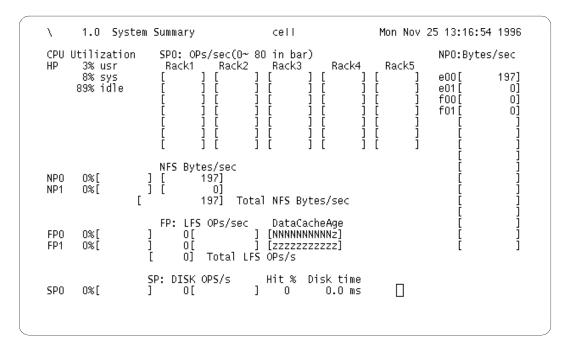


Figure 9-1. Sample System Summary screen for an NS 7000/200 Series NetServer

CPU NP0 NP1 NP2 NP3 NP4	1.0 Util 0%[ 0%[ 1%[ 0%[ 0%[	System Sum   1     1     1     1     1	nmary NFS Bytes/s [ 985] [ 268] [ 125] [ 346]			Util	Thu N NPO:Byte a00[ f00[ H00[ [		SP0 (0*	:58 1996 :0Ps/sec 80 bar)
		Г	179/1	Total	l Bytes/	000	Г	1	Г	1
		L	17241	iota	i bytes/	260	Ĺ	1	Ĺ	j
		L	LFS OPs/sec		DataCa	cheAge	[	]	Ε	]
FP0	0%[	1	0[	1	[777777		ī	ī	R2Ē	ī
FP1	0%[	í	ο̈́C	1	[88888]		ř	1	.,	i
FP2	0%[	1	30	1			Ė	1	ř	1
FP3	0%[	1		1			L r	1	L F	1
		1	0[	1	[zzzzzz]		Ĺ	1	L	1
FP4	0%[	]	0[	]		AAAAA]		]		]
		[	0] Tota	l LFS	OPs/s		[	]	[	]
							[	]	[	]
		9	SP: DISK OP:	5/s	Hit % D	isk Time	· [	]	R3E	]
SP0	4%E	]	0[	]	0	0.0 ms	t		[	]
SP1	0%[	1	30	]	0	0.0			[	j
SP2	0%[	ĩ	ο̈́C	ī	Ò	0.0			Ī	ĩ
SP3	ŎΫĒ	ĩ	ŏξ	í	ŏ	ŏ.ŏ			ř	ĩ
SP4	0%[	i	ŏΕ	1	ň	0.0			ř	i
31 4	V% L	ť		ים דת נ	K OPs/s	V+V			ř	1=
		L	vi lota	T DI2	V 0L2/2				L	1

Figure 9-2. Sample System Summary screen for an NS 7000/600 Series NetServer

The System Summary screen contains the following information:

The first column describes CPU utilization for NPs and SPs. CPU utilization is expressed both in percentage and in a bar graph made up of equal signs (=). If a dash (-) is displayed instead of a percentage, the processor is not installed.



**Note:** The NP's CPU statistics pertain to the network processor only, even though each NP board has two CPUs. Utilization statistics for the file processor CPU, also located on the NP, are listed by FPs.



- The upper center statistics show HP utilization. The utilization statistics for the HP, which show how busy the HP CPU is, are divided into three types:
  - User utilization (represented as usr on the screen) shows the percentage of CPU time in user mode.
  - System utilization (represented as sys on the screen) shows the percentage of CPU time used in system mode.
  - Idle represents the percentage of time when the CPU is neither in user nor system mode.

User utilization, system utilization, and idle time add up to 100 percent.

The rest of the screen displays statistics on operations processed by network interfaces, FPs, SPs, and statistics on disk activity.

For the network interfaces, the number of bytes processed by each network interface per second is calculated. See the "NBUF stats" row of Table 9-1 for an explanation of the interface definitions that also show up here on the summary screen. An N entered with the summary screen moves the NP portion of the screen to the next NP board. All interface numbering restarts from 0 on each board.

For the FP, two types of information are shown:

- The number of LFS operations per second In the bar, different characters represent different NFS operations; refer to the FP Statistics screen for definitions of each character.
- The ages of data cache buffers associated with the FPs

The letters in the bar represent the time since the buffers were used. "A" represents buffers used within the last minute; "B" represents buffers used within the last two minutes. "Z" represents buffers that have not been used for more than 26 minutes. Lowercase letters have no significance to most users.

For the SP, the display shows the number of operations per second on each device controlled by the SP. The devices are arranged by drive racks, each of which corresponds to a bar graph. Within each bar graph, "R" represents a read operation, and "W" represents a write operation.



**Note:** The rack numbers are relative to the SP being displayed, meaning that rack numbering always starts from 1.

Refer to your NetServer hardware manual for a detailed description of the drive racks supported by an SP.

#### **NP Statistics Screen**

The NP Statistics screen displays network statistics based on data collected from an NP. Figure 9-3 shows a sample NP Statistics screen.



/ 1.0 NP Stats		wi Total Byt	zard es/sec			Thu No	ov 21	13:52	:04 1	996
CPU: 0% [ MBUF stats( max/cur/	]	Total Bgt	41		work	k/que/b	ousy:	48/	0/	0
0: 0/ 0/ 0 f: 64/ 0/ 0 a: 64/ 32/ 0	1: 0/ E: 0/ a: 128/	0/ 0 0/ 0 44/ 0	2: 0: 9:	0/ 0/ 0/	0/ 0/ 0/	0 0 0	3: T: H:	0/ 0/ 0/	0/ 0/ 0/	0 0 0
ITEMS	f00 0 0 0 0 0 0 0 0 0 0 0 0 0 0	H00 0 0 0 0 0 0								

Figure 9-3. Sample of an NP Statistics screen

The upper portion of the screen contains statistics about the NP and buffering information for packets going in and out of network interfaces. See the "MBUF stats" row in Table 9-1 for an explanation of the MBUF stats interface definitions.

The NP processes frames from the networks. The lower portion of the screen is a table listing statistics for each network interface. Some statistics apply to more than one interface type, and some apply to only one interface type. Interface definitions for the "ITEMS" row are the same as for the MBUF stats.

For information on the next group of SBus interfaces, type c. For information on subsequent NPs, type N.

Table 9-1 explains the network statistics.

Table 9-1. NP statistics

Statistics	Explanation	
CPU Util	Utilization of the CPU for network processing.	
Nfs work/que/busy	Number of NFS workers running on the NP; number of NFS jobs queued, wai for free NFS workers to process; number of busy NFS workers. An NFS wor is an NFS server daemon servicing remote procedure calls from clients. Mult	
MBUF stats	Statistics for MBUF interfaces. <b>0</b> , <b>1</b> , <b>2</b> , and <b>3</b> are for BUFE slots on the NP. <b>f</b> is for FDDI. <b>E</b> is for half-duplex 100Base-T Ethernet. <b>O</b> is for offboard buffers taken from another NP. <b>T</b> is for TCP. <b>a</b> is for ATM (large buffers and small buffers). <b>H</b> is for full-duplex 100Base-T Ethernet.	
Frames	Number of frames received and sent by the Ethernet interface.	
Bytes	Number of bytes received and sent by the Ethernet interface.	
IBytes	Number of bytes received by the Ethernet interface.	
Obytes	Number of bytes sent by the Ethernet interface.	

Table 9-1. NP statistics (Continued)

Statistics	Explanation
InErr	Number of frame errors received by the Ethernet interface.
OutErr	Number of frame errors sent by the Ethernet interface.
Collis	Number of Ethernet collisions on the Ethernet interface.
OutDisd	Number of output packets discarded by the Ethernet interface.
InBig	Number of oversized frames received by the FDDI interface.
InUndrsz	Number of undersized frames received by the FDDI interface.
InInval	Number of frames with invalid data received by the FDDI interface.
InAbort	Number of frames received by the FDDI interface that are incomplete and cannot be processed.
InDrop	Number of frames received by the FDDI interface that are discarded because the receive buffer was full.
MTU	Maximum transmission unit size used by the server. It defaults to 4352.
Offnet MTU	Maximum transmission unit size used by the server for FDDI traffic sent to networks not directly attached to the server. It defaults to 4352.

#### **NP Protocol Screen**

The NP Protocol Statistics screen displays network statistics by protocol. Figure 9-4 shows a sample NP Protocol Statistics screen.

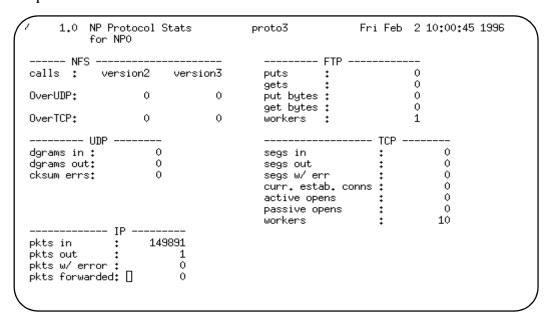


Figure 9-4. NP Protocol Statistics screen

Each protocol has its own section. Note that the NFS section also shows statistics by version2 and version3. Table 9-2 explains the network protocol statistics.



**Table 9-2. NP Protocol Statistics** 

Statistics	Explanation
NFS statistics	
calls	Version2 or version3 NFS calls.
OverUDP	Number of NFS calls over UDP.
OverTCP	Number of NFS calls over TCP.
UDP statistics	
dgrams in	Number of datagrams received.
dgrams out	Number of datagrams sent.
cksum errs	Number of checksum errors.
IP statistics	
pkts in	Number of packets received.
pkts out	Number of packets sent.
pkts w/error	Number of packets received with errors.
pkts forwarded	Number of packets forwarded.
FTP statistics	
puts	Number of puts.
gets	Number of gets.
put bytes	Number of bytes in puts.
get bytes	Number of bytes in gets.
workers	Number of FTP workers configured.
TCP statistics	
segs in	Number of segments received.
segs out	Number of segments sent.
segs w/ err	Number of segments received with errors.
curr. estab.conns	Number of connections currently established.
active opens	Number of active opens.
passive opens	Number of passive opens.
workers	Number of TCP workers configured.



#### **FP Statistics Screen**

The FP Statistics screen displays statistics for one FP. Figure 9-5 shows a sample FP Statistics screen.

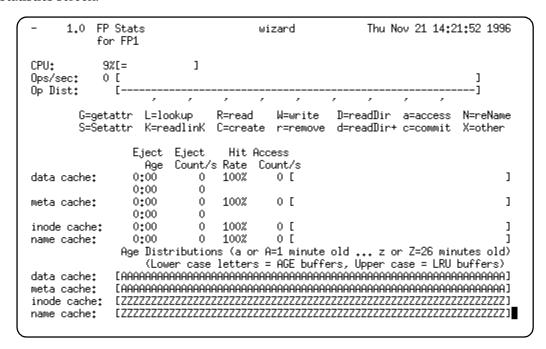


Figure 9-5. Sample FP Statistics screen

For information on subsequent FPs, type **f**.

The screen contains the following information:

#### ▲ FP activity

The first line of this area displays the CPU utilization of the current FP. The second line is an operations-per-second bar, in which letters represent kinds of operations.

The third line shows the distribution of the operations; it indicates the percentage of total operations that each operation type represents. The types of operations are listed below the operations-distribution bar.

#### ▲ Cache activity

- All cache memory is on NP boards. Four types of caches are displayed:
- Data cache is the user-data disk blocks in primary memory.
- Meta cache is the nonuser-data disk blocks in FP memory, which store information about file systems.
- Inode cache is the inodes cached in FP memory.
- Name cache is the file name information cached in FP memory.

For each cache, the following information is displayed:

- Eject age, which is the average age of all items ejected from the cache since the last screen update.
- Eject count, which is the number of items ejected per second from the cache.



- Hit access rate shows the percentage of cache accesses for items already in the cache.
- Hit access count is the number of times per second the cache was accessed. The bar following the hit access count shows the total number of cache accesses; "=" indicates a hit and ":" a miss.

For the data and meta caches, the first line shows LRU buffer statistics, and the second line shows AGE buffer statistics. The LRU buffers are the buffers that are more likely to be used soon. They are only ejected from the cache when no AGE buffers are left. AGE buffers contain data considered less likely to be referenced again than the data in LRU buffers.

Age distribution of items in the caches.

An "A" represents items that have been accessed within the past minute, "B" represents items that have been accessed within the last two minutes, and so on. "Z" represents items that have not been accessed for 26 minutes or more. Uppercase letters represent LRU buffers; lowercase letters represent AGE buffers.

#### **SP Statistics Screen**

The SP Statistics screen displays statistics for one SP. Figure 9-6 shows a sample SP Statistics screen.

	wizard Thu Nov 21 14:26:26 1996
0% [ 0	] Resp time 0.0 [] ] ] Service time 0.0 ms
] 0	] Queue time 0.0 ms
] 0	]
] 0	] NVRAM size MB
1 15: 0	1 29; 0 [ ]
1 16: 0	1 30; 0 [ ]
1 17: 0	1 31; 0 [ ]
1 19: 0	1 33: 0 [ ]
1 20: 0	1 34: 0 [ ]
1 22: 0	1 36: 0 [ ]
1 23: 0	1 37: 0 [ ]
1 25: 0	1 39; 0 [ 1
1 26: 0	1 40; 0 [ 1
	0 [ 0% [ 0

Figure 9-6. Sample SP Statistics screen

For all NetServers except the NS 7000/200, type s for information on subsequent SPs. (The NS 7000/200 supports only one SP.) The screen contains the following information:

- CPU utilization.
- Total response time, which is the combined times of the Service time and the Queue time.



- ▲ Number of operations per second. In the bar following the number, "R" represents a read operation and "W" represents a write operation.
- ▲ Write cache hit percentage is the sum of overrides and coalesced writes as a percentage of the total number of writes involving the write cache. It measures how well the write cache is used in avoiding write operations.
- ▲ Service time is the time taken in writing to disk.
- ▲ Overrides per second indicates how well the write cache eliminates redundant writes (of inode and indirect blocks). An override is a write that is not performed because a later write to the same location hit a request currently in the write cache.
- ▲ Queue time is the time spent waiting in the queue.
- ▲ Coalesce per second measures the number of write operations that the write cache coalesces. Coalescing data improves efficiency because data destined for adjacent disk areas are written with one write operation.
- ▲ NV bufs is zero unless a Write Cache board is present.
- ▲ NVRAM size shows the size of the write cache.
- ▲ Disk activity. For each disk controlled by the current SP, the number of I/O operations is displayed. In the bar following the number, "R" represents a read operation and "W" represents a write operation.

On this screen, the disks are numbered relative to the current SP. Disks controlled by an SP V are numbered 0 to 41.



**Note:** The NS 7000/200, which supports the SP IV or SP V, supports a maximum of 35 drives.

For a detailed description of write caching, refer to Chapter 7.



#### Virtual Partition Statistics Screen

The Virtual Partition Statistics screen displays statistics for virtual partitions controlled by the current SP. Figure 9-7 shows a sample Virtual Partition Statistics screen.

/		Partition				Thu	Nov	21	16:03:21	1996
256: 257: 258: 259: 260: 261: 262:	on SP1 0 [ 0 [ 0 [ 0 [ 0 [ 0 [ 0 [	] ] ] ]	56 VP' 278: 279: 280: 281: 282: 283: 284:	's	] ] ] ] ]	300: 301: 302: 303: 304: 305: 306:	0 0 0 0 0	[ [ [ [		] ] ] ] ]
263: 264: 265: 266: 267: 268: 269:	0 [ 0 [ 0 [ 0 [	] ] ] ] ]	285: 286: 287: 288: 288: 289: 290:	0 [ 0 [ 0 [ 0 [ 0 [ 0 [	] ] ] ]	307: 308: 309: 310: 311: 312: 313:	0 0 0 0 0			) ) ) )
270: 271: 272: 273: 274: 275: 276: 277:	0 C 0 C 0 C 0 C 0 C 0 C	] ] ] ] ]	292: 293: 294: 295: 296: 297: 298: 299:	0 [ 0 [ 0 [ 0 [ 0 [ 0 [ 0 [	] ] ] ] ] ]	314: 315: 316: 317: 318: 319:	0 0 0 0 0	[		] ] ] ] ]

Figure 9-7. Example of a Virtual Partition Statistics screen

Each SP supports up to 256 virtual partitions. Each SP board has four virtual partition screens. The first screen displays statistics for SP0 partitions vp0-vp63.

For information on subsequent virtual partitions on the same SP, type v. The next screen for the same SP displays statistics for vp64-vp127. The third screen displays statistics for vp128-vp191. The final screen displays statistics for vp192-vp255.

For information on subsequent SPs, type s. The partition numbers for each SP match the partition numbers set up in the /etc/vpartab file, so the first number on the SP1 display is 256. Figure 9-7 shows an example of the first screen for SP1. The system supports up to five SP boards.

The number of I/O operations follows the partition number. The operations bar shows which member of the virtual partition actually did the operation. A "0" represents I/O going to the first member; "1" represents I/O to the second member, and so on. Each write to a mirrored virtual partition generates a write to each member, so the I/O operations count may be higher than expected. A read to a mirrored virtual partition is satisfied by a read to just one member.



# Using a Filter File with ax\_perfmon

The **ax\_perfmon** command syntax allows you to specify a filter file listing performance parameters about which you want to save statistics. Use a filter file if you want to:

▲ keep the output file small:

The variables in the filter file define the types of statistics stored in the output file. For example, if you are interested only in statistics about Ethernet interfaces, create a filter file that filters out all **ax\_perfmon** data not related to Ethernet networks.

**▲** convert performance data to ASCII format:

Without the filter file, the **ax\_perfmon** output file contains ASCII data of decimal numbers in an undocumented format. It cannot be used as input for databases or spreadsheet software. Specifying a filter file in with **ax\_perfmon** automatically converts all the statistics to decimal numbers in an easily understood format.



**Note:** In pre-1.6 releases, **ax\_perfilter** converted performance data to ASCII format. In version 1.6 and later, the filter file specified with **ax\_perfmon** eliminates the need for a separate command for converting data formats. As a result, **ax\_perfilter** is removed in Auspex software Version 1.6 and later.

The following sections describe how to create a filter file and provide a sample output file.

# Creating a Filter File

These sample filter files are provided in the /usr/auspex directory:

- **▲** summary\_filter (for summary statistics)
- ▲ np\_filter (for NP statistics)
- ▲ fp filter (for FP statistics)
- ▲ sp\_filter (for SP statistics)
- ▲ vpar\_filter (for virtual partition statistics)

Use a text editor to view each of these sample files to see if they fit your needs. You can create your own filter files by deleting or adding parameters from an existing filter file. For example, if you want to capture data about both SP and NP statistics, you can create a new filter file that concatenates *sp\_filter* and *np\_filter*.

"Types of Statistics Displayed by ax\_perfhist" on page 9-40 lists the filter file parameters and definitions for ax\_perfhist and ax\_perfmon. Look at the file /usr/auspex/screens.std to see which filter file parameter displays in each field of the statistics screen.

# Specifying a Filter File on the ax\_perfmon Command Line

The following example shows an **ax\_perfmon** command with a filter file named *np0stats* and an output file named *out*:

```
ax_perfmon -o out -f np0stats
```

In this example, the command creates an output file named *out*, which contains statistics related to NP0 in decimal format. You can use this file as input for **ax\_perfhist**, in which



case you can only display histograms for the parameters defined in the filter file. For more information on **ax\_perfhist**, refer to the next section.



**Note:** If the filter file in an **ax\_perfmon** command contains parameters not applicable to the configuration of your server, ax perfmon displays a list of messages indicating some parameters are not found before displaying the summary screen. Disregard these messages because ax\_perfmon can still successfully save data pertaining to the applicable parameters to the output file.

Figure 9-8 shows a sample output file created by **ax\_perfmon** with a filter file.

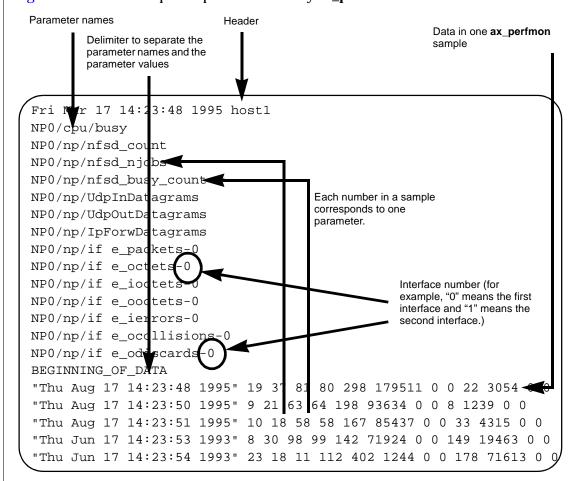


Figure 9-8. Sample output file created by ax\_perfmon with a filter file

The first line of the output file is the header equivalent to the header on an **ax perfmon** screen. Following the header is a list of parameter names. Refer to "Types of Statistics Displayed by ax\_perfhist" on page 9-40 for more information on parameters.

BEGINNING\_OF\_DATA serves as a delimiter separating the list of parameter names and the actual parameter values. Following the delimiter are the parameter values that are in decimal format. Each line contains one ax perfmon sample and the date and time the sample was collected. Each number in a sample has a one-to-one correspondence with the parameter names.



**Note:** An output file containing filtered statistics cannot be used as an input file to the **ax\_perfmon** -**i** *file* command.



# **Displaying Performance Data in Histograms**

To display performance data collected by **ax\_perfmon** in histograms, execute the **ax\_perfhist** command. This section provides the following types of information about **ax\_perfhist**:

- ▲ Overview of ax\_perfhist
- ▲ How to start **ax\_perfhist** and load statistics using the default settings
- ▲ How to configure the histograms and windows to suit your needs
- ▲ Information on parameters that control the types of statistics displayed in the histograms
- ▲ Suggestions on how to use **ax\_perfhist** efficiently

#### Overview

The **ax\_perfhist** tool presents server statistics collected by **ax\_perfmon** in an easy-to-read format. **ax\_perfhist** offers these advantages:

- ▲ The statistics are shown in histograms, which are easier to read than the characteroriented **ax\_perfmon** displays. For example, at a glance, you can identify the time periods during which the HP CPU load exceeded a predefined threshold.
- ▲ You can see the trend of server usage when playing back statistics using **ax\_perfhist**. Because the **ax\_perfmon** screen is updated at each time interval (one second by default), it may be difficult to determine, for example, the peak of CPU utilization for a particular process board.
- ax\_perfhist allows you to compare statistics on different processor boards. For example, you can display two histograms simultaneously for the CPU load—one for SP0 and one for SP1. You can easily tell which board is busier during a given period of time. ax\_perfhist can display up to eight histograms at once. You can also combine statistics gathered from different processor boards.
- ▲ You can customize the **ax\_perfhist** windows to display the statistics that interest you most, and filter out statistics that are irrelevant. Also, you can display only those statistics that exceed a user-defined threshold, zoom in on a few samples gathered in a short period of time, change the scales used on the horizontal and vertical axes, and so on.
- ▲ ax\_perfhist displays statistics omitted in the ax\_perfmon screens because of screen limitations.
- ▲ You can print statistical reports from **ax\_perfhist** with a mouse click.



**Note:** Because of changes in hardware components, **ax\_perfhist** may not be able to display some data collected with pre-1.9 software.



# Before Invoking ax\_perfhist

The ax\_perfhist command is useful only after you have used ax\_perfmon to save server statistics to a file. Use the following command to collect and save server statistics:

```
ax_perfmon -o filename
```

Alternatively, use the -s option to run ax\_perfmon in silent mode, in which ax\_perfmon collects statistics without displaying them.



**Note:** Use a consistent naming convention when saving ax\_perfmon statistics to a file. We recommend you end a filename with a period followed by a constant text string (for example, *server1.dat*). When choosing the input file for ax\_perfhist, specify listing only filenames with that particular text string suffix (for example, ".dat"). Refer to "Loading Statistics From a File" on page 9-21 for more information on selecting a file within **ax\_perfhist**.

You can run **ax\_perfhist** interactively or non-interactively. The non-interactive mode is useful mainly when you include ax\_perfhist in a script file.

# Starting ax perfhist Interactively

Because ax\_perfhist displays statistics graphically, you can only invoke it on an X terminal or a workstation running OpenWindows or X11/Motif. It does not work on an ASCII terminal. All the sample screens in this guide are from the OpenWindows environment. The appearance of the windows may be slightly different with X11/Motif.

To view the histograms on an X terminal or workstation on the network, first set the DISPLAY variable at the shell prompt as follows:

```
setenv DISPLAY displayname:0
```

where *displayname* is the name of your display.

To invoke ax\_perfhist on a host other than the one you are logged in to, enter the following command to allow the remote host to display histograms on your display:

```
xhost +hostname
```

where *hostname* is the name of the NetServer on which you start **ax\_perfhist**.

To display server statistics in histograms, simply enter ax\_perfhist without any arguments. An empty graph is displayed. The window shown in Figure 9-9 is the main window for viewing statistics. (For a description of the full syntax of ax perfhist, refer to "Starting ax\_perfhist with Options and Arguments" on page 9-46.)

When using ax perfhist, the mouse button functions are consistent with those in your windows environment. For example, to activate a command on the screen, press the left mouse button; to see a pull-down menu, hold down the right mouse button while pointing to a command. This guide assumes that you know how to close, open, move, and resize windows.

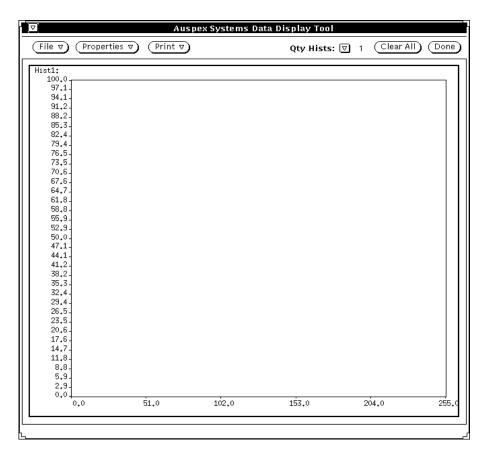


Figure 9-9. Data Display Window

Figure 9-10 shows the top portion of the Data Display Window and describes the general functions of the buttons used in the window. Detailed information about the windows opened by these buttons is provided in the sections following the figure.

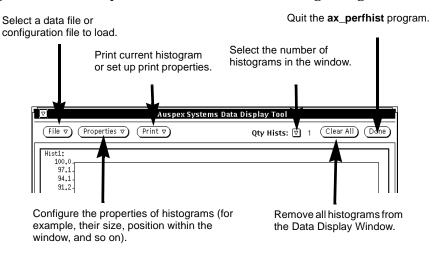


Figure 9-10. Buttons in the Data Display Window



## Loading Statistics From a File

The graph displayed by ax perfhist contains no statistics (see Figure 9-9) until you load the data from a file saved by **ax\_perfmon**. This section describes the histograms that are displayed using the default window properties; alternatively, you can configure the window properties before loading the statistics.

Follow these steps to load the statistics:

1. Click on the File button to display a pull-down menu. Select "Data Files..." from the menu. The File Load Window appears, as shown in Figure 9-11.

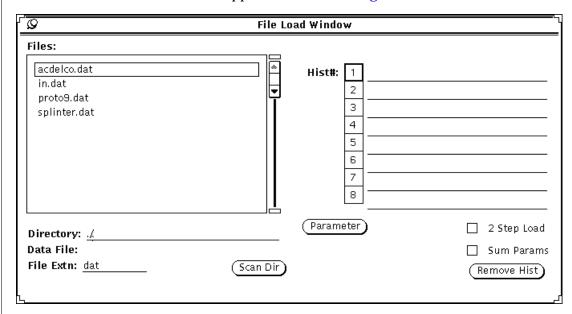


Figure 9-11. File Load Window

2. In the Directory field, type the name of the directory containing the statistics files. The If the File Extn field is empty, clicking on the Scan Dir button displays all the filenames in the directory. If you name your statistics files consistently, ending the filenames with a period followed by a constant string such as "dat," ax\_perfhist displays only the files with the specified suffix when you click on Scan Dir. (Another way to start scanning the directory is to press Return after typing the filename extension.)

In Figure 9-11, the file list displays four filenames. With a long list, use the scroll bar to move up and down the list.

- 3. Click on the file name you want to load. The filename appears in the Data File field.
- 4. Each data file contains more information than one histogram can show. You must select one type of statistics to be displayed in a histogram. To select a parameter, hold down the right mouse button when pointing at the Parameter button to display a pull-down menu that lists the statistics types.

The pull-down menu lists all the processor boards. Each board has two layers of submenus. For further information on selecting parameters, refer to "Types of Statistics Displayed by ax\_perfhist" on page 9-40.





**Note:** If the data file contains only filtered statistics, you can choose only from the parameters included in the data file. The pull-down Parameters menu displays the parameters without being categorized by boards.

By default, the statistics you selected appear immediately in the Data Display Window. If you do not want the statistics to appear immediately and prefer to display them after all the histograms have been defined, follow the procedure in "Loading Statistics Using 2 Step Load" on page 9-24.



**Recommendation:** Displaying both the Data Display Window and File Load Window allows you to select a different type of statistics quickly. Once you select an input file, you can send the File Load Window to the back, allowing only the pull-down menu buttons to show (as illustrated in Figure 9-12). When you activate any of these buttons to select a new parameter, the File Load Window does not obscure the Data Display Window because it remains in the back, and you can view the new statistics immediately.

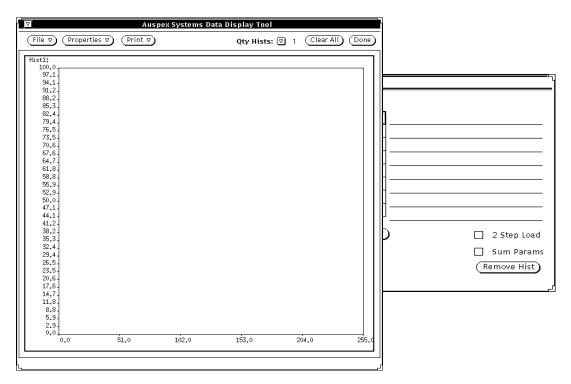


Figure 9-12. Displaying the Data Display and File Load windows simultaneously

## Clearing a Histogram

To clear the histogram, click on the Remove Hist button in the lower-right corner of the File Load Window. This button clears the statistics from the Data Display Window only. It does not delete the statistics file from your directory.



## **Displaying Multiple Histograms**

To display up to eight histograms simultaneously, first specify the histogram number, then specify the type of statistics in each histogram.

The following example shows how to display a second histogram in the Data Display Window:

- 1. Use one of the following ways to display multiple histograms:
  - In the File Load Window, click on 2 in the Hist# list.
  - In the Data Display Window, click on the Qty Hists button and select 2.
- To define the contents of histogram number 2, hold down the right mouse button while pointing to the Parameter button in the File Load Window. Select a parameter from the pull-down menu. The statistics appear in the second histogram in the Data Display Window, as shown in Figure 9-13. In this example, the first histogram shows the CPU load of the FP0; the second histogram shows the I/O byte counts on NP0.

The ax\_perfhist program automatically arranges and resizes the histograms to optimize the use of the Data Display Window. For example, if three histograms are displayed, the window is divided into three equal portions. If you delete two of the histograms, the remaining histogram is resized to occupy the entire Data Display Window.

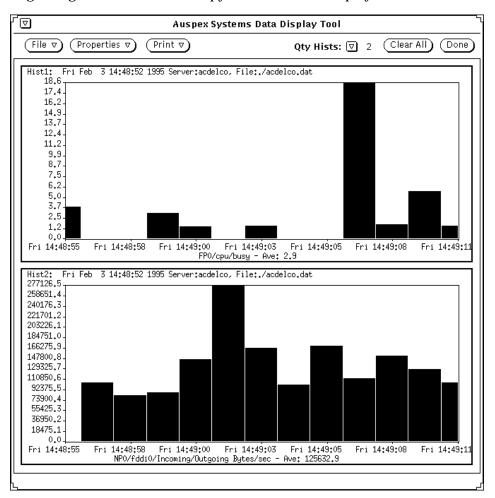


Figure 9-13. Data Display Window showing two histograms simultaneously



## **Loading Statistics Using 2 Step Load**

If you prefer to define the contents of all histograms first and load the statistics all at once later, click on the 2 Step Load box in the File Load Window. If this box is checked, follow these steps to load statistics to the histograms:

- 1. For each histogram, follow all the instructions in "Loading Statistics From a File" on page 9-21 to select an input file, a disk or virtual partition number (if applicable), and a parameter. Notice that no statistics appear in the histogram after you select a parameter.
- 2. In the File Load Window, use the right mouse button to click on the Load Hist button. In the pull-down menu, select Load Hist to load statistics to the current histogram or Load All Hists to load statistics to all histograms.

The 2-step load procedure is particularly useful if you have a large statistics file and you want to display multiple histograms. It eliminates the need for waiting for each histogram to display before you can specify the parameter and properties for the next histogram.

## Eliminating a Histogram from the Data Display Window

To display fewer histograms, remove the current histograms one by one. For example, if the Data Display Window contains four histograms, follow these steps to change the window to display only histogram number 3:

- 1. In the File Load Window, click on the number of the histogram that you want to remove. In this example, click on 1.
- 2. In the File Load Window, click on the Remove Hist button to remove the selected histogram.
- 3. Repeat the previous steps to remove other histograms.

## **Defining the Appearance of a Histogram**

This section describes how to define the appearance of an individual histogram and control the overall properties of the histograms in the Data Display Window.

## Configuring the properties of a histogram

You can customize a histogram by changing its size, scale, title, and so on. Although a single Data Display Window contains multiple histograms, each histogram can assume different characteristics.

To define the appearance of a histogram, click on the Properties button to display the pull-down menu. Select "Each Histogram..." in the menu. The Histogram Properties Window is displayed (see Figure 9-14). In this window, specify:

- ▲ which histogram to modify
- ▲ the values for a set of fields that determine the appearance of the selected histogram. Refer to Table 9-3 for information about the fields.

After modifying the parameter values in the Histogram Properties Window, click on the Apply button. The new parameter values take effect immediately.



After modifying a parameter value, click on the Reset button to undo the change. The Reset button restores the value of the parameter if you have not applied the new value.

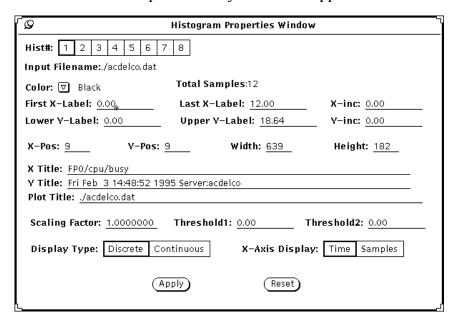


Figure 9-14. Histogram Properties Window

Table 9-3. Fields in the Histogram Properties Window

Field	Function
Hist#	The number of the histogram you want to modify. For example, if you select 2, the fields in the current Histogram Properties Window affect only the appearance of the second histogram.
Input Filename	The name of the statistics file that contains the statistics displayed in the histogram. This field is empty if you have not loaded the histogram.
Color Index	The color used for the selected histogram. The available colors are light gray, red, green, blue, yellow, dark gray, white, and black. The default color is black.
Total Samples	The total number of samples gathered and displayed in the histogram.
First X-Label	The first sample displayed in the histogram. By default, the histogram starts with the first sample and displays the time corresponding to the first sample as the first value on the X axis. However, you can, for example, specify that the histogram starts with the tenth sample by setting First X-Label to 10.
Last X-Label	The last sample in the histogram. By default, the histogram displays all samples. However, you can, for example, specify that only the first 10 samples are displayed by setting Last X-Label to 10.
X-inc	The increment on the X axis (horizontal axis). It controls the number of labels displayed on the X axis. For example, if you specify 10, the labels on the X axis are 10 seconds apart. If the first label is 11:32:10, then the second label is 11:32:20. A higher increment causes fewer timestamps to be displayed. The shape of the histogram is not affected by the X-inc.  The default value of X-inc is 0, which means that the X axis is automatically scaled.
Lower Y-Label	The first value marked on the Y axis (vertical axis). By default, the first value is 0.
Upper Y-Label	The last value marked on the Y axis. By default, it is the highest value attained by the statistics in the current histogram.



Table 9-3. Fields in the Histogram Properties Window (Continued)

Field	Function
Y-inc	The increment on the Y axis. It controls the number of labels displayed on the Y axis. For example, if you specify 2, the labels are two units apart. A higher increment causes fewer labels to be displayed. The shape of the histogram is not affected by Y-inc.  The default value of Y-inc is 0, which means that the Y axis is automatically scaled.
X-Pos	The spacing between the frame containing the histogram and the left edge of the window. A greater X-Pos value causes a larger amount of space between the frame and the left edge of the window.
Y-Pos	The spacing between the frame containing the histogram and the upper edge of the window. A greater Y-Pos value causes a larger amount of space between the frame and the upper edge of the window.
Width	The width of the box containing the histogram measured in pixels.
Height	The height of the box containing the histogram measured in pixels.
X Title	The title below the X axis. By default, it is the parameter name.
Y Title	The title at the top of the histogram. By default, it is the date and the server name. For example, in Figure 9-15, the Y Title is "Fri Feb 3 14:48:52 1995 Server: acdelco."
Plot Title	The string following Y Title. By default, it is the filename of the input file. For example, in Figure 9-15, the Plot Title is "./acdelco.dat."
Scaling Factor	A multiplier for the labels on the Y axis. For example, if Scaling Factor is 2, all the values on the Y axis are multiplied by 2. The shape of the histogram remains unchanged. By default, Scaling Factor is 1.
Threshold1 and Threshold2	The value on the Y axis at which <b>ax_perfhist</b> draws a dotted line across the histogram.
Display Type	The form of the histogram. The default is Discrete, which means that statistics are presented in bars; if Display Type is set to Continuous, a graph is plotted instead, as shown in Figure 9-16.
X-Axis Display	The units for the values on the X axis. By default, it is Time, which means that each value on the X axis is the time when the sample was collected. If set to Samples, the values on the X axis correspond to the numbers of the samples.



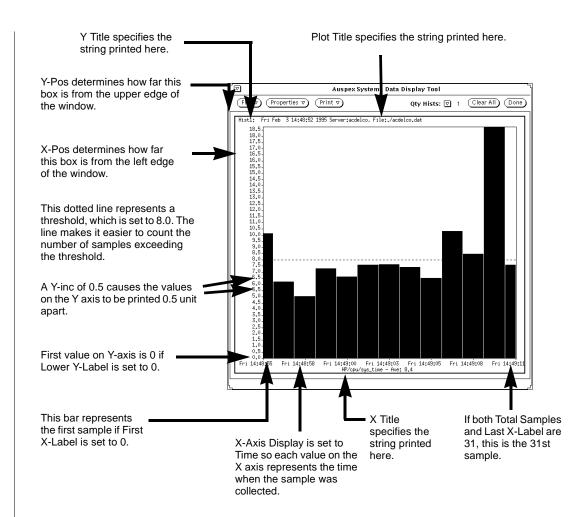


Figure 9-15. Effects of Histogram Properties parameters on a histogram

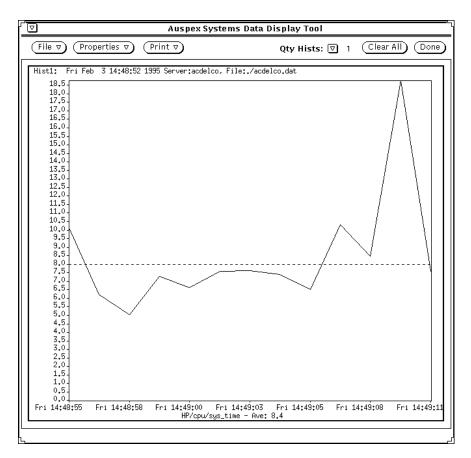


Figure 9-16. Server statistics plotted on a continuous graph

## Configuring the layout of the Data Display Window

You can design the overall appearance of the histograms in the Data Display Window. For example, you can determine whether histograms are arranged in columns or rows. The following procedure describes how to configure the overall properties of histograms:

1. In the Data Display Window, click on the Properties button to display a pull-down menu. Select "All Histograms..." in the menu. The Overall Properties Window opens (see Figure 9-17).

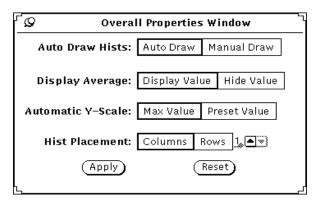


Figure 9-17. Overall Properties Window



- The Overall Properties Window lists the possible values for each parameter. Click on the appropriate value to select it. If you want to undo a modification and restore the original parameter value, click on the Reset button. Refer to Table 9-3 for the descriptions of the parameters in the Overall Properties Window.
- 3. Click on the Apply button to apply the change. The values apply to all histograms in the Data Display Window.



**Note:** The new Display Average parameter value does not take effect immediately after you click on the Apply button. It takes effect only in the histogram that is redrawn. For example, if three histograms are currently displayed, changing Display Average does not affect any of the histograms. However, when you load new statistics to the third histogram, the change takes effect in the third histogram. Similarly, if you redraw all the histograms, for example, by changing the Hist Placement parameter, the new Display Average value takes effect in all of the redrawn histograms.

Table 9-4. Fields in the Overall Properties Window

Field	Possible values	Functions				
Auto Draw Hists	Auto Draw (default)	The histogram is drawn automatically; you cannot adjust the size of a histogram by dragging it.				
	Manual Draw	To resize a frame, you can drag the frame that contains a histogram similar to the way you resize a window on the screen.				
Display Average	Display Value (default)	The average value of the statistics is displayed below the X axis.				
	Hide Value	The average value of the statistics is not displayed.				
Automatic Y- scale	Max Value (default)	The highest value on the Y axis is set to the maximum value of the statistics collected. For example, if the maximum value among all the samples is 28.1, the highest value on the Y axis is 28.1.				
	Preset Value	Automatic scaling is disabled. The highest value on the Y axis in the current histogram is maintained when you load different statistics to it. For example, if the highest value in the histogram is 28, Preset Value uses 28 as the highest value regardless of the parameter or input file you subsequently load to histogram 1.				
Hist Placement	Columns (default)	When multiple histograms are displayed, they are arranged in columns.				
	Rows	When multiple histograms are displayed, they are arranged in rows.				
	Number	Number of columns or rows used to display the histograms in the Data Display Window.				

# Saving and Applying a Configuration File

If you often use a specific layout of the Data Display Window, you can save the layout information to a configuration file. When you apply the configuration file, the Data Display Window arranges the histograms according to the specifications in the file.



Also, if you plan to use the **ax\_perfhist** command non-interactively—for example, for printing out histograms—define a configuration file. Refer to "Starting ax\_perfhist with Options and Arguments" on page 9-46 for more information on specifying and invoking the configuration file in the command syntax.

The following example illustrates how to save a configuration file that displays six histograms in the Data Display Window:

- 1. In the Data Display Window, select **6** for Qty Hists.
- 2. In the Load Window, select the type of statistics for each histogram using the Parameter button.
- 3. Configure each histogram and the global characteristics of the histograms according to "Defining the Appearance of a Histogram" on page 9-24.
- 4. In the Data Display Window, click on the Print button to display a pull-down menu. Select "Print Properties" in the menu. The Print Options window appears, which allows you to determine how the histograms are printed. For information on how to specify the print options, refer to "Configuring the print options" on page 9-39.
- 5. In the Data Display Window, click on the File button to display a pull-down menu. Select "Config files..." in the menu. The Configuration window appears.
- 6. In the Configuration window, type in the name of the configuration file and the directory that contains the file.
- 7. Click on the Save Config button in the Configuration Window (see Figure 9-18). When the system asks you to confirm, click on the Save button.

The configuration file contains information about how each histogram is displayed, how the histograms are arranged in the Data Display Window, and what print options are used if the histograms are sent to the printer or a file. All this information is loaded when you apply the configuration file.



Figure 9-18. Configuration Window



To apply a configuration file, follow these steps:

- 1. In the Configuration window, type in the name of the directory that contains the configuration file.
- Click on the Scan Dir button in the Configuration Window. The file names of the configuration files appear in the scroll list.
- Click on the configuration file you want to use. The histograms, which contain no data, appear in the Data Display Window. The number of histograms displayed depends on the configuration file.
- Open the File Load Window. While pointing to the input file name, hold down the right mouse button. A pull-down menu appears.
- In the pull-down menu, select "Load All Histograms." Statistics are loaded to the histograms, which pertain to the parameter types specified in the configuration file. For example, if the configuration file includes two histograms, one for HP/hp/usr\_time and one for NP0/cpu/busy, these two types of statistics from the selected input file are loaded to the histograms.

Figure 9-19 shows a Data Display Window that uses a predefined configuration file. In this configuration file, six histograms are displayed—the second histogram uses continuous plot, and the rest use discrete plot.

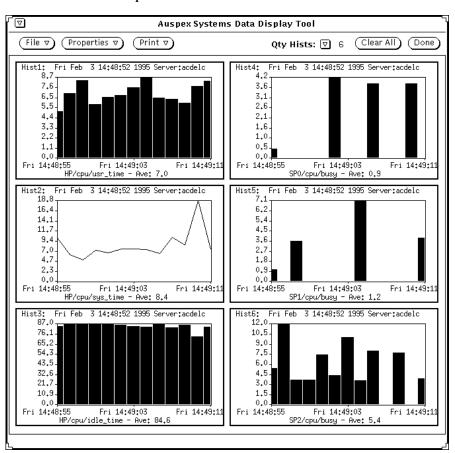


Figure 9-19. Displaying six histograms as specified in a configuration file



To remove a configuration file, open the Configuration Window and hold down the right mouse button. Click on the Delete Configuration File option. When the system asks you to confirm, click on the Delete button. The configuration file is removed from your directory.

## **Examining Samples in Histograms**

You can use the "zooming" feature provided in ax\_perfhist to take a close look at statistics that interest you. Zooming means that only samples collected within a specified period are displayed. The following example illustrates how to display samples from a histogram.

- 1. Load statistics to a histogram in the Data Display Window.
- 2. Click the middle mouse button on a sample displayed on the left side of the histogram. A marker, which is a vertical dotted line, appears. This line marks the first sample to be displayed after you zoom in.
- 3. To specify the last sample of the histogram after you zoom in, click the middle mouse button at the sample to the right of the first selected sample. Another marker appears as shown in Figure 9-20.

To change or remove a marker, follow these steps:

- a. Point anywhere in the histogram and hold down the right mouse button to display a menu, which is shown in Figure 9-20.
- b. Select "Reset Markers" from the menu. Both markers in the histogram disappear. To redraw a marker, repeat step 2. (The menu contains several other options, which are explained in "Menu for Zooming In and Zooming Out" on page 9-34.)
- c. If two markers exist and you want to redraw the first one, simply click the middle mouse button at the new location. For example, if the first marker is at 11:32:00 and the second at 11:33:00, you can click to move the first marker to 11:34:00. If you click at another location again, the marker that is initially at 11:33:00 is moved.



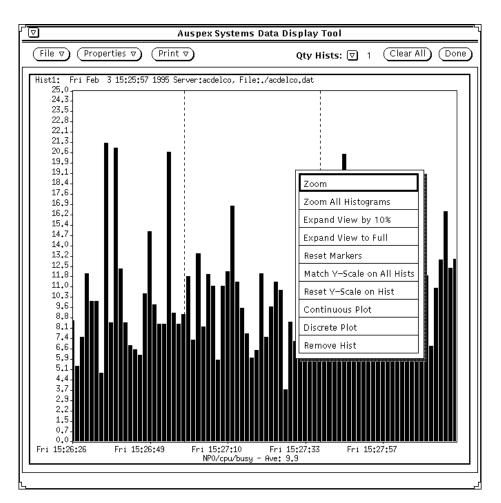


Figure 9-20. Pull-down menu for zooming samples between the markers

- To zoom in, click the right mouse button anywhere in the histogram. When the pulldown menu appears, click on the Zoom option. Figure 9-21 on page 9-34 illustrates the histogram after the zooming.
  - After you zoom in, it is easier to examine the statistics. The bars are now wider, and the X axis has more space to show the collection times for the samples.
- If multiple histograms are displayed, and you want to zoom in all the samples collected in the time period defined by the markers in one of the histograms, follow these steps:
  - Point anywhere in the histogram that contains the markers.
  - b. Press the right mouse button to display a pull-down menu. Click on the Zoom All Histograms option.

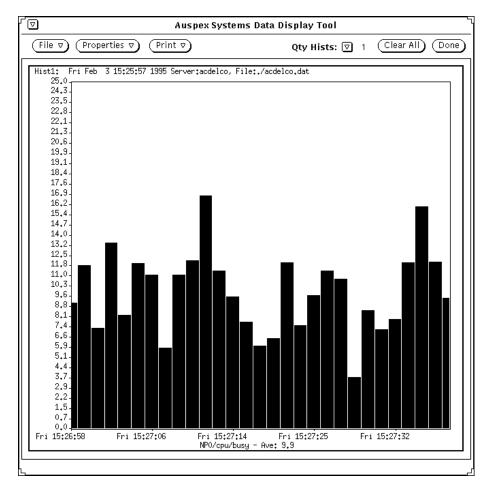


Figure 9-21. Histogram showing samples after zooming

## Menu for Zooming In and Zooming Out

When you point anywhere within a histogram and hold down the right mouse button, a menu consisting of 10 options appears. The functions of these options can be summarized as follows:

- ▲ Zooming in or zooming out samples you selected
- Resetting markers in a histogram
- ▲ Changing the scale on the Y axis
- ▲ Toggling between a continuous plot and a discrete plot
- Removing a selected histogram

Table 9-5 describes the function of each option in the pull-down menu. Some of the options affect only one histogram, while others affect all histograms currently in display. In the table, the histogram in which you click to invoke the pull-down menu is referred to as "the current histogram."



Table 9-5. Options in pull-down menu in Data Display Window

Option	Function
Zoom	Zoom in on the samples delimited by the markers in the current histogram. See "Examining Samples in Histograms" on page 9-32 for information on defining which samples to zoom.
Zoom All Histograms	If multiple histograms are displayed, zoom in on the samples in all the histograms. <b>ax_perfhist</b> selects the samples based on the markers set in the current histogram. This option is useful if you want to compare statistics that were collected in the same period of time.
Expand View by 10%	Display 10% more samples on either side of those samples that are currently zoomed. For example, if the histogram contains a total of 20 samples, this option displays two more samples next to each marker.
Expand View to Full	Display all the samples in the current histogram.
Reset Markers	Remove the markers currently in the histogram.
Match Y-Scale on All Hists	Standardize the Y axes in all the histograms by using the same scale, minimum value, and maximum value. The scale and values in the current histogram are copied to other histograms in the Data Display Window. For example, if histogram 1 uses 0.0 and 28.1 as the minimum and maximum values, respectively, and if you click in histogram 1 to select the menu option, the corresponding values in the other histograms change to match the ones in histogram 1.
Reset Y-Scale on Hist	Restore the scale, minimum value, and maximum value on the Y axis to the original settings before you used "Match Y-Scale on All Hists."
Continuous Plot	Display the statistics in a continuous graph. This option is equivalent to the Display Type field in the Histogram Properties window.
Discrete Plot	Display the statistics in bars. This option is equivalent to the Display Type field in the Histogram Properties window.
Remove Hist	Remove the current histogram from the Data Display Window. It is equivalent to the Delete Hist button in the Load Window. It does not remove the statistics file from which the histogram is derived.

## **Summing Parameters in Histograms**

Sometimes you might find it useful to sum two or more histograms to get an overall picture of server usage. For example, you can combine two histograms to display the total number of packets processed by Ethernet interfaces 0 and 1 on a server.

To combine histograms, follow this procedure:

- 1. Load the statistics as you normally would, specifying the type of statistics you want to display.
- 2. In the Load Window, click on the Sum Parameters box. A check mark appears in the box.
- 3. To add another histogram to the one being displayed, follow these steps:
  - a. If the other histogram is from another statistics file, select the file in the Load Window.
  - b. Click on the Parameters button, and specify the type of statistics to add to the current one.

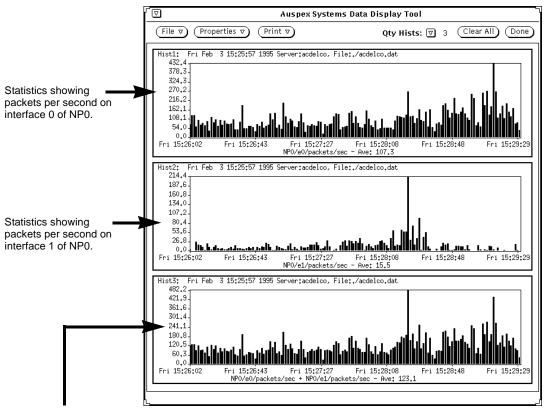


The histogram now contains statistics that you loaded in steps 1 and 3. In the Load File window and Data Display Window, the histogram title changes according to the summed parameters. The following example shows a histogram title after you sum two parameters in the histogram:

NP0/e0 packets/sec + NP0/e1 packets/sec

In addition to summing parameters, you can add statistics from different input files. For example, you can combine statistics from *server1.dat* and *server2.dat*, which contain data collected from two NetServers. Refer to "Pull-Down Menu Displayed in the Input File List" on page 9-37 for more information on adding statistics from different input files.

The example in Figure 9-22 shows a Data Display Window containing three histograms. The bottom histogram is the sum of the first two.



Sum of statistics from the first two histograms. That is, the statistics show packets per second on NP0 (both interface 0 and interface 1). To create this combined histogram, select this histogram, load the parameter used in the first histogram, click the Sum button, and load the parameter used in the second histogram.

Figure 9-22. Summing two histograms



## **Deleting an Input File**

To remove a statistics file that you have saved with **ax\_perfmon**, follow these steps:

- 1. Open the Load File window.
- 2. In the input file list, hold down the right mouse button to display a pull-down menu. Select "Delete File" in the menu. A box appears asking whether you want to delete the file. Click on Delete, and the file is removed from your directory.

The pull-down menu contains parameters other than "Delete File." These parameters are described in the next section.

## **Pull-Down Menu Displayed in the Input File List**

In the File Load Window, a pull-down menu appears when you point to the input file list and hold down the right mouse button. The menu offers a convenient way to sum statistics from different files, load statistics from an input file, delete a file from your directory, and so on. Table 9-6 explains the function of each option in the pull-down menu.

Table 9-6. Pull-down menu options displayed in the input file list

Option	Function
Load File	Load statistics from a selected file to a selected histogram. For example, if stats.dat and histogram 4 are selected in the File Load Window, data from stats.dat are displayed in histogram 4.  The most recently used parameter is used for the selected histogram. Suppose histogram 2 currently displays data from stats.dat and uses the HP/hp/usr_time parameter. If you want to use histogram 2 to display data from stats2.dat, use Load File from the pull-down menu. Statistics from stats2.dat replace the statistics from stats.dat, but the histogram continues to use the HP/hp/usr_time parameter.
Load All Histograms	Similar to "Load File," except all the histograms defined in the Hist# list are affected. For example, if histograms 1 through 8 are displaying statistics from <i>stats.dat</i> , you can select <i>stats2.dat</i> and invoke Load All Histograms to load statistics from <i>stats2.dat</i> . The histograms continue to use the parameters used before you loaded statistics from the new file.
Sum File with Existing File	Statistics of the same type but from different input files are added. For example, if histogram 1 currently contains statistics about HP/hp/usr_time collected in input file 1, you can add statistics about HP/hp/usr_time collected in input file 2. The resulting histogram shows the combined statistics from input files 1 and 2.
Load File to Threshold1	Display the samples up to the first threshold marker. For example, if Threshold1 is set at 10.0, this option causes the histogram to use 10.0 as the maximum value on the Y axis. If you want to display the entire histogram again, invoke the Load File option.
Delete File	Remove the input file from your directory.

# **Printing Histograms**

To print the histograms in the Data Display Window, click on the Print button in the Data Display Window. The Print button invokes the predefined print command from the Print Options window. You can print the histogram to a printer or a file.

This section describes the Print Options Window, which contains parameters determining how histograms are printed. After setting the print options, click on the Apply button in the Print Options Window. The Reset button restores the values of the print options if they have not been applied. Figure 9-23 explains the functions of the print options; for detailed information about configuring the options, refer to the following sections.

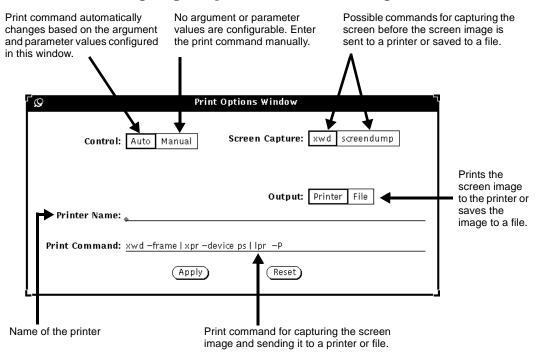


Figure 9-23. Print Options Window

#### **Print Command**

In the Print Options Window, you can specify the command that **ax\_perfhist** uses for printing the Data Display Window. The default print command does the following:

- Captures the screen using xwd
- Sends the output to a printer using lpr

The following is an example of the print command:



**Note:** Treat the default print command as a suggested command; it may need customizing depending on the printer setup at your site. For example, if you want to print to a printer other than the default one, append a printer name to the command.



## Changing the print command automatically

The Print Options Window contains several parameters that configure the print command. If you want the print command to be configured automatically, select Auto for the Control option. When Auto is selected, the change you make to any print option is reflected immediately in the Print Command field. Select Manual for the Control option to enter a command manually in the Print Command field.

## Configuring the print options

The information in this section applies only if the Control option in the Print Options Window is set to Auto.

The **screendump** and **pssun** commands accept several arguments: for example, the X and Y arguments for **screendump** set the X and Y coordinates of the upper left corner of the area to be dumped; the X and Y arguments set the width or height of the area to be dumped. The Print Options Window allows you to specify these values individually, thereby eliminating the need for memorizing the print command syntax. Follow these steps to configure the print command arguments:

- 1. Two commands can capture a screen: **xwd** and **screendump**. If you are unsure which command to use, refer to "Capturing the screen using xwd or screendump" now.
  - If you use xwd to capture the screen, go to step 2. If you are using screendump to capture the screen, assign values to other print options as follows:
  - a. Specify the **screendump** and **pssun** arguments. For more information about the arguments, refer to the man pages for these commands.
  - b. Select Monochrome or Color for the Screen Type option.
  - Select Portrait or Landscape for the View option to determine the screen image orientation in the output.
- 2. Select Printer or File for the Output option. The Output command is described in detail in "Print Command Output" on page 9-40.

As you configure the print options, the print command changes accordingly. After setting all the print options, click on the Apply button.

#### Capturing the screen using xwd or screendump

You can specify that ax\_perfhist uses xwd or screendump to capture the screen before sending the frame-buffer image to a printer or saving the image to a file.



**Note:** Be sure the directory containing **xwd** or **screendump** is in your path. ax perfhist does not enter the complete pathname for these commands in the Print Options Window.

Table 9-7 describes the differences between **xwd** and **screendump**.



Table 9-7. Differences between xwd and screendump

xwd	screendump
Used in the OpenWindows or X11/Motif environment.	Used with SunView and OpenWindows.
No arguments need to be specified if you print the screen dump to a printer. If you print to a file, you can specify the destination file name.	You can configure the x, y, X, and Y argument values, the Screen Type option, and the View option.
After clicking on the Print button in the Data Display Window to invoke the print command, the cursor on the screen changes to a plus sign (+). Click in the window that you want printed or saved. Make sure the window is not obscured by other windows.	Simply click on the Print button in the Data Display Window to invoke the print command.
The default <b>xwd</b> command is as follows: xwd -frame   xpr -device ps   lpr	The default <b>screendump</b> command is as follows: screendump -x -9 -y 38 -X 661 -Y 624   pssun -s 7.5 10.0 -  0.5 0.5   lpr

## **Print Command Output**

The Output option in the Print Options Window can be Printer, which prints out the Data Display Window, or File, which saves the captured screen image to a specified file.

## Saving a screen to a file

If you save the data to a file, two additional fields appear in the Print Options Window:

Dir Specifies the directory containing the destination file. By default, the

directory is /, which is the directory in which you started ax\_perfhist.

File Specifies the name of the destination file that contains the captured screen.

After specifying the output file, click on the Apply button in the Print Options Window. The file name appears in the Print Command field.

#### Printing a screen to a printer

If you select **xwd** to capture the screen, the screen image is piped to the **xpr** command, which prints out the image. If you select **screendump** to capture the screen, **ax\_perfhist** automatically uses the **lpr** command for printing. Regardless of the command you use for screen capture, always verify that the commands included in the Print Command field are in your path.

# Types of Statistics Displayed by ax\_perfhist

As discussed in "Loading Statistics From a File" on page 9-21, you must specify a parameter when loading statistics to the Data Display Window. The Parameter button in the File Load Window displays a pull-down menu that organizes the statistic types by processor types. The processors for which you can display statistics are HP, NP, FP, and SP.

This section describes the parameters by processor boards. For the NP and SP parameters, processor numbers or network interface numbers are used in the display to identify the processors or interfaces from which statistics were collected. For example, the NP0/e0/packets/sec parameter pertains to Ethernet interface 0 on NP0.



The Parameters menu lists the maximum number of boards. Your server may not have all the processor boards listed in the menu.



**Note:** The **ax\_perfhist** parameters for network interfaces, disks, virtual partitions, NFS operations, and the write cache on the SP are per second, not per interval as in ax\_perfmon.

#### HP

HP/cpu/usr\_time HP CPU utilization time in application.

HP/cpu/sys\_time HP CPU utilization time in kernel.

HP/cpu/idle\_time HP CPU idle time.

## NP

NP/cpu/busy NP CPU utilization.

NP/np/NFS/workers Number of NFS workers on the NP. NP/np/NFS/jobs queued Number of NFS jobs queued on the NP. NP/np/NFS/busy workers Number of busy NFS workers on the NP. NP/np/NFS/nfs vers 2 over UDP Number of NFS version 2 calls over UDP. NP/np/NFS/nfs vers 2 over TCP Number of NFS version 2 calls over TCP. Number of NFS version 3 calls over UDP. NP/np/NFS/nfs vers 3 over UDP NP/np/NFS/nfs vers 3 over TCP Number of NFS version 3 calls over TCP.

NP/np/FTP/puts Number of puts over FTP. NP/np/FTP/gets Number of gets over FTP.

NP/np/FTP/put bytes Number of put bytes over FTP. NP/np/FTP/get bytes Number of get bytes over FTP.

NP/np/FTP/workers Number of FTP workers.

NP/np/UDP/dgrams in Number of UDP dgrams received. NP/np/UDP/dgrams out Number of UDP dgrams sent out. Number of UDP checksum errors. NP/np/UDP/cksum errs NP/np/TCP/segs in Number of TCP segments received. NP/np/TCP/segs out Number of TCP segments sent out. NP/np/TCP/segs w/ error Number of TCP segments with errors.

Number of active connections. NP/np/TCP/curr. estab. conns NP/np/TCP/active opens Number of active open TCP. NP/np/TCP/passive opens Number of passive open TCP. Number of TCP workers. NP/np/TCP/workers



NP/ip/TCP/pkts in	Number of TCP packets received.
NP/ip/TCP/pkts out	Number of TCP packets sent out.
NP/ip/TCP/pkts w/ error	Number of TCP packets with errors.
NP/ip/TCP/pkts forwarded	Number of TCP packets forwarded.
NP/e/packets/sec	Number of packets processed by the Ethernet interface.
NP/e/bytes/sec	Number of bytes processed by the Ethernet interface.
NP/e/incoming bytes/sec	Number of bytes received by the Ethernet interface.
NP/e/outgoing bytes/sec	Number of bytes sent by the Ethernet interface.
NP/e/incoming errors	Number of errors received by the Ethernet interface.
NP/e/outgoing errors	Number of errors sent by the Ethernet interface.
NP/e/outgoing collisions	Number of collisions on the Ethernet interface.
NP/e/outgoing discards	Number of outgoing packets discarded by the Ethernet interface.
NP/e/outgoing error percentage	Percentage of errors sent by the Ethernet interface.
NP/fddi/Incoming/Outgoing Frames/sec	Total number of packets received and sent on the FDDI interface.
NP/fddi/Incoming/Outgoing Bytes/sec	Total number of bytes received and sent on the FDDI interface.
NP/fddi/Incoming Frames/sec	$Number \ of \ frames \ received \ on \ the \ FDDI \ interface.$
NP/fddi/Incoming Bytes/sec	Number of bytes received on the FDDI interface.
NP/fddi/Incoming Frame Errors/sec	Number of frames and errors received by the FDDI interface.
NP/fddi/Outgoing Frames/sec	Number of frames sent by the FDDI interface.
NP/fddi/Outgoing Bytes/sec	Number of bytes sent by the FDDI interface.
NP/fddi/Outgoing Frames Errors/sec	Number of frames and errors sent by the FDDI interface.
NP/fddi/Max transmit frame size (mtu)	The MTU value on the FDDI interface.
NP/fddi/Outgoing error percentage	Percentage of errors sent on the FDDI interface.
NP/E/packets/sec	Number of packets processed by the Ethernet interface.
NP/E/bytes/sec	Number of bytes processed by the Ethernet interface.



NP/E/incoming errors Number of errors received by the Ethernet

interface.

Number of errors sent by the Ethernet interface. NP/E/outgoing errors

NP/E/outgoing collisions Number of collisions on the Ethernet interface.

NP/E/outgoing discards Number of outgoing packets discarded by the

Ethernet interface.

NP/E/outgoing error percentage Percentage of errors sent by the Ethernet

interface.

NP/atm/packets/sec Number of packets processed by the ATM

interface.

NP/atm/bytes/sec Number of bytes processed by the ATM interface.

NP/atm/incoming bytes/sec Number of bytes received by the ATM interface.

NP/atm/outgoing bytes/sec Number of bytes sent by the ATM interface.

NP/atm/incoming errors Number of errors received by the ATM interface.

NP/atm/outgoing errors Number of errors sent by the ATM interface.

NP/atm/outgoing discards Number of outgoing packets discarded by the

ATM interface.

NP/atm/outgoing error percentage Percentage of errors sent by the ATM interface.

NP/atm/Max transmit frame size

(mtu)

NP/atm/Offnet transmit frame size

(offmtu)

The MTU value on the ATM interface.

The MTU value on the ATM interface for networks not directly attached to the server.

FP

FP/cpu/busy CPU utilization on the FP.

FP/lfs/mount LFS mount operations.

FP/lfs/unmount LFS unmount operations.

FP/lfs/read NFS read operations.

FP/lfs/write NFS write operations.

FP/lfs/readdir NFS read directory operations.

FP/lfs/readdirplus NFS V3 read directory plus operations

FP/lfs/access NFS V3 file access check

FP/lfs/commit NFS V3 commit (flush to disk) operations

FP/lfs/readlink NFS read link operations.

FP/lfs/null NFS null operations.

FP/lfs/getattr NFS get attributes operations.



FP/lfs/setattr NFS set attributes operations.

FP/lfs/lookup NFS lookup operations.
FP/lfs/create NFS create operations.
FP/lfs/remove NFS remove operations.
FP/lfs/rename NFS rename operations.
FP/lfs/link NFS link operations.

FP/lfs/symlink NFS symlink operations.

FP/lfs/rmdir NFS remove directory operations.

FP/lfs/statfs NFS statfs operations.
FP/lfs/fsync NFS fsync operations.
FP/lfs/access NFS access operations.

FP/lfs/syncfs NFS sync file system operations.

FP/lfs/quota NFS quota operations.

FP/lfs/total Total number of NFS operations.

FP/lfs/max Maximum number of NFS operations.

FP/fp/data cache LRU Eject Age Average age BQ\_LRU data bufs dropped from

cache.

FP/fp/data cache LRU Eject Count/s Count of BQ\_LRU data bufs dropped from cache.

FP/fp/data cache AGE Eject Age Average age BQ\_AGE data bufs dropped from

cache.

FP/fp/data cache AGE Eject Count/s Count of BQ\_AGE data bufs dropped from cache.

FP/fp/meta cache LRU Eject Age Average age BQ\_LRU ctl bufs dropped from cache.

FP/fp/meta cache LRU Eject Count/s Count of BQ\_LRU ctl bufs dropped from cache.

FP/fp/meta cache AGE Eject Age Average age BQ\_AGE ctl bufs dropped from cache.

FP/fp/meta cache AGE Eject Count/s Count of BQ\_AGE ctl bufs dropped from cache.

FP/fp/inode cache Eject Age Average age of inodes dropped from cache.

FP/fp/inode cache Eject Count/s Count of inodes dropped from cache.

FP/fp/inode cache Hit Access Rate Rate of inode cache hit access.

1

Count/s

FP/fp/inode cache Hit Access

Number of inode cache hits per second.

FP/fp/name cache Eject Age Average age of items dropped from the name

cache.

FP/fp/name cache Eject Count/s

Number of items dropped from the name cache per

second.

FP/dat bstats/data cache Hit Access Rate of data cache hit access.

Rate



FP/dat\_bstats/data cache Hit Access Number of data cache hits per second.

Count/s

FP/ctl bstats/meta cache Hit Access Rate of meta cache hit access.

Rate

FP/ctl bstats/meta cache Hit Access Number of meta cache hits per second.

Count/s

FP/nc/name cache Hit Access rate Rate of name cache hit access.

FP/nc/name cache Hit Access

Count/s

Number of name cache hits per second.

## SP

SP/cpu/busy CPU utilization on the SP.

SP/nvram/io count Number of I/O operations.

SP/nvram/overrides NVRAM overrides. SP/nvram/coalesce **NVRAM** coalesces.

SP/nvram/buf used [PAGE 8K] Number of page size 8K buffers used in NVRAM.

SP/nvram/buf\_used [PAGE\_512B] Number of page size 512B buffers used in

NVRAM.

SP/vpar\_total/vp Total operations per second in the virtual partition.

Number of sectors read on a selected disk. SP/Disk/disk/sctr\_rd Number of sectors written on a selected disk. SP/Disk/disk/sctr\_wr

Number of read operations on a selected disk. SP/Disk/disk/ops\_rd SP/Disk/disk/ops\_wr Number of write operations on a selected disk.

Total number of operations on a selected disk. SP/Disk/disk/ops\_total

Total amount of time spent on each operation on a SP/Disk/disk/total\_time\_per\_op

selected disk. It is the sum of queue time and

service time.

SP/Disk/disk/queue\_time\_per\_op Time an operation request waits to be serviced.

SP/Disk/disk/ Amount of time needed to service an operation

request. service\_time\_per\_op

SP/Disk/disk/drive utilization Percentage of utilization on the drive. SP/all disks/sctr\_rd Number of sectors read on all disks. SP/all disks/sctr wr Number of sectors written on all disks. SP/all disks/ops\_rd Number of read operations on all disks.

SP/all disks/ops wr Number of write operations on all disks. SP/all disks/ops\_total Total number of operations on all disks.



SP/all disks/total_time_per_op	Total amount of time spent on operations on all disks. It is the sum of queue time and service time on all disks.
SP/all disks/queue_time_per_op	Amount of time that operation requests for all disks wait to be serviced.
SP/all disks/service_time_per_op	Amount of time taken to service operation requests on all disks.
SP/all disks/drive_utilization	Percentage of utilization of all disks.

## Starting ax\_perfhist with Options and Arguments

The full command syntax for **ax\_perfhist** is as follows. Refer to Table 9-8 for explanations for the options and arguments used in the syntax.

ax\_perfhist [-cf configfilename] [-df datafilename] [-pf
print\_to\_filename] [-lp] [-qt]

Table 9-8. Options and arguments used in the ax\_perfhist command

Option and argument	Function
-cf configfilename	Specifies the configuration file that determines the type of statistics in each histogram, the appearance of each histogram, the window layout, and so on. For more information on saving a configuration file, refer to "Configuring the layout of the Data Display Window" on page 9-28.  Always define a configuration file before using the ax_perfhist command to print the histograms to ensure that the output contains the types of statistics you want and prints in the proper format.
-df datafilename	Specifies the name of the file containing the statistics recorded by <b>ax_perfmon</b> . The statistics in this file are loaded to the histograms.
-pf <i>print_to_filename</i>	Specifies the name of the file to which the histograms are saved. You can save the output of <b>ax_perfhist</b> and print them out later.
-lp	Prints the histograms once <b>ax_perfhist</b> loads the configuration and data files.
-qt	Quits ax_perfhist after loading the configuration and data files and printing.

The following are examples of the **ax\_perfhist** command:

```
ax_perfhist -cf 6histogram.cfg -df host1.dat
```

This command displays the Data Display Window using the configuration in *6histogram.cfg*. The histograms display statistics loaded from *host1.dat*.

```
ax_perfhist -cf 6histogram.cfg -df host1.dat -pf host1histogram -qt
```

This command prints the histograms to the file named *host1histogram*. **ax\_perfhist** quits after it saves the screen image to *host1histogram*.

```
ax_perfhist -cf 6histogram.cfg -df host1.dat -lp -qt
```

This command prints out the histograms and quits when it is finished printing.



# **Analyzing Performance Monitor Data**

By observing the performance data over an extended period of time, you can determine when to distribute file systems over multiple drives, whether to change the distribution of client workstations on the attached networks, and whether additional memory or processor boards in the NetServer will improve system performance.

## Do I need more I/O cache memory?

Observe the data cache age bar on the System Summary screen. If the bar shows all 'A's or 'a's, then all buffers in the cache were accessed within the last minute. This is a strong indicator that your NP needs more memory. All the I/O cache memory is on NP boards. A NetServer with enough memory has a few 'A's at the left, representing the most active buffers, but has higher letters towards the right. As long as a few buffers that are 5 to 10 minutes old ('J' or greater), then the system does not need more memory.



**Note:** The difference between uppercase and lowercase letters in the data cache age bar is insignificant; as long as some buffers are greater than 'J' or 'j,' the NetServer has enough memory. (Refer to the ax\_perfmon(8) man page for details.)

#### Do I need more NPs?

A NetServer might need another NP for two reasons: first, individual networks are too heavily loaded and another interface is required to attach additional networks to the NetServer; second, an NP is CPU-bound.

The top section of the System Summary screen shows CPU utilization for all processors, so you can easily identify a CPU-bound NP. If an NP's CPU utilization is consistently above 80 percent, the processor may be a bottleneck. You may avoid adding an extra processor by rearranging the way networks are attached. One NP may have two heavily loaded networks while another has two lightly loaded networks. Putting one lightly loaded network and one heavily loaded network on each processor may eliminate the bottleneck. To identify heavily loaded networks, check the bytes-per-second bar on the NP Statistics screen.

#### Do I need more SPs?

Add more SPs if you are adding more devices. The SP IV and SP V support 42 drives each. Adding SPs may also be indicated if SP processor utilization is consistently above 80 percent.

## Are my file systems and client workstations properly distributed across the available disk drives?

The disks in a NetServer can handle up to 70 to 80 disk I/O operations per second. The bars in the bottom section of the System Summary screen measure up to 80 operations per second, so a bar that is consistently full may indicate a disk bottleneck.

Disk traffic typically occurs in bursts. When a disk is active, it often shows 40 to 50 operations per second. This is a problem only if a particular disk seems to be much more heavily used than the others. If two heavily used file systems are on that one disk, consider moving one of the file systems to another disk. In cases where the files are not convenient to separate, use virtual partitions to stripe the file system across multiple disks.



#### ▲ Do I need write acceleration, or more of it?

The Write Accelerator is optional on the NetServer. The Write Accelerator improves the client NFS response time and the single-user write performance for large files.

The SP statistics help you evaluate the need for write acceleration on an SP. If the write I/O activity for an SP appears high, adding a Write Accelerator can improve the performance. With a Write Accelerator installed on an SP, you can enable the Write Acceleration function for the busy file systems with an option added to the entry in <code>/etc/fstab</code>. The file systems experiencing a high number of writes receive help from the write cache, reducing the overall disk write activity. To determine which file systems might require write acceleration, review the <code>ax\_perfmon</code> disk statistics for drives that consistently experience a high volume of write operations.

If your SP has a Write Accelerator installed and the disk I/O activity continues to appear high, consider adding a Write Accelerator to another SP board if that board does not already have one. Then you can distribute the busy file systems on different SPs.

Refer to Chapter 7, "Write Acceleration" for more information.



# 10 Special Maintenance

# **About This Chapter**

This chapter describes Auspex-specific features and commands that may not be essential for your NetServer's day-to-day operation but are valuable for system management and reporting server problems to Auspex.

This chapter discusses the following types of commands or procedures:

A NetServer configuration tool (the **ax\_config** command) that organizes configuration data such as part numbers for field replaceable units and revision numbers for software modules.



**Caution:** Only Auspex-authorized personnel should modify the configuration file, /var/adm/config.report. Use ax\_config only to display configuration information or print out the information. This chapter describes only the contents of the report and how to display it.

- A NetServer utility,ax\_load\_flash, to download Flash PROM firmware to the SP, and NP boards, as well as to 10Base-T Ethernet, FDDI, and MLT-3 SBus adapters installed on the NP board.
- Procedure for recovering from a system crash.
- Network-related issues, including commands for improving the server's connectivity and availability, such as **ax\_arp**, **ax\_netstat**, and **nfsstat**.



**Note:** For information on optional software products, including installation and documentation, refer to the Auspex Premier Software Series CD-ROM.



# **Configuration Management**

The ax\_config command is a valuable aid in configuration management. It displays and modifies system configuration data, which is stored in the /var/adm/config.report file. (Only Auspex-authorized personnel should modify the data.) The data is particularly useful when you troubleshoot the server and try to determine whether you need to replace a hardware unit.

**ax\_config** extracts system configuration information from three sources:

- /var/adm/config.report, which contains information reflecting the system configuration the last time **ax\_config** modified /var/adm/config.report
- Software modules, which ax\_config probes to obtain the software version numbers
- Hardware devices, which ax\_config probes to obtain hardware part numbers



**Caution:** The data format in /var/adm/config.report is critical for ax\_config. Do not manually use a text editor to modify this file.

When you invoke ax\_config in verbose mode (that is, when you enter ax\_config -v), messages similar to the following appear, indicating the types of information being extracted:

```
Loading Hardware Data
Performing Software Probe
Performing Hardware Probe
Devices on SP0: 0 1 2 3 6 7 8 9 10
Devices on SP1: 0 1 2 3 4 5 6
```

Figure 10-1 illustrates how **ax\_config** obtains the configuration information.



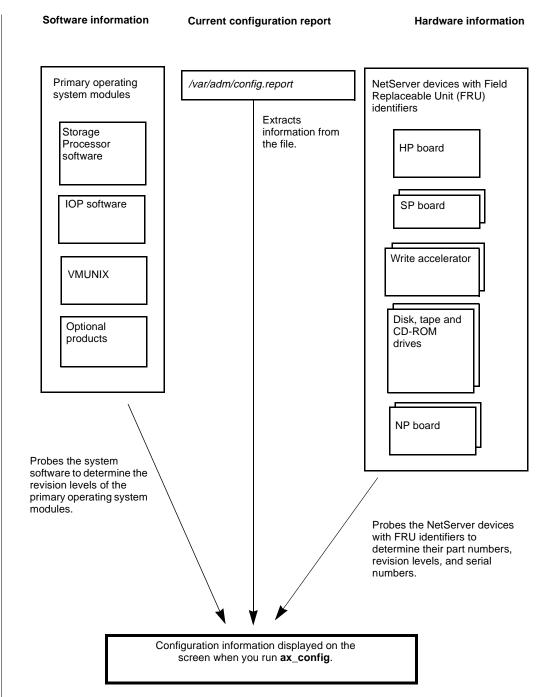


Figure 10-1. How ax\_config obtains configuration information

# Starting ax\_config

You must log in as root to use ax\_config, which runs in interactive or non-interactive mode:

Interactive mode

The following commands invoke **ax\_config** interactively:

ax\_config, which displays the main menu

ax\_config -v, which displays status messages when probing the software and hardware before displaying the main menu

If the -a option is in either command, ax\_config displays information from /var/adm/config.report in addition to the probe data when you choose the **d** command in the **ax\_config** main menu.

Non-interactive mode The following commands invoke **ax\_config** non-interactively:

```
ax config [ -o filename ] [ -dva ]
ax config [ -i filename ] [ -wv ]
```

The first command form is described in detail in "Non-interactive Mode" on page 10-5. Use the command to copy the contents of /var/adm/config.report to another file. The second command form copies the configuration information from a file that you saved or edited into *config.report*; it is used only by Auspex-authorized personnel.

## **Interactive Mode**

To run ax\_config interactively, enter ax\_config or ax\_config -v. The -v option means verbose mode, which displays status messages while ax\_config probes the software and hardware. The following are sample messages that appear if you execute ax\_config -v:

```
Loading Hardware Data
Performing Software Probe
Performing Hardware Probe
Main Menu: ax config
   c Compare Hardware Data to Hardware Probe
  d Display Configuration Data
   e Edit Hardware Data
   i Initialize Hardware Data
   l Load Hardware Data from EEPROM
   w Write Hardware Data to file
   q Quit
   enter Choice ==>
```

The following section describes how to display configuration data with the **d** command in the main menu. Other commands in the menu are for modifying the configuration report; only Auspex-authorized personnel should use the other options.



## **Displaying Configuration Data**

A **d** entered in the **ax\_config** main menu prints a header listing basic system configuration information (such as customer name, the system's serial number, installation date, date of most recent hardware upgrade, and name of field service representative). Auspexauthorized personnel review the header information and edit it if necessary when making hardware or software changes to the system.

The software probe data and hardware probe data follow the header. If the **ax\_config** command uses the **-a** option, hardware data from /var/adm/config.report also displays.

The following list describes special symbols used in the configuration report:

x's in the part number field When the part number is xx-xxxx, the line is used as a

delimiter to separate the devices that can store Field Replaceable Unit (FRU) identifiers from those that cannot.

? When the revision number or serial number cannot be

determined, a question mark (?) is printed.

Numbers in parentheses If the number is in an entry for a drive or board, the

number is the firmware revision number. If it is for a memory module, the number is the size of the memory

module.

After a hardware upgrade, the information stored in /var/adm/config.report may become obsolete. The Auspex-authorized engineer who upgrades the hardware for your server usually modifies the configuration information to facilitate future troubleshooting efforts.

## **Non-interactive Mode**

Use **ax\_config** in non-interactive mode if you want to display configuration information or write information to a file so you can transmit or print it.

## **Displaying Configuration Information**

To display configuration information, enter the **ax\_config** command as follows:

```
% ax_config -d
```

**ax\_config** displays configuration data obtained from the software and hardware probe. To include data from /var/adm/config.report, enter the following:

```
% ax config -da
```

If the configuration file does not exist, the following message appears:

```
ax_config: /usr/adm/config.report not found.
```

These commands, however, do not write data to the *config.report* file.



## **Copying Configuration Information to a File**

To save the configuration information to a file, enter the **ax\_config** command as follows:

% ax\_config -o filename -da

This command saves the hardware data from config.report, hardware probe data, and software probe data to a file, which can then be transmitted or printed. You can also manually edit this file to add information about hardware components that ax\_config cannot probe.



# Flash PROM Download Utility

This section describes the flash PROM download utility used to install new versions of PROM code on NetServer processor boards and SBus cards.

The following utility is available:

ax\_load\_flash for downloading new Flash PROM firmware to the SP V, and NP IV boards, as well as to the FDDI and MLT-3 SBus adapters installed on the NP board.

As a result, PROM version changes to these hardware components no longer require returning the board to the factory for a PROM upgrade. Instead, the upgrade is accomplished entirely through the download utility and firmware code files distributed either on the CD-ROM for a major release or as part of a patch release.

The **ax\_load\_flash** download utilitys provides the following features:

- Verifies whether a specified processor board or SBus adapter's current PROM code matches a specified file containing PROM code.
- Reads a specified processor board or SBus adapter's current PROM code to a designated file for backup purposes.
- Replaces a specified processor board or SBus adapter's PROM code by downloading a specified file and automatically verifies that the PROM code now matches the file.



**Caution:** The ax\_load\_flash command erases the original contents of the Flash PROM. Failure or interruption of the download process could leave the network interface or processor board inoperative. Downloading incompatible firmware to the interface or board will also leave it inoperative. It is highly recommended that only approved firmware revisions be used.

For FDDI and MLT-3, we also highly recommend that the interface be taken offline and physically disconnected from the ring before attempting the firmware download.

If a failure occurs, the download process may be repeated until it succeeds.

# ax load flash

The following processor boards and SBus cards support the **ax\_load\_flash** utility:

- ▲ SP V
- NP IV
- FDDI and MLT-3 SBus adapter on the NP IV



**Caution:** Downloading firmware to SP, NP, and MLT-3 boards with ax load flash requires you to place the NetServer in single-user mode and reboot the NetServer once the download is complete and verified. All other **ax\_load\_flash** requests can be done in single- or multiuser mode.



The following are examples of using the **ax\_load\_flash** utility:

To display the boot and Flash firmware revision levels for a specified interface, such as afddi1, type the following:

ax\_load\_flash -d afddi1

where **-d** is the display-only option.



**Note:** Flash firmware is not downloaded with the -d option.

To download the contents of a specified file (for example, fddi.4.4.flash) to a specified interface such as afddi0, enter the following:

ax\_load\_flash afddi0 fddi.4.4.flash

When a major release or patch release requires a processor board's Flash PROM to be upgraded, the release note includes specific instructions that identify the board(s) to upgrade and provides the pathname to the file containing the PROM code. Follow those instructions carefully.

For additional information, refer to the ax\_load\_flash man page.



# **Dealing with Server Problems**

This section provides special maintenance information about error checking, system crash procedures, and message logging, which can be useful if you encounter system problems. This section also gives useful tips for resolving problems if they occur. This section also describes how to obtain assistance from Auspex.

# Rebuilding the System Kernel

The procedure for rebuilding the Auspex NetServer kernel is identical to the equivalent Sun procedure. The directory /usr/sys/aushp/conf holds binaries and object files needed to rebuild the kernel. The default configuration file in the *conf* directory is named *AUSPEX1*.

The following steps outline the procedure for rebuilding the kernel:

1. As root, change directories to /usr/sys/aushp/conf and copy the default configuration file:

```
cd /usr/sys/aushp/conf
cp AUSPEX1 NEWCONFIG
```

where NEWCONFIG is the new configuration file.

2. If necessary, edit the new configuration file:

```
vi NEWCONFIG
(edit information)
:wq
```



**Note:** When modifying the configuration file, edit only sections of the file that describe non-Auspex devices. Do not change any lines in the "Auspex Devices" section of the file.

3. Complete the kernel rebuild:

```
config NEWCONFIG
cd ../NEWCONFIG
make
mv /vmunix /vmunix.old
mv vmunix /
reboot
```

This completes the procedure for rebuilding the kernel. For more information, refer to Sun's System & Network Administration manual.



**Note:** If you experience problems with rebuilding the kernel, contact Auspex Technical Support.

# **System Panics and Crashes**

The NetServer can hang or crash, causing a failure to respond to commands. This section discusses what happens when the NetServer encounters a system problem.



**Note:** When a NetServer hangs with no response to commands, consider contacting Auspex Technical Support before rebooting. In some cases, the system is not hung and Technical Support can bring back the system without rebooting.

When the NetServer crashes, it performs the following steps:

- ▲ Attempts to execute a **sync** to force changed blocks to be written to the disks.
- ${\color{red}\blacktriangle}$  Prints a short message describing the cause of the crash. The message is in this format:

```
panic: error message
```

- ▲ Writes all kernel panic messages to /usr/adm/messages. If you cannot obtain the crash error message on the system console, look in this file for the information. You can also retain crash messages in this file in order to keep a record of system behavior over time.
- ▲ Attempts to save a crash dump by writing a core image of memory from each processor into the primary swap partition on disk. After reboot, the **savecore** program writes this image to the directory /var/crash/hostname.

Upon a system crash, an NP produces two core dumps—one for each CPU on the board. For NP III an NP IV boards, the file names are "core3iop\_np*n.m*" and "core3iop\_fp*n.m*." As in the file names for other core dumps, the *n* in the filename is the number of the board (the first NP in the system is numbered 0); the *m* is the suffix of the core dump file.

- ▲ Attempts to reboot.
- ▲ Runs **fsck** on the file systems when rebooting.
- ▲ If no inconsistencies are found during **fsck**, the NetServer resumes multi-user operations.



**Note:** To examine /var/adm/messages or /var/crash/hostname in single-user mode, first mount /var; it is not mounted by default in single-user mode.

Crash dumps can accumulate in /var/crash/hostname until they fill up available disk space. Save the files to tape, if appropriate, and delete periodically.

You can also disable the saving of crash dumps by commenting out the following lines in /etc/rc.local:

```
mkdir -p /var/crash/'hostname'
echo -n 'checking for core dump...'
intr savecore /var/crash/'hostname'
echo ''
```

To enable the saving of crash dumps to a specified directory, leave the lines uncommented.



Sometimes the system hangs without crashing. In this situation, you may want to force a crash dump for debugging purposes. To do this, first abort to the PROM monitor prompt by pressing the Break key. Then enter:

HP> go 0



**Note:** Remember to type a space between the command name and its argument.

This forces the crash dump to be saved in the directory /var/crash/hostname according to the savecore procedure described previously.

# Message Logging

Auspex's error message handling resembles that of SunOS in most respects. The major difference is that messages are recorded from the multiple system processors instead of from a single processor. The **savecore** program logs messages from the processors (see the preceding section).

Some processor error messages are logged by the daemon **syslogd** to the file /var/adm/messages, according to the configuration parameters in the syslog.conf configuration file (see the syslog.conf(5) man page). All kernel panic messages are logged to the file. Appendix E contains a list of Auspex-specific messages, including those related to the boot function and those related to the Auspex processors.



**Note:** Each error message includes the name of the Auspex processor that reports the error. For example, the following message is generated by the third SP in the system:

```
SP2: 1203:Recovered, drive write fault, slot 44.
```

If the File Processor on an NP generates the error, the processor is identified as "IOPn/FPn CPU B"; if the network processor on an NP generates the error, the processor is identified as "IOP*n*/NP*n* CPU A." In both cases, *n* is the number of the NP board. For example, in the following messages, the first message is generated by the File Processor on the first NP board, and the second one is generated by the Network Processor on the second NP board:

```
IOPO/FPO CPU B: 1010: /disks/hweng.home: file system full
IOP1/NP1 CPU A: 1114: ae4: No carrier - transceiver cable
problem?
```

For more information on error logging, see Chapter 8 in Sun's System and Network Administration Guide.

In addition to messages being logged to /var/adm/messages, Auspex-specific notice, informational, and diagnostic messages are logged to /var/log/auspex-messages.



# **Using the Continue Command**

At the monitor prompt, the **continue** (**co**) command resumes execution of the operating system. This command is especially helpful if you unintentionally enter a Break on the console, because it allows you to continue without rebooting the system.



**Note:** A Break signal may be generated when you turn the console terminal off and then back on. In this case, you can also use the **co** command to continue without rebooting.

The **continue** command, which can be abbreviated to **co**, is executed in the following format:

```
HP> co [ virtual_address ]
```

The *virtual\_address* argument allows you to select the virtual address of the instruction to be executed when the program resumes. By default, the program resumes at the instruction located at the address pointed to by the program counter.



**Note:** The **co** command may not be able to restore the system to the state in effect at the time of the Break, depending on how long you waited before entering **co**. If **co** fails to restart the operating system, reboot the system.

# **Remote Diagnostics**

If you contact Auspex for help in resolving an Auspex NetServer problem, the service representative may need to perform diagnostics on the system remotely. This section outlines the procedure for obtaining remote diagnostic assistance.

- 1. Enable diagnostic mode by setting the diagnostic switch on the HP to the diag position. (Refer to the hardware manual for your server for the location of the diagnostic switch.)
  - The following procedure describes access with a modem connected to the NetServer. Other access methods may be to a terminal server, which can be connected to serial port b (ttyb). You can access a terminal server through a modem or the Internet.
- 2. Attach a modem to ttyb. Verify the modem is powered on and the telephone cable is properly attached. Refer to the modem's user documentation for information on connecting and configuring the modem.
- 3. Depending on the nature of the problem, the Auspex service representative may ask you to shut down the system. If so, do the following:
  - a. Shut down the system using **shutdown**(8) or **halt**(8).
  - b. When shutdown is complete and a PROM monitor prompt appears on the console, power down the NetServer.
- 4. Notify the Auspex service representative to establish a connection to the modem attached to the NetServer.

Once the modem connection is established, follow the service representative's instructions to reboot the NetServer.



As the system boots, the service representative can put the system in echo mode so that all input and output is displayed both on the local and remote consoles. During the session, the service representative may use the echo mode messaging facility to send instructions or information to your local console. Follow all instructions from the service representative carefully.

- When the remote diagnostic session is completed and the NetServer is ready for normal operation, follow these steps to restart the system:
  - a. Power off the modem.
  - b. Disable diagnostic mode by setting the diagnostic switch on the HP to *norm*.
  - c. At the monitor prompt, boot the server to multiuser mode by using the **b** command.

# **Managing Network Interfaces**

This section gives information about commands for network management and then gives a tip for network maintenance.

The ax\_perfmon command described in Chapter 9 provides an overall picture of server performance and of statistics on each network interface. Occasionally, however, you might need to get detailed information on a server interface such as high-level protocol statistics to investigate network problems. Auspex provides three commands for you to manage and monitor individual network interfaces on your NetServer:

- **ax\_arp**(8C) displays and flushes the ARP table on a network interface.
- **ax\_netstat**(8C) shows network statistics for a network interface.
- ▲ **nfsstat**(8C), which is part of SunOS, is modified by Auspex to display NFS statistics *per interface*.

The following three sections describe how to use these commands. For further information on the syntax, refer to the corresponding man pages.

# Displaying and Flushing ARP Tables Used by an NP Board

Because a NetServer can have multiple network processors and each of them maintains a different ARP table, Auspex provides a command to manipulate ARP tables per-board.

The command for flushing and displaying the ARP table is /auspex/ax\_arp, which has the following syntax:

```
ax_arp -f|-d -I interface
```

where interface is a network interface number such as **ae0** for an Ethernet interface or **afddi0** for an FDDI interface.



**Note:** The output of the command pertains to the entire board that contains the specified interface.

To display the current entries in the ARP table, use the -a option. The following is an example of the ax\_arp -a command and its output:

```
host0> ax_arp -a -I ae0
host1-e1 (144.48.252.4) at 0:0:3c:0:39:1d
```

This example shows the ARP table used by host0-ae0 contains one entry, which resolves the IP address (144.48.252.4) of host1-e1 to the physical address (0:0:3c:0:39:1d).

The -f option displays and flushes the ARP table used by an interface. Flushing a table means deleting *all* table entries. The following shows the **ax\_arp** -f command and output:

```
host0> ax_arp -f -I ae1
host1-e1 (144.48.252.4) at 0:0:3c:0:39:1d
host1-e1 (144.48.252.4) deleted
host1-e2 (144.48.252.6) at 8:0:3c:0:30:10
host1-e2 (144.48.252.6) deleted
```

Flush the ARP cache if the IP-to-physical address binding for a host on the network changed and you want to update the cache quickly. In the previous example, suppose the physical address of host1-e1 changed and you flush the cache. When host0-e1 tries to reach



host1-e1, because the entry for host1-e1 is missing, host0-e1 sends an ARP request for the new address binding information without waiting for the ARP entry to time out. When host0-ae1 receives the new information, it updates the cache immediately. If you do not flush the cache after the address change, host0-e1 uses the out-of-date information for host1-e1 until the entry times out before it sends an ARP request. In this case, updating the cache might take up to several minutes.

Auspex has not modified arp(8C), which is part of SunOS and only pertains to the ARP table maintained by the Host Processor (vmunix). Because ax\_arp allows you to specify an interface number, use ax\_arp, not arp, to query the ARP table used by a selected interface.

# Displaying Network Statistics for a Network Processor

To display the contents of various network-related data structures on an Auspex network processor, use the **ax\_netstat** command.

The ax\_netstat command differs from ax\_perfmon in that statistics displayed by ax\_netstat are cumulative: they do not include the per-second rates as in the ax\_perfmon screens. Also, ax netstat can display routing tables and various networking statistics on a per-protocol basis.

The ax\_netstat commands have three forms, depending on the command line's first flag:

To show the statistics for a specified interface, enter:

```
ax netstat -i [ -n ] [ -I interface ] [ interval ]
```

The information displayed includes the MTU value, number of packets received, number of packets sent, number of errors, and number of collisions. You can display only information about interfaces that were configured and detected at boot time.

The -i flag displays the interface statistics.

The -**n** option displays IP addresses in the output instead of names.

The -I interface option specifies the interface for which statistics are displayed. These are examples of interface names: ae0, afddi1. If no interface is specified, the command displays statistics gathered from all network interfaces in the system. If the interface specified is not enabled, the interface number in the screen output is followed by an asterisk (for example, ae0\*).



**Note:** When ax netstat prints statistics for all interfaces, it may not list them in numerical order, but in the order they were detected when booting the system.

The *interval* option, specified in number of seconds, displays statistics gathered since boot time and incremental statistics for each time interval. The statistics since boot time repeat every 24 lines in the output. If no interval is specified, only one line of statistics displays to show the statistics since boot time. With the interval option, the command displays statistics continuously until you interrupt by pressing Ctrl-C. Also, the statistics are presented in different formats, depending on whether an interval is specified.



# The following example shows the output from the **ax\_netstat** -**i** command with a specified interval:

	host0> ax_netstat -i -I ae1 2									
input (ae1)			outp	ıt	inp	out	(Total)	out	tput	
	packets colls	errs	packets	errs	colls	packets	s err	s packet	s err	5
	1265620 14404	0	1941468	3 0	14404	126562	0 0	1941	468 0	
	8	0	27	0	0	8	0	27	0	0
	33	0	56	0	0	33	0	56	0	0
	18	0	14	0	0	18	0	14	0	0
	19	0	14	0	0	19	0	14	0	0
	21	0	12	0	0	21	0	12	0	0
	43	0	40	0	0	43	0	40	0	0
	22	0	13	0	0	22	0	13	0	0
	2	0	3	0	0	2	0	3	0	0
	1	0	8	0	0	1	0	8	0	0
	4	0	15	0	0	4	0	15	0	0
	9	0	9	0	0	9	0	9	0	0
	2	0	11	0	0	2	0	11	0	0
	5	0	34	0	0	5	0	34	0	0
	5	0	10	0	0	5	0	10	0	0
	2	0	2	0	0	2	0	2	0	0
	4	0	23	0	0	4	0	23	0	0
	17	0	15	0	0	17	0	15	0	0
	7	0	8	0	0	7	0	8	0	0
	4	0	2	0	0	4	0	2	0	0
	8	0	9	0	0	8	0	9	0	0
	input (ae1)		outpi	ıt	inp		(Total)		tput	
	packets	errs	packets	errs	colls	packets	errs	packets	errs o	colls
	1265857	0	1941800	0	14404	1265857	0	1941800	0 3	14404
	6	0	5	0	0	6	0	5	0	0
	1	0	2	0	0	1	0	2	0	0

The following example shows the output from the **ax\_netstat -i** command without a specified interval:

```
host0> ax_netstat -i -I ae1

Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Coll ae0 1500 192.42.160.0 192.42.160.245 1470 0 3607 0 0
```

▲ To show statistics on a per-protocol basis, enter:

```
ax_netstat -s -I interface
```

The -s flag displays statistics about the UDP, IP, and ICMP protocols gathered since boot time. The statistics pertain to the entire processor board to which the specified interface is attached. In this command form, -I *interface* is mandatory.



## The following example shows the **ax\_netstat** -**s** command and its output:

```
host0> ax_netstat -s -I ae3
udp:
        0 incomplete headers
        0 bad data length fields
        0 bad checksums
ip:
        65516383 total packets received
        0 bad header checksums
        0 with size smaller than minimum
        0 with data size < data length
        0 with header length < data size
        0 with data length < header length
        165188 fragments received
        0 fragments dropped (dup or out of space)
        911 fragments dropped after timeout
        9684974 packets forwarded
        0 packets not forwardable
        0 redirects sent
icmp:
        0 calls to icmp_error
        O errors not generated 'cuz old message was icmp
        Output histogram:
                echo reply: 35
        0 messages with bad code fields
        0 messages < minimum length</pre>
        0 bad checksums
        0 messages with bad length
        Input histogram:
                echo reply: 62
                destination unreachable: 9
                routing redirect: 489
                echo: 35
        35 message responses generated
```

#### ▲ To show the routing table on an interface, enter:

```
ax_netstat -r [ -n ] -I interface
```



The -r flag displays the routing table used on the specified interface. In this command form, -I *interface* is mandatory. The -n option displays IP addresses in the output instead of names. For more information about the routing table, refer to the ax\_netstat man page. The following is an example of an ax\_netstat -r command and its output:

host0> ax_netstat -r	-n -I ae1				
Routing tables					
Destination	Gateway	Flags	Refs	Use	Interface
127.0.0.1	127.0.0.1	UHO	0	0	
144.48.216.0	144.48.252.65	UG	0	86	ae1
192.42.160.0	144.48.254.1	UGO	0	70620	
144.48.248.0	144.48.248.4	UO	0	0	
144.48.217.0	144.48.252.65	UG	0	55	ae1
144.48.153.0	144.48.252.126	UG	0	2	ae1
144.48.9.0	144.48.9.1	UO	1	818999	)
144.48.218.0	144.48.252.65	UG	0	63	ae1
144.48.170.0	144.48.252.37	UG	0	3600	ae1
144.48.10.0	144.48.254.1	UGO	0	957	
144.48.250.0	144.48.250.4	UO	0	37980	
144.48.219.0	144.48.252.65	UG	0	60	ae1
144.48.11.0	144.48.254.1	UGO	0	244	
144.48.220.0	144.48.252.65	UG	0	61	ae1

To show routing statistics (not the routing table) for a network processor board, enter:

```
ax_netstat -r -s -I interface
```

The following is an example of an **ax\_netstat -r -s** command and its output:

```
host0> ax_netstat -r -s -I ae3
routing:

0 bad routing redirects
0 dynamically created routes
0 new gateways due to redirects
0 destinations found unreachable
14 uses of a wildcard route
```

If you do not specify the -i, -s, or -r flag in the ax\_netstat command, the flag defaults to -s and protocol information displays.

The **netstat**(8C) command, which is part of SunOS, displays information similar to that printed by **ax\_netstat**. The major differences between the commands are listed as follows:

- ▲ Statistics gathered by **ax\_netstat** pertain to packets processed by a network processor board; **netstat**, when used with the -**I** *interface* option, displays statistics for packets that are received or sent out by the interface but are processed by the HP.
- ▲ If the -I interface option is used in a netstat command and the specified interface is down or absent, nothing prints except a header. The ax\_netstat command, however, displays zero counts for various types of statistics and places an asterisk next to the interface name to show an interface that is down. If the interface is not installed in the server, an error message appears as follows:

```
ax_netstat: device "ae17" is not an installed network interface.
```



## **Displaying NFS Statistics**

The nfsstat(8C) command in SunOS displays NFS statistics such as client information and RPC information. The command is modified by Auspex to include the -I Interface option for displaying information about a selected network interface such as ae0 and afddi1. Please refer to the **nfsstat** man page. for complete **nfsstat** syntax.

The reporter.sh(8) script, when enabled, automatically mails site reports to Auspex Customer Service. The script calls **nfsstat**(8C) with the **-z** option to reinitialize **nfsstat** statistics. If you do not want the statistics fields to automatically reinitialize, edit the script /usr/auspex/reporter.sh. Comment out the nfsstat command that uses the -z option and uncomment the **nfsstat** command that does not use the -**z** option.

# **Network Changes**

Whenever you add a new network interface or change an IP address, the router daemon does not automatically pick up the change. The daemon complains with an error message similar to:

packet from unknown router

After additions or changes, always restart the daemon in.routed. See the in.routed(8C) man page for further information.





# **UNIX Manual Pages**

## **About This Appendix**

This appendix lists the online manual pages for the commands and utilities added or modified in SunOS by Auspex. Some man pages are modified because changes have been made to the software, and some are modified to follow Auspex conventions.



#### **User Commands**

**a2p**(1) Awk to perl translator

**arch**(1) Display the architecture of the current host

**ax\_dbx**(1) Auspex debugger

**ax\_sparc\_dbx**(1) Allows source level debugging and execution of programs

downloaded to Auspex SPARC-based processors

ax\_tapestats(1)
Tape drive statistics

**bar**(1) Create tape archives, and add or extract files

**cpio**(1) Copy file archives in and out

**dd**(1) Convert and copy files with various data formats

**eject**(1) Eject media device from the drive

installf(1) Adds a file to the software installation database
 mach(1) Display the processor type of the current host
 md(5) Calculate a message digest (checksum) for a file

mt(1) Magnetic tape control

**nohup**(1V) Run a command immune to hangups and quits

**pax**(1) Portable archive exchange

**perl**(1) The practical extraction and report language

pkgadd(1)Installs an optional software package from CD-ROMpkgask(1)Store answers for an interactive package installationpkgchk(1)Verifies the installation of an optional software packagepkginfo(1)Provides information on optional software packages

**pkgmk**(1) Produce an installable package for optional software installation

**pkgparam**(1) Lists parameters for **pkg** commands

**pkgproto**(1) Generates prototype file entries for **pkgmk** command

**pkgrm**(1) Removes an installed optional software package

**pkgtrans**(1) Translates an installable software package between formats

rm(1) Remove files

**removef**(1) Remove files from the software database

**rsh**(1C) Remote shell

**s2p**(1) Sed to perl translator

**stty**(1v) Set or alter the options for a terminal

tar(1) Create tape archives, and add or extract files



## **System Calls**

intro(2) Introduction to system services and error numbers

**chown**(2) Change the user and group ownership of files

mount(2v) Mount file systems

nfssvc(2) NFS daemons



#### **Devices and Networking**

acd(4) CD-ROM driver for the Auspex Storage Processor

ad(4) Disk driver for Auspex Storage Processor

**ae**(4) Auspex Ethernet interface

afddi(4) Auspex FDDI/MLT-3 interface
afe(4) Auspex Fast Ethernet interface
anp(4) Auspex Network Processor
apm(4) Auspex Primary Memory
ard(4) Auspex RAID array driver
asp(4) Auspex Storage Processor

**ast**(4) Tape driver for Auspex Storage Processor

**atm\_accept**(4) Accepts a pending connection request

atm\_bind(4) Binds an Application Service Access Point (ASAP) to an

ATM end-point

atm\_client\_aal4(4)Example client ATM program using AAL 3/4atm\_client\_null(4)Example client ATM program using null AALatm\_close(4)Closes a connection with an ATM endpoint

**atm\_connect**(4) Establishes a connection with an ATM endpoint

atm\_error(4) Prints an ATM error message

atm\_getbatchsize(4)Gets batch size for null AAL interfaceatm\_gethostbyname(4)Gets the ATM address for a hostname

**atm\_intro**(4) Provides introduction to ATM user-level library functions

atm\_listen(4) Listens for an ATM connection request

atm\_open(4) Opens an ATM connection endpoint and gets connection

information

atm\_recv(4)Receives data from an established ATM connectionatm\_send(4)Sends data over an established ATM connectionatm\_server\_aal4(4)Example server ATM program using AAL 3/4atm\_server\_null(4)Example server ATM program using null AAL

**atm\_setbatchsize**(4) Sets batch size for null AAL interface

atmio(4) ATM device control operations cdromio(4) CD-ROM control operations

compver(4)compatible versions file for software packagesdepend(4)dependencies file for software packages

**if**(4n) C Shell built-in command



iop(4) Auspex I/O Processor

lfs(4) Local file system

mtio(4) Magnetic tape interface

**pkginfo**(4) Display software package information

**pkgmap**(4) Package contents description file

nit\_if(4m) STREAMS NIT device interface module

**vp**(4) Auspex virtual partition driver



#### **File Formats**

acct(5) Execution accounting file

Auspex key and license file  $ax_keys(5)$ 

exports(5) and xtab(5)Directories to export to NFS Clients

fstab(5) File system mounting tables

**installation.report**(5) List of installation tasks performed

**mtab**(5) Mounted file system tables

passwd(5) Password file

raidtab(5) RAID array table

securenets(5) List IP addresses to which the NIS server should respond

syslog.conf(5) Configuration file for system log daemon syslogd

vpartab(5) Virtual partition table



#### **Maintenance Commands**

MAKEDEV.auspex(8) Make Auspex special files

NSconfig(8) Configure the software on an Auspex network server

NSinstall(8) Install the software on an Auspex network server

**NSupdate**(8) Auspex utility to install new releases

**SetupClient**(8) Add a diskless SunOS client to the Auspex network

server, or remove a client from the server

**SetupExec**(8) Install architecture-dependent executables on the

server

**SetupTty**(8) Configure a tty device for a modem or terminal

auspex(8) Introduction to the Auspex network servers

ax\_add\_device(8) Utility to allow disk drives to be added or replaced

while the NetServer is running

ax\_admmsg(8) UNIX-level routine to extract Auspex error messages

from the message files in /usr/adm

ax\_admmsg.check(8) UNIX-level routine that mails an error message report

to a designated list of people

ax\_arp(8) Display and flush address resolution table on an

Auspex network interface

**ax\_checklicense**(8) Checks for valid optional software product licensing

**ax\_chkdrive**(8) Auspex drive initializer

**ax clonefs**(8) Auspex command to create and configure a virtual

partition that duplicates data in another virtual

partition

**ax config**(8) Auspex utility to display and modify system

configuration information stored in EEPROM

ax\_crash(8) System crash analysis

**ax\_decyrpt**(8) Decrypt optional product files

ax\_diag(8) Auspex utility to access the PROM Extended

Diagnostics menus from UNIX

**ax\_diskconf**(8) Auspex utility to display disk configuration

information

**ax\_docs**(8) Display online system and optional product

documentation

ax\_drives(8) Auspex UNIX-level utility to obtain FRU information

stored on Auspex drives

**ax\_drstat**(8) Drive statistics maintainer



ax\_drvbuild(8) Maintains SCSI disk firmware files downloaded by

ax\_drvmaint

ax\_drvmaint(8)Downloads SCSI disk firmware filesax\_enable(8)Enables Auspex optional products

**ax\_expand**(8) Enlarge a file system on a virtual partition

ax\_fddistat(8c) Shows status of FDDI/CDDI/MLT-3 network

interfaces

**ax\_fsmgr**(8) Auspex file system manager

ax\_fsutil(8) Utility for miscellaneous file system services

ax\_hot\_plug(8) Utility for replacing multiple disk drives while the

NetServer is running

**ax\_isolated**(8) Daemon for monitoring file system isolation events

ax\_keyenvoyd(8c) Auspex daemon that talks to keyserver

ax\_kill(8) Command for killing all processes sleeping on an

isolated file system

ax\_label(8) Partition a disk and label the drive

ax\_label\_rootRepartition and write a label for a root driveax\_lfsd(8)Start LFS daemons on Auspex Host Processor

ax\_lfstest(8) Auspex LFS diagnostics

ax\_load\_flash(8) Utility to download Flash firmware

**ax\_loadvpar**(8) Auspex utility to load the virtual partition table

ax\_lslabel(8) Show a disk's label and partitioning

ax\_mattach(8) Auspex utility to attach a new member to a mirrored

virtual partition

**ax\_mconvert**(8) Convert a virtual partition to a mirrored virtual

partition

**ax\_mdetach**(8) Auspex utility to detach a member from a mirrored

virtual partition

**ax\_meminfo**(8) Prints the amount and type of primary memory

ax\_memtest(8) VME data transfer test

**ax\_mrestore**(8) Auspex utility to copy data from one member of a

mirrored virtual partition to the other member

ax\_netstat(8c) Utility to obtain network statistics from Auspex

**Ethernet Processors** 

ax\_nfsd(8) Start NFS daemons on Auspex Network Processors

ax\_perfhist(8) Auspex command to display graphically the statistics

gathered by **ax\_perfmon** 

**ax\_perfmon**(8) Auspex performance monitoring tool



ax\_ping(8) Send null FMK messages to an Auspex processor

**ax\_purgelock**(8) Releases remote locks held by the server for a specified

Internet address

**ax\_remove\_device**(8) Utility to allow disk drives to be removed while the

NetServer is running

ax\_request\_key(8) Generate email or a file requesting an Auspex optional

product key

**ax\_resolve**(8) Find the FMK process with the specified name

**ax\_restore\_mirrors**(8) Restores all mirrored partitions to an active state

**ax\_set\_timeout**(8) Specify how long the system waits before declaring a

board dead

**ax\_sperr** Storage Processor error description program

ax\_sputil(8) Auspex utility for various disk-related functions

ax\_startup(8) Auspex daemon that downloads and boots Auspex

processors

ax\_statd(8) and Auspex statistics daemons

ax\_statd2(8)

**ax\_statfp**(8) Utility to report statistics for a specified Auspex File

**Processor** 

ax\_statm16(8) Utility to report M16 statistics for a specified Auspex

Processor

ax\_systest(8) Auspex system stress test suite

**ax\_systest.setup**(8) Generate an Auspex system test suite environment file

template

**ax\_tcbt** Test 100Base-T Fast Ethernet port on NP boards

ax\_tcbt2 Test Sun Fast Ethernet Adapter 2.0 on NP boards

**ax\_tcp**(8) Add NP TCP workers, show status of existing workers,

drop connections

ax\_tfddi Test FDDI or CDDI port on NP boards

**ax\_timed**(8) and **ax\_errd**(8) Daemon that reports and logs errors from the Auspex

processors

**ax\_trace** Extracts, formets, or configures Auspex processor

traces

**ax\_util**(8) Auspex utility used to perform board-level debugging

on the Auspex processors

**ax\_vold**(8) Auspex virtual partition daemon

**ax\_vpstat**(8) Auspex utility to display the status of virtual partitions

**ax\_whatrel**(8) Utility to print the release number of files that are part

of Auspex releases



**ax\_write\_cache**(8) Control the SP write acceleration feature

**boot**(8S) Start the system kernel or a standalone program

config(8) Build system configuration files

**corescreen**(8), **corescreen.sh**(8) Gather information from Auspex core dumps

diskusg(8) Generate disk accounting data by user

**dkinfo**(8) Report information about disk geometry and

partitioning

dmesg(8) Collect system diagnostic messages to form error log

dump(8)Incremental file system dumpdumpfs(8)Dump file system informationeeprom(8s)EEPROM display and load utility

exportfs(8) Export and unexport directories to NFS clients

fastboot(8) and Reboot or halt the system without checking disks

fasthalt(8)

**fastfs**(8) Sets the delayed metadata write capability of local file

systems

**fsck**(8) File system consistency check and interactive repair

**getcores**(8), **getcores.sh**(8) Compress and **tar** core files to tape

growfs(8) Enlarge a file system

halt(8) Stop the Host Processor

**hpboot**(8S) Start the system kernel or a standalone program

hphalt(8) Stop the processor

**hpreboot**(8) Restart the operating system

hpshutdown(8)Close down the system at a given timeicheck(8)File system storage consistency check

**ifconfig**(8c) Configures network interface parameters

inetd(8c) Internet services daemon

installboot(8S) Install bootblocks in a disk partition

looptest(8c) Tests an ATM interface for cell reception and

transmission

mailmessages(8), Inform Auspex Customer Service of a reboot

mailmessages.sh(8)

mkfs(8) Construct a file system

**mount**(8) Mount and unmount file systems

mountd(8) NFS mount request server

netstat(8c) Show network status for Auspex Host Processor



newfs(8) Create a new file system

NFS daemon nfsd(8)

nfsstat(8c) Network file system statistics

panic(8s) Explains what happens when the system crashes rc(8), rc.auspex(8), Scripts for autoreboot, shutdown, and daemons

rc.boot(8), rc.local(8),

rc.shutdown(8)

reboot(8) Restart the kernel

reporter(8), reporter.sh(8) Prepare and mail site reports to Auspex Customer

Service

restore(8), rrestore Incremental file system restore route(8c) Manipulate the routing tables

savecore(8) Save core dumps of the operating system shutdown(8) Close down the system at a given time stopnfsd(8) Auspex utility to shut down the NFS

sundiag(8) System diagnostics

swapon(8) Specify additional device for paging and swapping

ypserv(8), ypbind, ypxfrd NIS server and binder processes



## **Online Documentation**

## **About This Appendix**

This appendix provides information on installing online documentation and setting up your workstation's X environment to view the online documents.



#### **Online Documentation**

Auspex offers online documentation to users who have an X display that runs the X Window System protocol (MIT X.11, Release 4 or later), or that runs Sun OpenWindows 2.0 or later. The online documentation provides hypertext links embedded in the text, allow you to move easily from topic to topic within the manual. Hypertext links are text in the online documents.

This section covers the following topics:

- ▲ Installing online documentation
- ▲ Starting the online documentation
- ▲ Overview of online documentation
- ▲ Printing a document

#### **Installing Online Documentation**

All of the online documentation is on the Auspex Premier Software Series CD-ROM in a package named AXdocs. The package includes the system manuals, hardware manuals, and Optional Product manuals. Use the **pkgadd**(1M) command to install the online documents.



**Note:** The **pkgadd** process sets userid to "install" (or root if there is no install userid) before adding a package. If your system has an install userid that does not have super user permissions, then the **pkgadd** may fail. Either give the install userid super user permissions, or remove the install userid.

Before adding the package, you must find enough space for the package in a directory. The **pkgadd** command calculates the size directory needed and displays the size. In the following procedure, the size given is 27,000 kilobytes, but the actual size for your installation may be larger. Find a directory of at least 27,000 kilobytes, then use the **pkgadd** command to get the actual size needed. You may need to do a second pass with **pkgadd** if you need to find a larger target directory.

1. Mount the Premier Software Series CD-ROM; for example:

```
# mount -rt hsfs /dev/acd1 /cdrom
```

This command mounts the CD in drive slot 1 on /cdrom.

2. Mount /usr with read/write privilege:

```
# mount -o remount,rw /usr
```

3. Add the document package to the server by entering the following command:

```
# pkgadd -d /cdrom
```

The system returns:



```
(HP-VII, HP-VIII) 1.10
   AXatm2
                  (HP-VII, HP-VIII) 1.10
 5 AXbackup
                  FastBackup
                  (HP-VII, HP-VIII) 1.10
   AXdgrd
                  DataGuard
                  (HP-VII, HP-VIII) 1.10
 7 AXdocs
                  Auspex System Documentation
                  (HP-VII, HP-VIII) 1.10
   AXdrvgrd
                  DriveGuard
                  (HP-VII, HP-VIII) 1.10
 9 AXftp
                  NP Resident FTP
                  (HP-VII, HP-VIII) 1.10
 10 AXsrvgrd
                  ServerGuard
                  (HP-VII, HP-VIII) 1.10
Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

4. Enter the number **5** for the 1.10 document package.

Processing package instance <AXdocs> from </cdrom>

```
Auspex System Documentation (HP-V,HP-VI,HP-VII,HP-VIII) 1.10
```

5. The following question needs a path to the destination directory for the document package, not a path to the source on the CD. Enter a question mark.

Enter path to package base directory [?,q] ?
Installation of this package requires that a UNIX directory be available for installation of appropriate software. This directory may be part of any mounted filesystem, or may itself be a mount point.

In general, it is unwise to select a base directory which already contains other files and/or directories.

6. Now enter the path name to the destination directory for the package.

Enter path to package base directory [?,q] /docs Using </docs> as the package base directory.

AXdocs requires the following disk space: 11572 kbytes

Checking if /docs has enough free space for install...

```
## Processing package information.
```

## Checking for setuid/setgid programs.

This package contains scripts which will be executed with superuser permission during the process of installing this package.

Do you want to continue with the installation of this package [y,n,?]

<sup>##</sup> Processing system information.

<sup>##</sup> Verifying disk space requirements.

<sup>##</sup> Checking for conflicts with packages already installed.



7. Answer **y** if you wish to continue. Answer **n** if you want to find another source directory for the package.

```
Installing Auspex System Documentation as <AXdocs>
## Installing part 1 of 1.
/documents/FSG/17_FRU.PDF
/documents/FSG/181_FRU.PDF
.
```

.

A long listing of files is deleted in this example, but will appear during your installation.

```
[ verifying class <none> ]
## Executing postinstall script.
ln -s /docs /usr/auspex/docs
Installation of <AXdocs> was successful.
The following packages are available:
 1 AXEC1
            etherchannel
                 (HP-VII, HP-VIII) 1.10
 2 AXNTBios
                 Auspex NetBIOS
                 (HP-VII, HP-VIII) 1.10
 3 AXNeTsrv
                Advanced Server for UNIX Systems
                 (HP-VII, HP-VIII) 1.10
 4 AXatm2
                 ATM 2
                 (HP-VII, HP-VIII) 1.10
 5 AXbackup
                FastBackup
                 (HP-VII, HP-VIII) 1.10
 6 AXdgrd
                 DataGuard
                 (HP-VII, HP-VIII) 1.10
   AXdocs
                 Auspex System Documentation
                  (HP-VII, HP-VIII) 1.10
 8 AXdrvgrd
                 DriveGuard
                 (HP-VII, HP-VIII) 1.10
 9 AXftp
                 NP Resident FTP
                  (HP-VII, HP-VIII) 1.10
 10 AXsrvgrd
                 ServerGuard
                  (HP-VII, HP-VIII) 1.10
Select package(s) you wish to process (or 'all' to process
```

8. Enter **q** to quit the **pkgadd** routine. Note that the **pkgadd** command copies the session into two log files as shown in the following display.

```
pkgadd session finished on Tue Jan 14 15:28:50 PST 1997
```

all packages). (default: all) [?,??,q]:

A log of this session of pkgadd is at /tmp/pkgadd.log on the current root disk, and at /var/log/pkgadd.log on the target disk.



#### Starting the Online Documentation

To open the online document from an X display on the network, use the following procedure. In the procedure, the machine foobar has the local X display screen, and the server dochost1 holds the **ax\_docs**(8) utility and files.

- 1. Verify your path contains the path entry /usr/auspex.
- 2. Add the server dochost1 to your xhost list.

```
foobar% xhost +dochost1
```

3. Use  $\mathbf{rlogin}(1C)$  or  $\mathbf{tip}(1C)$  to login to dochost1.

```
foobar% rlogin dochost1
dochost1%
```

4. From dochost1, set the display environment for the shell type. For example, for the C shell, enter the following command to send the display back to foobar:

```
dochost1% setenv DISPLAY foobar:0
```

For the Bourne shell or Bourne-compatible shell, follow this example:

```
dochost1% DISPLAY=foobar:0; export DISPLAY
```

5. Enter the following on dochost1 to start the online documentation utility:

```
dochost1% ax docs
```

This script invokes the documentation Main Menu.

#### Overview

Auspex NetServers support online documentation through the licensing of Acrobat software from Adobe Corporation. The online documentation contains system hardware, system software, and optional software information. The software information is in the online version of the System Manager's Guide; the hardware information is extracted from the hardware manual that ships with your server.

Figure B-1 shows part of the Main menu. Click on a menu choice to see documentation.

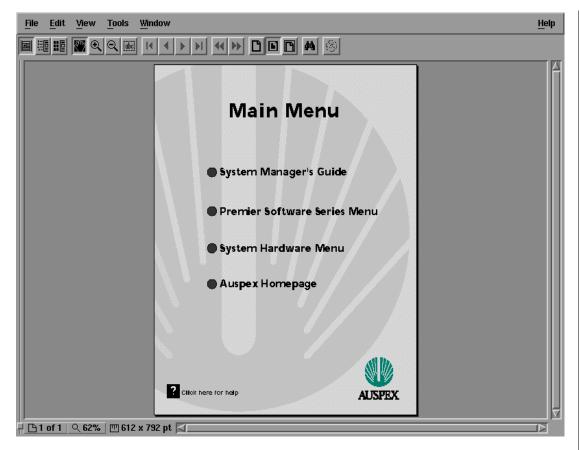


Figure B-1. Main Menu window (for Auspex online documentation)

The entries on the Main Menu are:

- ▲ System Manager's Guide. This entry brings up a window with the *System Manager's Guide*.
- ▲ Premier Software Series Menu. This entry brings up another menu with choices for all of the optional products offered with this release. Clicking on one of the optional product choices brings up a window with the document for that product.
- ▲ System Hardware Menu. This entry brings up another menu with entries for all the hardware manuals and the Hardware Release Note. Click on the entry for your system (such as the NS 7000 Model 700 Series) to see the manual for that system.
- ▲ Field Service Menu. This entry brings up a menu of field service documents, including FRU documents and the *Field Service Guide*.
- ▲ Auspex Homepage. This entry connects to the Auspex home page over the World Wide Web, provided you have an Internet browser running on your system.
- ▲ Help button. The help button, a large question mark, explains how to use the Acrobat Reader window menus and icons

Using the *System Manager's Guide* as an example, note that when the window containing that document appears, a table of contents sidebar entry appears in the window. The entries are called bookmarks. A bookmark with more levels of table of content entries can be expanded to other levels by clicking on the arrow for that entry. Clicking on the entry



itself takes you to that part of the document. Figure B-2 shows a portion of the System Manager's Guide window with the bookmarks on the left. The Main Menu choice takes you back to the documentation Main Menu.

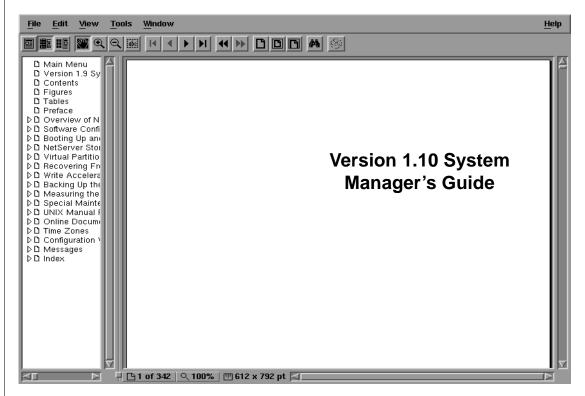


Figure B-2. Bookmarks

#### **Printing a Document**

For your convenience, you can print individual pages or the entire contents of the online manual. Bring down the File menu from the top of any of the windows, and choose the print option. The Premier Software Series CD also has a directory named *psdocs* that holds PostScript files for each document. You can print the documents directly from psdocs. To print documents in A4 format, use the files with the ".psA4" suffix.



# C

# **Time Zones**

#### **World Time Zones**

This appendix contains a listing of world time zones. A time zone name from this table is required during the NetServer configuration process.

Table C-1. Time zone listing

Time zone name	Time zone area
North America:	
U.S./Eastern	Eastern time zone, U.S.
U.S./Central	Central time zone, U.S.
U.S./Mountain	Mountain time zone, U.S.
U.S./Pacific	Pacific time zone, U.S.
U.S./Pacific-New	Pacific time zone, U.S., with proposed changes to daylight saving time near election time in presidential election years.
U.S./Yukon	Yukon time zone, U.S.
U.S./East-Indiana	Eastern time zone, U.S., no daylight saving time
U.S./Arizona	Mountain time zone, U.S., no daylight saving time
U.S./Hawaii	Hawaii
Navajo	Same as U.S./Mountain
Canada/Newfoundland	Newfoundland
Canada/Atlantic	Atlantic time zone, Canada
Canada/Eastern	Eastern time zone, Canada
Canada/Central	Central time zone, Canada
Canada/East-Saskatchewan	Central time zone, Canada, no daylight saving time
Canada/Mountain	Mountain time zone, Canada
Canada/Pacific	Pacific time zone, Canada
Canada/Yukon	Yukon time zone, Canada
Mexico/BajaNorte	Baja Norte time zone, Mexico
Mexico/BajaSur	Baja Sur time zone, Mexico



Table C-1. Time zone listing (Continued)

Time zone name	Time zone area		
Mexico/General	Mexico general time zone		
South America:			
Brazil/East	Eastern Brazil time zone		
Brazil/West	Western Brazil time zone		
Brazil/Acre	Territory of Acre time zone, Brazil		
Brazil/DeNoronha	Fernando De Noronha time zone, Brazil		
Chile/Continental	Continental Chile time zone		
Chile/Easter Island	Easter Island time zone, Chile		
Europe:			
GB-Eire	Great Britain and Ireland		
WET	Western European time		
Iceland	Iceland		
MET	Middle European time (same as CET)		
CET	Central European time (same as MET)		
Poland	Poland		
EET	Eastern European time		
Turkey	Turkey		
W-SU	Western Soviet Union		
Asia:			
PRC	People's Republic of China		
ROK	Republic of Korea		
Japan	Japan		
Singapore	Singapore		
Hongkong	Hong Kong		
ROC	Republic of China		
Israel	Israel		

#### Australia and New Zealand:

Australia/Victoria

Australia/Tasmania Tasmania, Australia
Australia/Queensland Queensland, Australia
Australia/North Northern Territory, Australia
Australia/West Western Australia
Australia/South South Australia

Victoria, Australia



Table C-1. Time zone listing (Continued)

Time zone name	Time zone area
Australia/NSW	New South Wales, Australia
Australia/Broken-Hill	Broken-Hill, Australia
Australia/Sturt	Sturt, Australia
Australia/Yancowinna	Yancowinna, Australia
Australia/LHI	Lord Howe Island, Australia
NZ	New Zealand



Other time zones (none include daylight saving time unless otherwise noted):

Table C-2. Additional time zone listings

Time Zone Name	Time Zone Area
GMT	Greenwich mean time
UTC	Same as GMT
UCT	Same as GMT
Universal	Same as GMT
Greenwich	Same as GMT
GMT-1	1 hour west of Greenwich mean time
GMT-2	2 hours west of Greenwich mean time
GMT-3	3 hours west of Greenwich mean time
GMT-4	4 hours west of Greenwich mean time
GMT-5	5 hours west of Greenwich mean time
GMT-6	6 hours west of Greenwich mean time
GMT-7	7 hours west of Greenwich mean time
GMT-8	8 hours west of Greenwich mean time
GMT-9	9 hours west of Greenwich mean time
GMT-10	10 hours west of Greenwich mean time
GMT-11	11 hours west of Greenwich mean time
GMT-12	12 hours west of Greenwich mean time
GMT+13	13 hours west of Greenwich mean time (GMT+12 with daylight saving time)
GMT+12	12 hours east of Greenwich mean time
GMT+11	11 hours east of Greenwich mean time
GMT+10	10 hours east of Greenwich mean time
GMT+9	9 hours east of Greenwich mean time
GMT+8	8 hours east of Greenwich mean time
GMT+7	7 hours east of Greenwich mean time
GMT+6	6 hours east of Greenwich mean time
GMT+5	5 hours east of Greenwich mean time
GMT+4	4 hours east of Greenwich mean time
GMT+3	3 hours east of Greenwich mean time
GMT+2	2 hours east of Greenwich mean time
GMT+1	1 hour east of Greenwich mean time



# **Configuration Worksheets**

#### **About This Appendix**

This appendix contains a set of four worksheets for system configuration tasks:

- ▲ The Host Information worksheet compiles information for host configuration with the **NSconfig** command and is useful when using the **NSinstall** command.
- The Client Information worksheet compiles information for client configuration with the **SetupClient** commands.
- The Disk Drive Information worksheet may be helpful during disk partitioning.
- The Virtual Partition Information worksheet compiles information about the virtual partitions defined on your server.

These worksheets are designed to be duplicated for your convenience.



#### **Host Information Worksheet**

#### **Network Information:**

Primary Hostname:

NIS Type: (Master/Slave/Client/None)

NIS Domain:

SNMP Daemon: (Yes/No)

Route Daemon: (Yes/No/Quiet)

Time Zone:

#### **Host Information:**

	Host name	Internet Address	Subnet Mask	Ethernet Address
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
22				
23				



Notes:	



### **Client Information Worksheet**

Server Informati	ion:			
Hostname:	·	NIS Domain:		
Client Informati	on:			
Client Name	Arch.	SunOS Version	Swap Size	Addresses, Names, and Paths
				Ethernet addr:
Notes:				Internet addr:
				Domain name:
				Export path:
				Root path:
				Swap path:
				Executables path:
				Kernel executables path:
				Home path:
				/usr/share files path:
				Ethernet addr:
Notes:				Internet addr:
				Domain name:
				Export path:
				Root path:
				Swap path: Executables path:
				Kernel executables path:
				Home path:
				/usr/share files path:



Arch.	SunOS Version	Swap Size	Addresses, Names, and Paths
			Ethernet addr:
			Internet addr:
			Domain name:
			Export path:
			Root path:
			Swap path: Executables path:
			Kernel executables path:
			Home path:
			/usr/share files path:
			Ethernet addr:
			Internet addr:
			Domain name:
			Export path:
			Root path:
			Swap path: Executables path:
			Kernel executables path:
			Home path:
			/usr/share files path:
	Arch.		



### **Disk Drive Information Worksheet**

Drive Type Information:
Disk Drive no. (slot number):
Drive size (1 GB, 1.35 GB, 1.76 GB, 2 GB, 3 GB, 4 GB, 9 GB):
Drive Partitioning:
Drive partitioning scheme (root, default, stripe, other):

Partition	Size	Mount Point	Major Directories
a			
b			
С			
d			
е			
f			
g			
h			

**Notes:** 



## **Virtual Partition Information Worksheet**

**Virtual Partition Information:** 

VP No.	Members	Size	Type	Mount Point	Major Directories

**Notes:** 



# E

# Messages

#### **About This Appendix**

This appendix describes messages unique to the NetServer. Auspex NetServer messages fall into the following categories:

- ▲ Standard boot messages
- **▲** System startup error messages
- ▲ Auspex processor error messages

This appendix makes no attempt to document standard Solaris messages. For information about Solaris messages, refer to Appendix B in Sun's *System and Network Administration*. Also, you can refer to the **Intro**(2) man page for an introduction to SunOS system calls and error numbers.

#### **Standard Boot Messages**

This section provides an example of boot messages that display as part of a normal boot process. The boot sequence is annotated with circled numbers that refer to the boot event explanations in Table E-1.

The example is for a NetServer with an HP VII. NetServers with an HP VIII have a similar boot sequence.

```
IDPROM checksum test.
SPARC Processor (10-0070) Boot PROM Selftest.
POST RAM (Clear)
Setup MMU table
Copying PROM Data to RAM
EPROM Checksum Test
                                                                    2
Sys Control Register Test
SW Interrupt 1 (level 1) Test
SW Interrupt 2 (level 6) Test
MPC Register Test
ECC Valid Access Test
Timer 0 Interrupt (level 10) Test
Timer 1 Interrupt (level 14) Test
Command Ready Interrupt Test
MMU PTE Reference Bit Test
MMU PTE Modified Bit Test
MMU PTE Valid Bit Test
MMU PTE Access Bit Test
MMU TLB Test
MMU Data Access Trap Test
MMU Probe Test
Cache Data RAM Test
Cache Tag RAM Test
Cache Write Miss Test
Cache Write Hit Test
NVRAM Battery Test GOOD BATTERY done
NVRAM Memory Test saving restoring restoring checksum done NVRAM
Test, checksum=0x77
TOD Test127 119 95 127 123 127 95 127
POST complete: Pass = 1, Errors = 0
Auspex NetServer: SPARC
96MB memory installed, Hostid ffffffff
Part Number 10-0107, Rev CD, Serial #3
AUSPEX-VERSION 2.9_HP SUNMON 09/12/96 10:58
Initialize 96 MB of Memory ... Completed.
Test 0 MB of Memory (Address Test) ... Completed.
```



```
EEPROM boot device...ad(0,0,0)
sp - status 0 70 71 f0
sp found in slot = 12
Boot device: /mvic/vme/asp@6d,1180/ad@0,0:0 File and args:
0x4000
bootblock loaded
sp - status f0
sp - status f0
root on ad0a fstype 4.2
Boot: vmunix
Size: 1122304+229176+85512 bytes
m16u\_npages\_dvma = 358
m16u_net_interfaces = 15
VAC ENABLED in COPYBACK mode
Auspex 1.9/SunOS 4.1.4 (AUSPEX1) #1: Tue Nov 5 04:41:40 PDT 1996
Copyright 1989-1996 Auspex Systems, Inc.
Copyright 1983-1994 Sun Microsystems, Inc.
cpu = ASPX, Auspex SPARC Processor
mod0 = Ross, RT625 (mid = 2)
mem = 98304K (0x10000000)
avail mem = 88797184
dma0 at SBus slot 1 0x400000
esp0 at SBus slot 1 0x800000 pri 3
zs0 at obio 0x200000 pri 12
SPO at Auspex VME slot 12 VME address 0x0
SP1 at Auspex VME slot 13 VME address 0x0
SP2 at Auspex VME slot 14 VME address 0x0
IOPO at Auspex VME slot 5 VME address 0x10000000 memory 256 MB
IOP1 at Auspex VME slot 6 VME address 0x20000000 memory 256 MB
IOP2 at Auspex VME slot 7 VME address 0x30000000 memory 256 MB
ad0: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
                                                                   12
acd1: <TOSHIBA CD-ROM XM-3301TA (0272)>
ad2: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad3: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad4: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad5: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad6: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad7: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad8: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad9: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad10: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad11: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad12: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad13: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad14: <Seagate 4GB cyl 4094 alt 1 hd 16 sec 128>
ad15: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ad16: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ad17: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
```



```
ad18: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ad19: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ad20: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ad21: <MICROP 4GB cyl 4094 alt 1 hd 16 sec 128>
ASPX, buf-e unit 0 at iop0 SBus slot 1
                                                                    13
ASPX, buf-e unit 1 at iop0 SBus slot 2
ASPX, buf-e unit 2 at iop0 SBus slot 3
ASPX, net-acc unit 0 at iop0 SBus slot 4
ASPX, buf-e unit 3 at iop1 SBus slot 1
ASPX, buf-e unit 4 at iop1 SBus slot 2
ASPX, buf-e unit 5 at iop1 SBus slot 3
ASPX, net-acc unit 1 at iop1 SBus slot 4
root on ad0a fstype 4.2
swap on ad0b fstype spec size 409600K
dump on ad0b fstype spec size 409588K
checking root and /usr filesystems
/dev/rad0a: 9009 files, 20571 used, 38521 free
/dev/rad0a: (249 frags, 4784 blocks, 0.4% fragmentation))
/dev/rad0q: is stable.
rc.auspex: Running ax_startup (download boards and start daemons)
running ax chkdrive...
ax write cache: SPO initializing write cache.
                                                                    15
ax_write_cache: SP1 initializing write cache.
ax_write_cache: SPO enabling write cache.
ax_write_cache: SPO write cache is 'ON'.
ax write cache: SP1 enabling write cache.
ax_write_cache: SP1 write cache is 'ON'.
rc.auspex: Running from rc.boot (start virtual partitions).
/dev/rad2c: is clean.
/dev/rad0f: is clean.
/dev/rad0d: is clean.
/dev/rvp3: is clean.
/dev/rvp6: is clean.
/dev/rvp9: is clean.
Fri Nov 8 14:31:21 PDT 1996
rc: mounting 4.2 and lfs file systems.
checking quotas: done.
rc.auspex: Running from rc (start more auspex daemons).
starting rpc port mapper.
starting NIS services: ypserv ypxfrdypbind -ypsetme: allowing local
ypset!
(this is insecure)
ypbind ypupdated.
starting RPC key server.
starting Auspex key daemon.
network interface configuration:
                                                                    16
ae0: flags=43<UP, BROADCAST, RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a7:90
ae1: flags=43<UP, BROADCAST, RUNNING>
```



```
inet 123.45.678.1 netmask ffffff00 broadcast 123.45.6784.0
        ether 0:0:3c:0:a7:91
ae2: flags=43<UP,BROADCAST,RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a7:92
ae3: flags=43<UP, BROADCAST, RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a7:93
ae4: flags=43<UP, BROADCAST, RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a7:94
ae5: flags=43<UP, BROADCAST, RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a7:95
ae6: flags=43<UP,BROADCAST,RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a6:b0
ae7: flags=43<UP, BROADCAST, RUNNING>
        inet 123.45.678.1 netmask ffffff00 broadcast 123.45.678.0
        ether 0:0:3c:0:a6:b1
running routing daemon.
running SNMP agent.
mount -vat nfs
starting additional services: biod.
starting system logger
starting local daemons: auditd sendmail xntpd ax_lfsd nfsd ax_nfsd
rarpd bootparamd statd lockd.
link-editor directory cache
ax isolated
checking for crash dump...
preserving editor files
clearing /tmp
standard daemons: update cron uucp.
starting network daemons: inetd printer xdm.
Fri Nov 8 14:31:53 PDT 199
hostname login:
```



Table E-1. NetServer boot event explanation

Ref. number	Explanation
1	Boot process initiated by Host Processor EPROM
2	Power-on self tests running in RAM
3	Host Processor PROM banner
4	Memory initialization and testing
5	EPROM boot messages
6	Software copyright and version information
7	Host Processor type
8	Host Processor memory size
9	Probe of HP SCSI bus
10	Serial port autoconfigured
11	IOP boards (NP boards) and SP boards detected and autoconfigured
12	Labeled disk drives detected and autoconfigured (an unlabeled disk drive does not appear in the list)
13	SBus cards on the IOP boards detected
14	Initiate start-up sequence for Auspex processor boards
15	Write Accelerator initialization
16	Configuring network interfaces
17	Host login prompt



## System Startup Error Messages

System startup error messages may be received from the PROM monitor or from boot programs.

```
SP: VME STATUS REGISTER ACCESS FAILED
```

There is a problem accessing the SP. Check to be sure the board is seated properly. If this fails to resolve the problem, contact your Auspex representative for assistance.

```
SP: FAILED INTERNAL DIAGNOSTIC CHECK
```

There is a problem with the SP. Contact your Auspex representative for assistance.

```
SP: FAILED INTERNAL LOOPBACK CHECK
```

The HP cannot communicate with the SP. Check to be sure the board is seated properly. If this fails to resolve the problem, contact your Auspex representative for assistance.

```
SP: BOOT DRIVE READ LABEL FAILED
```

The SP was unable to read the label from the boot drive. Check the SCSI cables and connectors. Try replacing the SCSI cable connecting the SP to the boot drive. Check the boot drive. Try starting up the system with a different boot drive.

```
No label found - attempting boot anyway.
```

The label on the boot disk was read correctly but contained bad information. Possibly a problem exists with the installation procedure. Try reinstalling the software on the boot drive. Try replacing the boot drive.

```
Corrupt label
```

The label on the boot disk was read correctly but contained bad information. A problem may exist with the installation procedure. Try reinstalling the software on the boot drive. Try replacing the boot drive.

The following messages may also appear. These messages give trained service personnel more detailed information about the fault that occurred.

```
sp: hard reset
sp - status X X X X
sp loopback: err code X
sp_msg: err_codes X X X X
sp: 16 retries
sp_scsi_cmd_error: err_code X
boot_simple_send: TIMEOUT
```

## **Auspex Processor Error Messages**

Auspex processor error messages appear in the system log in the format:

```
processor type and no.: error no.: message text
```

For example, for error 1209 on the first SP board, the message is:

```
SPO: 1209: Recovered, drive seek failure, slot 10, drive 1.
```

In the example, SP0 is the processor type and number, 1209 is the error number, and the remainder is the message text.

If you need to report an error message to your service representative, please have all relevant information available, including the entire message recorded in the system log, the state of the server and network, and the type of activity occurring at the time of the error.

Table E-2 provides the following information for each message:

- ▲ Error number
- ▲ Priority code, which is a number in the range 1–3. The following list explains the meaning of each number:
  - System panic or isolated file system: report these messages to your Auspex service representative immediately.
  - 2 Recovered errors: report these messages only if they recur frequently.
  - Notifications: these messages are generated by routine conditions, such as user errors or recovered errors on a peripheral device.
- ▲ Message text. In the table, the following conventions are used to represent variables:
  - %d a decimal number (variable length)
  - %x a hexadecimal number (variable length)
  - %*n*x an *n*-digit hexadecimal number (in the actual error message, *n* is a decimal number)
  - %s a variable-length string



Table E-2. Auspex Processor Error Messages

Error	Priority code	Message text	
1001	1	cm_bread: size 0	
1002	1	cm_bread	
1003	1	cm_breada	
1004	1	cm_breadrabp	
1005	1	cm_bwrite	
1006	1	cm_getblk: size too big	
1007	1	cm_geteblk: size too big	
1008	1	cm_brealloc	
1009	1	cm_alloc: bad size	
1010	3	file system full	
1011	3	out of inodes	
1012	3	bad block	
1013	1	ialloc: bad size	
1014	1	realloccg: bad size	
1015	1	realloccg: bad bprev	
1016	3	%s: optimization changed from SPACE to TIME	
1017	3	%s: optimization changed from TIME to SPACE	
1018	1	realloccg: bad optim	
1019	1	ialloc: dup alloc	
1020	1	Out of memory in structure allocation.	
1021	1	Value in fc_conf.bufhash must be a power of two	
1022	1	Value in fc_conf.cm_bufhash must be a power of two	
1023	1	Value in fc_conf.inodehash must be a power of two	
1024	1	Value in fc_conf.ndquothash must be a power of two	
1025	1	Value in fc_conf.ncachehash must be a power of two	
1026	1	User requested File Processor panic from console	
1027	3	fc_read: attempt to read from non-file	
1028	3	fc_readlink: attempt to readlink from non-link	
1029	1	getdiskquota	
1030	3	dquot: is full	
1031	3	inode: table is full	
1032	1	diskquota	
1033	1	dqput	
1034	1	dqupdate	
1035	1	dqinval	



Table E-2. Auspex Processor Error Messages (Continued)

	Auspex Processor Error Messages (Continued)		
Error	Priority code	Message text	
1036	1	closedq	
1037	3	free inode %s/%d had %d blocks	
1038	1	alloccgblk: cyl groups corrupted	
1039	1	alloccgblk: can't find blk in cyl	
1040	1	ialloccg: map corrupted	
1041	1	ialloccg: block not in map	
1042	1	free: bad size	
1043	1	free: freeing free block	
1044	1	free: freeing free frag	
1045	1	ifree: range	
1046	1	ifree: freeing free inode	
1047	1	alloccg: map corrupted	
1048	1	alloccg: block not in map	
1049	1	direnter	
1050	1	direnter: target directory link count	
1051	2	%s: bad dir ino %d at offset %d: %s	
1052	1	dirprepareentry: new block	
1053	1	DIRBLKSIZ > fsize	
1054	1	dirprepareentry: invalid slot status	
1055	1	dirmakeinode: no attributes	
1056	1	direnter: dquot	
1057	1	DIRBLKSIZ > fsize	
1058	1	dirremove	
1059	1	iget: bad dev	
1060	1	iget: bad fs	
1061	1	free inode isn't	
1062	1	iput	
1063	1	irele	
1064	1	idrop	
1065	1	iinactive	
1066	1	itrunc: newspace	
1067	1	itrunc1	
1068	1	itrunc2	
1069	1	update: ro fs mod	
1070	1	isblock	
1071	1	cirblock	



Table E-2. Auspex Processor Error Messages (Continued)

		,	
Error	Priority code		
1072	1	setblock	
1073	1	getmp: bad magic	
1074	1	ufs_statfs	
1075	1	ufs_badvfsop: Shouldn't be called.	
1076	1	rwip	
1077	1	rwip type	
1078	1	ufs_readdir: dir reclen == 0!	
1079	1	ufs_readdir: dir reclen > 512!	
1080	1	ufs_badop	
1081	1	vn_rele	
1082	1	vfs_unlock	
1083	1	bread: size 0	
1084	1	bread	
1085	1	breada	
1086	1	breadrabp	
1087	1	bwrite	
1088	1	getblk: size too big	
1089	1	geteblk: size too big	
1090	1	brealloc	
1091	1	dup biodone	
1092	1	binval_vp_ctrl: not implemented.	
1093	1	dnlc_purge: zero vp	
1094	3	dnlc_purge1: no entries to purge	
	1	ufs_readdir_nfs: dir reclen == 0 !	
1095	2	Warning: Proc 0x%x received bogus message	
	1	ufs_readdir_nfs: dir reclen > 512 !	
1096	1	sleep: chan == 0.	
1097	1	I_subr_on_stack: Stack overflow for func	
1098	1	I_stack_depth_used: Stack overflow detected	
1099	2	nfs_server: bad proc number	
1100	2	nfs_server: bad version number	
	1	ufs_readdir_nfs: dir reclen not word aligned!	
1101	2	nfs_server: Can't allocate args struct	
	1	ufs_readdir_nfs: dir reclen not word aligned!	
1102	2	nfs_server: bad getargs	
1103	2	nfs_server: weak authentication, source IP address=%s	



Table E-2. Auspex Processor Error Messages (Continued)

Error	Priority code	Message text	
	1	dirlook: dir reclen not work aligned!	
1104	2	nfs_server: Can't allocate results struct	
	1	dircheckforname: dir reclen not word aligned!	
1105	2	nfs_server: bad freeargs	
	1	ufs_readdirplus: dir reclen not word aligned!	
1106	2	nfs_server: bad sendreply	
	1	ufs_getdirattrs: dir reclen == 0 !	
1107	2	NFS request from unprivileged Channel.	
	1	ufs_getdirattrs: dir reclen > 512 !	
1108	1	DMA channel %d: transfer error %d	
	1	ufs_getdirattrs: dir reclen not word aligned!	
1113	2	ae%d: resetting lance!	
1114	2	ae%d: No carrier - transceiver cable problem?	
1115	2	ae%d: Transmit retried more than 16 times	
1116	2	ae%d: Transmit late collision - net problem?	
1117	2	xdr_long: FAILED	
1118	2	xdr_u_long: FAILED	
1119	2	xdr_u_short: decode FAILED	
1120	2	xdr_u_short: bad op FAILED	
1121	2	xdr_bool: decode FAILED	
1122	2	xdr_bool: bad op FAILED	
1123	2	xdr_opaque: decode FAILED	
1124	2	xdr_opaque: encode FAILED	
1125	2	xdr_opaque: bad op FAILED	
1126	2	xdr_bytes: size FAILED	
1127	2	xdr_bytes: bad size FAILED	
1128	2	xdr_bytes: out of memory	
1129	2	xdr_bytes: bad op FAILED	
1130	2	xdr_enum: dscmp FAILED	
1131	2	xdr_string: size FAILED	
1132	2	xdr_string: bad size FAILED	
1133	2	xdr_string: out of memory	
1134	2	xdr_string: bad op FAILED	
1135	2	xdr_array: size FAILED	
1136	2	xdr_array: bad size FAILED	
1137	2	xdr_array: out of memory	



Table E-2. Auspex Processor Error Messages (Continued)

Error	Priority code	Message text	
1138	2	xdr_mbuf: long crosses mbufs!	
1139	2	xdr_mbuf: putlong, long crosses mbufs!	
1140	2	xdrmbuf_getmbuf failed	
1141	2	bad auth_len gid %d str %d auth %d	
1142	2	ku_fastsend error: %s, dest. IP address=%s	
1143	2	svckudp_send: xdr_replymsg failed	
1144	3	%s: fs dirty fsck or mount read-only	
1145	1	mbuf_to_fc_sattr: x_handy < nfssattr fhandle_to_mbuf: x_handy < FHSIZE	
1180	3	ard%d: Automatic rebuild has started.	
1181	3	ard%d: Automatic rebuild has terminated due to error (%2x).	
1182	3	ard%d: ard number has been reassigned due to conflict.	
1187	3	Warning, Battery voltage low on SP NVRAM, write cache disabled. %d pages unwritten.	
1188	3	Warning, NVRAM checksum error, write cache disabled.	
1189	3	Warning, a cached write failed on slot %d	
1190	3	A mirrored partition has a dirty or damaged component, vp%d	
1191	2	Recovered, unknown sense key and code, slot %d, drive %d, sense key %2x, sense code %2x, VME addr %8x, sector %d.	
1192	2	VME Transfer Error, slot %d, Status %2x, VME address %8x.	
1193	2	Drive Fault, slot %d, drive %d, sense key %2x, sense code %2x.	
1194	2	Drive Offline, slot %d, drive %d.	
1195	2	SCSI Interface Err, slot %d, drive %d, SnsKey %2x, SnsCode %2x, Stat-0 %2x, Stat-1 %2x, Ctl %2x, Bus %2x.	
1196	2	SP Data Transfer Error, slot %d, drive %d, SnsKey %2x, SnsCode %2x, sector %9x.	
1197	2	Drive Box Power Failure, slot %d, Box %d.	
1198	2	SP Data Transfer Error, slot %d, drive %d. SnsKey %2x, SnsCode %2x, sector %9x.	
1199	2	Access to unknown SCSI device at slot=%d denied.	
1200	3	Recovered, drive error, no sense info, slot %d, drive %d.	
1201	3	Recovered, no drive index or sector signal, slot %d, drive %d.	
1202	3	Recovered, no drive seek complete, slot %d, drive %d.	
1203	3	Recovered, drive write fault, slot %d, drive %d.	
1204	3	Recovered, drive not ready, slot %d, drive %d.	
1205	3	Recovered, LUN communication error, slot %d, drive %d.	
1206	3	Recovered, drive ID CRC error, slot %d, drive %d, sector %d.	
1207	3	Recovered, drive DATA ECC error, slot %d, drive %d, sector %d.	



Table E-2. Auspex Processor Error Messages (Continued)

Error	Priority code	Message text
1208	3	Recovered, sector ID not found, slot %d, drive %d, sector %d.
1209	3	Recovered, drive seek failure, slot %d, drive %d.
1210	3	Recovered, retryable media error, slot %d, drive %d, sector %d.
1211	3	Recovered, correctable ECC error, slot %d, drive %d, sector %d.
1212	3	Recovered, drive sync transfer error, slot %d, drive %d.
1213	3	Recovered, drive power up or SCSI reset, slot %d, drive %d.
1214	3	Recovered, drive media corrupted, slot %d, drive %d, sector %d.
1215	3	Recovered, SCSI controller RAM error, slot %d, drive %d.
1216	3	Recovered, drive data path error, slot %d, drive %d.
1217	3	Recovered, drive power up diagnostic error, slot %d, drive %d.
1218	3	Recovered, drive message reject error, slot %d, drive %d.
1219	3	Recovered, SCSI drive internal error, slot %d, drive %d.
1220	3	Recovered, SCSI interface parity error, slot %d, drive %d.
1221	3	Recovered, SP detected SCSI parity error, slot %d, drive %d.
1222	3	Recovered, illegal SCSI message, slot %d, drive %d.
1223	3	Recovered, SCSI selection error, slot %d, drive %d.
1224	3	Recovered, SCSI disconnection error, slot %d, drive %d.
1225	3	Recovered, SCSI selection timed out, slot %d, drive %d.
1226	3	Recovered, SCSI handshake timed out, slot %d, drive %d.
1227	3	Recovered, SCSI bad status byte, slot %d, drive %d.
1228	3	Recovered, SCSI request sense failed, slot %d, drive %d.
1229	3	Recovered, SCSI reconnection error, slot %d, drive %d.
1230	3	Recovered, SCSI error in message in, slot %d, drive %d.
1231	3	Recovered, SCSI error in message out, slot %d, drive %d.
1232	3	Recovered, SCSI message reject error, slot %d, drive %d.
1233	3	Recovered, SCSI bad bus phase at data xfer, slot %d, drive %d.
1234	3	Recovered, SCSI bad bus phase at command xfer, slot %d, drive %d.
1235	3	Recovered, VME data transfer timed out, slot %d, drive %d.
1236	3	Recovered, AIC6250 fifo status error, slot %d, drive %d.
1237	3	Recovered, AFC chip status error, slot %d, drive %d.
1238	3	Recovered, AFC status error at disconnection, slot %d, drive %d.
1239	3	Recovered, SCSI inbound data parity error, slot %d, drive %d.
1240	3	Recovered, drive disconnected at odd byte, slot %d, drive %d.
1241	3	Recovered, AIC6250 status error, slot %d, drive %d.
1242	3	Recovered, AIC6250 error at disconnection, slot %d, drive %d.
1243	3	Recovered, DMA illegal operation, slot %d, drive %d.



Table E-2. Auspex Processor Error Messages (Continued)

Error	Priority code	Message text
1244	3	Recovered, DMA VME bus timed out, slot %d, drive %d.
1245	3	Recovered, DMA VME bus error, slot %d, drive %d.
1246	3	Recovered, DMA VME bus parity error, slot %d, drive %d.
1247	3	Too many recovered errors, replace the disk ASAP, slot %d, drive %d.
1248	3	Recovered, drive servo lost, slot %d, drive %d.
3001	1	readdirplus can't get inode %d
3002	1	readdirplus got unallocated inode %d
3003	1	getdirattrs can't get inode %d
3004	1	getdirattrs got unallocated inode %d
3005	1	isolating filesystem mounted on %s
3100	1	bread error in ufs_addblocks?



## Index

Symbols	/etc/remote 2-6, 2-18
	/etc/rpc 2-6
.rhosts file 2-22	/etc/sendmail.cf 2-6
/etc 2-4	/etc/services 2-6
/etc/aliases 2-5	/etc/shutdown 3-4
/etc/bootparams 2-4, 2-5, 2-28	/etc/syslog.conf 2-6
/etc/configuration 2-5, 2-17	/etc/ttytab 2-6, 2-17, 2-19
/etc/dumpdates	/etc/vpartab 2-6, 4-26, 4-31, 5-9
functions of 2-5	/export 2-21, 4-17, 6-33
specifying a date 8-2	/export/exec 4-11
specifying a partition name 8-3, 8-16	/export/exec/sun4.sunos.4.1 2-21
writing to using dump 8-2	/home 2-30, 6-33
/etc/ethers 2-4, 2-5, 2-28	/tftpboot 2-28
/etc/exports 2-5, 2-21, 2-28, 4-30	/usr 2-30, 4-10, 4-15, 6-32
/etc/format.dat 2-5, 4-7, 4-9	/usr/adm/messages 10-10
/etc/fstab 5-15	/usr/etc 2-9
/home mount point 2-30	/usr/kvm 2-30
adding mount entry 4-29	/usr/openwin 4-10, 4-16, 6-32
default 4-19	/usr/share 2-30
defining file type for root 4-13	/usr/share/lib/zoneinfo 2-17
disk partition entries 4-19	/var 4-10, 4-14, 6-32, 10-10
file system mount points 2-5	/var/adm/config.report 10-1, 10-2
modifying for write caching 7-8	/var/adm/messages
nosuid option 4-20	message logging 10-11
read-only (ro) option 4-5	messages related to file system
type of file systems mounted on the	isolation 6-15
HP 4-18	/var/log/auspex-messages 10-11
/etc/gateways 2-5	/var/spool/cron/crontabs/root 2-7
/etc/group 2-5	/var/yp 2-29
/etc/hosts 2-4, 2-5, 2-17, 2-21, 2-28	, vai, yp 2 20
/etc/hosts.equiv 2-6	Nonegaine
/etc/inetd.conf 2-6	Numerics
/etc/netgroup 2-6	1.35-GB disk 4-7
/etc/networks 2-6	
/etc/passwd 2-6	1.5.1 tapes
/etc/printcap 2-6	encoding 8-18
/etc/protocols 2-6	1.7 tape
/etc/rc.boot 2-6, 2-17	encoding 8-18 1.76-GB disk 4-7
/etc/rc.local 2-6	
/etc/rc.shutdown 3-9, 3-10, 3-11	1-GB disk 4-7
/ Ctc/ 1C.511utuo wii	2 step load 9-24

2-GB disk 4-7	client traffic 1-11
3-GB disk 4-7	consolidation 1-11
4.2 file system type 1-14, 4-13	expandability 1-11
4-GB disk 4-7	features 1-11
4-mm tape drives	auspex manual page 1-16
data format 8-5	Auspex Performance Monitor 9-1 to 9-48
naming conventions 4-3	Auspex processor error messages E-8
8-mm tape drives	Automatic file system mapping 4-21
naming conventions 4-3	ax_add_device command 1-13, 4-27,
9-GB disk 4-7	5-11, 6-4, 6-5
	ax_admmsg command 1-15
Α	ax_arp command 1-12, 10-14
^	ax_clonefs command 1-14, 4-27, 4-32,
acd device 1-13, 4-5	4-33, 8-2, 8-14, 8-15
Active file system 8-4	ax_config command 1-15, 10-2
Active status (virtual partition) 5-12	how to extract configuration
Adding a file system back on-line 6-14	information 10-2
Adding drives 4-25, 7-4	interactive and noninteractive
Adding entries to /etc/vpartab 5-14	modes 10-4
Address	ax_diskconf command 1-13, 4-26, 4-31,
broadcast 2-15	5-10, 5-23, 6-1, 8-13
Ethernet 2-30	ax_errd daemon 1-12
Internet 2-11, 2-14, 2-30	ax_expand command 1-14, 5-10, 5-15,
X terminal 2-11	5-18
AGE buffer statistics 9-13	ax_fddistat command 1-12
Aliases for host names 2-14	ax_fsutil command 1-14, 6-14
Alternate boot device 3-6	ax_hot_plug command 1-13, 4-27, 4-32,
Alternate superblock 6-24	4-33, 6-4, 6-5, 6-6
Analyzing performance monitor	ax_isolated command 6-12
data 9-47	ax_isolated daemon 6-12
Architecture	ax_keyenvoyd daemon 1-12
application 2-22	ax_kill command 1-14, 6-15, 6-16
FMP 1-2	ax_label command 1-13, 4-7, 4-11, 4-36,
hardware 1-5	4-41
kernel 2-22	ax_lfsd daemon 1-12
software 1-10	ax_load_flash command 1-15, 10-7
Sun system 2-22	ax_loadvpar command
Architecture-dependent executables	function of 1-13, 5-10
in /export/exec 4-11, 4-17	invoked by ax_mconvert 5-20
installation 2-21	ax_lslabel command 1-13, 4-42
location of 2-28	ax_mattach command 1-13, 5-10
arp command 10-15	ax_mconvert command 1-13, 5-11, 5-20,
ARP table	5-21
displaying or flushing 10-14	ax_mdetach command 1-13, 5-11
ask me option in boot command 3-6	ax_mrestore command 4-27, 4-32, 4-33,
astn 1-13	5-11, 5-13
ATM 1-8	ax_netstat command 1-12, 10-14, 10-15
Attaching a member to a mirrored	ax_nfsd daemon 1-12
partition 5-10	ax_perfhist
Auspex environment	options and arguments 9-46
availability 1-11	starting 9-46
	ax_perfhist command 1-12, 9-1,



9-18 to 9-46	single-user option 3-6
ax_perfmon command 1-12, 9-1,	Break key 10-12
9-2 to 9-17	Broadcast address 2-15
ax_remove_device command 1-13, 4-33,	Buffered I/O daemons (BIODs) 7-3
5-11	
ax_sputil command 6-16, 6-27	С
ax_startup daemon 1-12	C
ax_statd daemon 1-12	a (continue) command 10.19
ax_statd2 daemon 1-12	c (continue) command 10-12 Cache
ax_statuz daemon 1-12 ax_tapestats command 4-27, 4-32	
*	age distribution of contents 9-13
ax_timed daemon 1-12	disk 1-6
ax_vold daemon 1-13, 5-11	information about four cache
ax_vpstat command 1-13, 5-11	types 9-12
ax_write_cache command 1-15, 7-6, 7-8	write 7-2
ax_write_cache command options 7-8	Capacity of tape drives 8-5
	Capturing a screen in ax_perfhist 9-39
В	CD-ROM
	adding a drive 4-25
b (boot) command 3-2, 4-2	booting from 3-6, 4-5, 6-27
Backing up	device name 1-13, 4-2
a clone partition to tape 8-15	distribution medium 2-25
a virtual partition 4-11	ejecting 4-6
by cloning 8-12	how to use 4-5
full 8-2	in various commands 4-2
incremental 6-26, 8-2	loading 4-5
more than 20 drives 8-10	location 4-26
on-line 8-12	mount point 2-25
Backspace 2-10	mounting and unmounting file
Backup copy of root disk 6-26	systems 4-5
Backup superblock 6-24	numbering 4-2
Bad spots on a tape 8-20	CD-ROM drives
BADCHECKSUM 7-6	acdn 1-13
Battery for NVRAM on the Write	Changing
Accelerator 7-3, 7-6	from single-user mode to multi-user
BATTERY_LOW (write cache status) 7-6	mode 6-4
Baud rate 2-19	the drive configuration 7-11
Blank tape error 8-19	the partitioning on a disk 4-39
Blocking factor of tape 8-7	the root password 2-3
Boot	Checking an isolated file system 6-14
alternate device 3-6	Checking the write cache state 7-9
arguments 3-4	Checksum for validating write cache
ask me option 3-6	control data 7-6
command 1-13	checksumming and NetServers 2-5
default boot device 3-5	Cleaning tape drive heads 8-20
from backup root disk 6-26	Clearing a histogram 9-22
from CD-ROM 3-6, 4-5, 6-27	Client
initializing the Write Accelerator 7-6	diskless 2-20
manual 3-4	
	effects of file system isolation on 6-13
messages 4-28, 6-3, E-2	information worksheet D-1, D-4
multi-user mode 3-2	server 2-15
procedure 3-2	when to reboot 2-30

workstation 2-28	eject 1-13, 4-2, 4-6
client	exportfs 4-30
UDP checksumming 2-5	fastboot 3-2, 3-11
Client files, distributing among multiple	fasthalt 3-2, 3-11
disks 4-11	fddistat 1-12
Cloning a virtual partition	format 1-13, 4-7, 4-36, 5-7, 6-26
important guidelines 8-13	fsck 1-14, 3-8, 4-22, 6-26, 10-10
meaning of 8-12	getcores.sh 1-15
procedure for 8-14	go (goto) 10-11
Commands	halt 1-15, 3-2, 3-10, 10-12
arp 10-15	init 1-12, 2-19
ax_add_device 1-13, 4-27, 5-11, 6-4,	installboot 6-26, 6-28
6-5	MAKEDEV.auspex 1-13, 4-4
ax_admmsg 1-15	mkdir 4-29
ax_arp 1-12, 10-14	mount 4-2, 4-22, 4-30
ax_clonefs 1-14, 4-27, 4-32, 4-33, 8-2,	mt 8-7, 8-22
8-14, 8-15	ncheck 8-17
ax_config 1-15, 10-2	newfs 4-10, 4-29, 5-7, 6-26, 6-28
ax_diskconf 1-13, 4-26, 4-31, 5-10,	nfsstat 1-12, 10-14, 10-19
5-23, 6-1, 8-13	NSconfig 2-10, 2-12, 2-17
ax_expand 1-14, 5-10, 5-15, 5-18	NSinstall 6-26
ax fsutil 1-14, 6-14	passwd 2-3
ax_hot_plug 1-13, 4-27, 4-32, 4-33,	rc 1-15
6-4, 6-5, 6-6	reboot 1-15, 3-2, 3-8, 3-10
ax_isolated 6-12	reporter.sh 2-7
ax_kill 1-14, 6-15, 6-16	restore 8-2, 8-3, 8-9
ax_label 1-13, 4-7, 4-11, 4-36, 4-41	SetupClient 2-27
ax_load_flash 1-15, 10-7	SetupExec 2-22
ax_loadvpar 1-13, 5-10, 5-20	shutdown 3-8, 3-9, 10-12
ax_lslabel 1-13, 4-42	software configuration 2-9
ax_mattach 1-13, 5-10	sync 3-8, 3-9, 10-10
ax_mconvert 1-13, 5-11, 5-20, 5-21	tunefs 4-22
ax_mdetach 1-13, 5-11	umount 4-22
ax_mrestore 4-27, 4-32, 4-33, 5-11,	xwd 9-38, 9-39
5-13	ypinit 2-4
ax_netstat 1-12, 10-14, 10-15	Concatenated virtual partition
ax_perfhist 1-12, 9-1, 9-18 to 9-46	changing to a mirrored partition 5-19
ax_perfmon 1-12, 9-1, 9-2 to 9-17	definition 5-4
ax_remove_device 1-13, 4-33, 5-11	moving 5-21
ax_sputil 6-16, 6-27	Configuration
ax_tapestats 4-27, 4-32	client-specific 2-1
ax_vpstat 1-13, 5-11	commands 2-9
ax_write_cache 1-15, 7-6, 7-8	displaying data about 10-5
b (boot) 1-13, 3-2	form 2-9
c (continue) 10-12	HELP function 2-10
crash 10-10	management 10-2
cron 8-10	server-specific 2-1
dd 4-38, 6-26	software 2-1
df 4-22, 5-3	worksheets D-1
dkinfo 1-13, 4-2, 4-9, 4-28, 5-24, 6-27	Configuration Window 9-30
dump 1-15, 8-2	Console
dumpfs 4-2, 4-22	port 2-18



turning on and off 10-12	Damaged status (virtual partition) 5-12
Conventions	Data cache 9-12
naming disks 4-10	Data cache buffers for the FP 9-8
naming partition tables 4-10	Data compression 4-3, 8-6
representation of variables E-8	Data Display Window 9-20, 9-22
Converting a striped or concatenated	configuring layout 9-28
partition to a mirrored partition 5-11,	options 9-35
5-19	two histograms 9-23
Copying	DataGuard 1-3
ax_config information to a file 10-6	Date and time formats 2-17
backup root disk 6-27	Date and time on the server 2-16
data from one disk to another 6-22,	dd command 4-38, 6-26
6-23	Default boot device 3-5
the root partition using dd 6-26	Default disk partitions 4-10
Core dumps from an NP 10-10	Defining a virtual partition 5-14
CPU utilization, monitoring 9-2	Delete key 2-10
crash command 10-10	· · · · · · · · · · · · · · · · · · ·
	Detaching a member from a mirrored
Crashes, system 5-13, 10-10	partition 5-11 Device
Creating a virtual partition 5-14 cron command 8-10	boot 3-6
Ctrl-C 2-10	creating new 4-4
Ctrl-D 3-6, 6-4, 6-6, 6-7	defining with
Ctrl-F 2-10	MAKEDEV.auspex 1-13
Ctrl-H 2-10	names 4-2, 4-5
Ctrl-L 2-10	special files 4-4
Ctrl-R 2-10	Device driver
Ctrl-U 2-10	UNIX 6-25
Ctrl-W 2-10	vp 1-13, 5-9
Customer support	Device names
International customers xx	acdn 1-13
North America xx	adn 1-13
	df command 4-22, 5-3
D	Diagnostic switch on the Host
	Processor 10-12, 10-13
Daemons	Diagnostics
ax_errd 1-12	extended 3-3
ax_isolated 6-12	power-on 3-3
ax_keyenvoyd 1-12	Dial-in modem 2-19
ax_lfsd 1-12	Dial-out modem 2-19
ax_nfsd 1-12	Differences between mirroring and cloning
ax_startup 1-12	a virtual partition 8-12
ax_statd 1-12	Dirty status
ax_statd2 1-12	virtual partition 5-12
ax_timed 1-12	write cache status 7-6
ax_vold 1-13, 5-11	Disabling the Write Accelerator 7-9
nfsd 1-12	Disk
routed 2-11, 2-16	1.35-GB 4-7
stopnfsd 1-12	1.76-GB 4-7
syslogd 10-11	1-GB 4-7
Damaged root disk 6-26	2-GB 4-7
Damaged root disk 0 20  Damaged root partition 6-26	3-GB 4-7
Zamagea root partition 0 20	4-GB 4-7

9-GB 4-7	multiple histograms 9-23
adding 4-25	network statistics with
blocks unreadable 6-25	ax_netstat 10-15 to 10-18
cache 1-6	NFS statistics with nfsstat 10-19
configuration tool 5-23	system configuration data 10-5
copying data 6-22, 6-23	Distributing client files 4-11
copying the label 6-23	Distributing file systems 9-47
damaged root 6-26	Distribution medium
defect 4-38	CD-ROM 2-25, 4-5
device name 4-2	
	default mount point 2-25
displaying configuration	tape 2-25
information 5-23	dkinfo command 5-24
drive information worksheet D-1,	checking disk label 4-28
D-6	checking root disk label 6-27
drive serial number 7-4	different from SunOS 1-13
dynamic table format 4-22	displaying partition table 4-9
formats 4-22	used with CD-ROM 4-2
formatting 4-38	Documentation
label 4-7, 4-28, 5-7, 5-8, 6-24	applicable to the NetServer xvii
numbering 4-2	installing online B-1
partition schemes 4-7	Domain name 2-15, 2-30
partitioning 4-12	Down arrow key 2-10
raw device 4-3	Drive
reducing latency 5-2	adding 4-25, 7-4
remapping a sector 6-23	hot-plugging 4-26
root backup 8-11, 8-14	LED indicator 8-6
<u>-</u>	
sample ax_diskconf display 5-24, 5-25	removing 4-31
	SCSI ID 4-34
sector 0 6-24	types supported by the
static table format 4-22	NetServer 4-25
surface analysis 4-38	Drive configuration change, recovering
Disk errors	from 7-11
definition 6-2	Drive rack, installing 4-25
during normal operation 6-8	Drive slot number in an SP error
intermittent 6-21	message 6-8
on a mirrored partition 5-13	Driver, virtual partition 5-9
recovering on a running system 6-19	dump command
Disk partition tables	different from SunOS 1-15
default 4-8	overview 8-2
striped 4-8	syntax 8-2
Disk partitioning 4-7	with the T and m options 8-15
Disk problems 6-11	Dump level 8-2
Disk type	dumpfs command 4-2, 4-22
choosing using format 4-37	Dynamic Table Format 4-22
Diskless SunOS client	Dynamic Table Format 4 22
	_
adding 2-28 loading executables 2-20	E
<u> </u>	
removing 2-28	ECC (Error Checking and Correcting)
Displaying	error 8-20
a disk partition table 4-9	Echo mode 10-13
ARP table 10-14	eject command 1-13, 4-2, 4-6
disk configuration information 5-23	



Email to Auspex 2-7, 2-16, 3-10	FFFS (Fat Fast File System) 4-22
Enabling the Write Accelerator 7-8	FFS (Fast File System) 4-22
End-of-tape error 8-20	Field
Enlarging a striped or concatenated virtual	input for configuration forms 2-10
partition 5-10, 5-15, 5-16	NSconfig form 2-14
Error	NSinstall form 6-31
Auspex processor E-8	SetupClient form 2-30
detected by LFS 6-10	SetupExec form 2-25
file system problems 6-9	SetupTty form 2-19
logging 10-11	toggle 2-9
media 6-2, 6-23	File 6-10
message from the FP 6-9, 6-25	damaged inode 8-17
message from the NP 10-11	device 4-4
message from the SP 6-3, 6-8, 6-23, 6-24	restoring from 1.5.1-produced dump tapes 8-18
message from vmunix 6-9	virtual partition 5-9
notification E-8	File Load Window 9-21, 9-22
recovered E-8	input file list menu options 9-37
recovery on mirrored partitions 5-13	File Processor
Error conditions for file system	contraints 4-22
isolation 6-10	error messages 6-9, 6-25
Errors detected by UFS 6-10	mounting file systems on 4-18
Ethernet	File system
100Base-T 1-8, 2-14, 2-32	/export 2-21, 4-17
address 2-30	/usr 4-15
analyzing data 9-47	/usr/openwin 4-16
interfaces 2-14	/var 4-14
ports 1-4	active 8-4
Examples	adding back on-line 6-14
ax_diskconf display with -p	backing up on-line 5-2
option 5-25	disk formats 4-22
disk partitioning among multiple	distributing across disk drives 5-2,
disks 4-12	5-4, 9-47
Executables, architecture-	inconsistency 6-3, 6-13
dependent 2-21, 2-28, 4-11	isolation 2-31, 6-10
Expanding a striped or concatenated	isolation after reparable media
virtual partition 5-10, 5-15, 5-16	errors 6-17
exportfs command 4-30	isolation caused by corruption 6-18
exports file 1-14	isolation caused by irreparable media
Extended diagnostics 3-3	errors 6-15
· ·	killing sleeping processes on 6-15
F	larger than a physical disk 5-2
•	listing using ax_diskconf 5-23
Fast File System (FFS) file system	mapping to FPs 4-20
format 4-22	on an inaccessible drive 6-19
fastboot command 3-2, 3-11	protecting from disk or media
fasthalt command 3-2, 3-11	failure 5-2
Fat Fast File System (FFFS) file system	quiescent 8-4
format 4-22	releasing 6-14
FDDI	root 4-13
interfaces 2-14	specifying the type 4-19
	Tahoe 4-22

type LFS 1-14, 4-18, 4-30, 5-11	Hardware upgrade tracking 10-5
Filter file	Hayes-compatible modem 2-18
creating 9-16	HELP function in configuration
used in ax_perfmon 9-16 to 9-17	forms 2-10
First time NetServer configuration 2-3	High Sierra File System (HSFS) 4-5, 4-18
Flushing an ARP table 10-14	Histogram
	2 step load 9-24
Flushing unwritten data from the Write	
Accelerator 7-10	clearing 9-22
FMK (Functional Multi-processing	continuous plot 9-28
Kernel) 1-10	defining its appearance 9-24
FMP (Functional Multi-processing)	displaying average statistics 9-29
advantages of 1-2	printing 9-38
hardware implementation 1-9	Properties Window 9-25 to 9-27
meaning of 1-5	removing 9-24
Font files for X terminals 4-16	selecting samples 9-32
Form	server data 9-18
configuration 2-9	summing parameters 9-35
NSconfig 2-13	Home 2-10
NSinstall 6-31	Home directories for clients 2-30
SetupClient 2-29	Host information worksheet D-2
SetupExec (remote devices) 2-24	Host name
SetupTty 2-19	alias 2-14
format command	
	server 2-11, 2-14
difference from SunOS 1-13	Host Processor
format option 4-38	functions of 1-5
invoking 4-36	mounting file systems on 4-18
label option 6-24	PROM monitor 3-3
repairing a drive 6-26	SCSI port 4-35
repartitioning a disk 4-7	Host Processor VII (HP VII)
verifying if disk label exists 6-5	architecture 1-3
rintual portition F 7	
virtuai partition 5-7	Hot-plugging a drive 4-26
virtual partition 5-7 Formats, date and time 2-17	Hot-plugging a drive 4-26 HSFS (High Sierra File System) 4-5
Formats, date and time 2-17	Hot-plugging a drive 4-26 HSFS (High Sierra File System) 4-5
Formats, date and time 2-17 Formatting a disk 4-38	HSFS (High Sierra File System) 4-5
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39	
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12	HSFS (High Sierra File System) 4-5
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2	HSFS (High Sierra File System) 4-5  I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15 go (goto) command 10-11	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26 freeing 6-13
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15 go (goto) command 10-11 Greater than 2-GB virtual partitions 5-3	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26 freeing 6-13 number per cylinder group 4-22
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15 go (goto) command 10-11	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26 freeing 6-13 number per cylinder group 4-22 Input fields in configuration forms 2-10
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15 go (goto) command 10-11 Greater than 2-GB virtual partitions 5-3  H	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26 freeing 6-13 number per cylinder group 4-22 Input fields in configuration forms 2-10 Input file list 9-37
Formats, date and time 2-17 Formatting a disk 4-38 Formatting time 4-39 FP Statistics screen 9-10, 9-12 Free disk space on the server 5-3 fsck command 1-14, 3-8, 4-22, 6-26, 10-10 FTP on the Network Processor 1-8 Full backup 8-2  G getcores.sh command 1-15 go (goto) command 10-11 Greater than 2-GB virtual partitions 5-3	I/O cache memory 9-47 I/O monitoring 9-2 I/Os, reduction of 7-3 Inconsistent file systems 6-3, 6-13 Incremental backup 6-26, 8-2 init command 1-12, 2-19 Inode cache 9-12 Inodes for restored files 6-26 freeing 6-13 number per cylinder group 4-22 Input fields in configuration forms 2-10



drive 6-31	file system 6-15
Interfaces	
Ethernet 2-14	1
FDDI 2-14	_
Intermittent disk errors 6-21	Label
International customer support xx	disk 4-7, 4-28, 4-40, 5-7, 5-8, 6-23, 6-24
Internet	option in format command 6-24
address 2-11, 2-14, 2-30	LED on the tape drive 8-6, 8-22
Invisible disk on a running system 6-19	Left arrow key 2-10
Isolating file systems	Level 0 backup 6-26, 6-28
effects on network clients 6-13	LFS
error messages in	changes to df, mount, and umount
/var/adm/messages 6-15	commands 4-22
killing sleeping processes on 6-15	default format 4-22
sample procedures for recovering	file system isolation 4-18
from 6-15 to 6-19	file system type 1-14, 4-18, 4-30, 5-11
stale file handles seen by clients 6-16	
state the handles seen by chefits 0 10	mapping file systems to FPs 4-20
17	Listing backup superblocks 6-24
K	Listing file systems on a drive 5-23 Load Hist button 9-24
Kernel	
alternate file 3-6	Loading statistics to a histogram 9-21
	Local drive location 2-25
executables 2-30	Local terminal 2-19
panic messages 10-11	LRU buffer statistics 9-13
rebuilding 10-9	
reconfiguring 2-5	M
setting parameters 2-6	_
Keys	Mail Recipients 2-7
? (question mark) 2-10	mailmessages.sh script 2-7, 2-16
Backspace 2-10	Maintaining tape drives 8-1
Break 10-12	MAKEDEV.auspex command 1-13, 4-4
Ctrl-C 2-10	Making a new NIS map 4-14
Ctrl-F 2-10	maning a new rue map
	Man pages
Ctrl-H 2-10	
	Man pages
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10	Man pages OpenWindows 4-16
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU)
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10 Return 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15 specifying in NSconfig form 2-15 Maximum virtual partition size 5-3
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10 Return 2-10 Right Arrow 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15 specifying in NSconfig form 2-15
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10 Return 2-10 Right Arrow 2-10 Space Bar 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15 specifying in NSconfig form 2-15 Maximum virtual partition size 5-3 Maxusers kernel parameter 2-6 Media errors
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10 Return 2-10 Right Arrow 2-10 Space Bar 2-10 Space bar 2-10 Tab 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15 specifying in NSconfig form 2-15 Maximum virtual partition size 5-3 Maxusers kernel parameter 2-6 Media errors definition 6-2
Ctrl-H 2-10 Ctrl-L 2-10 Ctrl-R 2-10 Ctrl-U 2-10 Ctrl-W 2-10 Delete 2-10 Down Arrow 2-10 Home 2-10 Left Arrow 2-10 Page-Down 2-10 Page-Up 2-10 Return 2-10 Right Arrow 2-10 Space Bar 2-10 Space bar 2-10	Man pages OpenWindows 4-16 Managing virtual partitions 5-14 Mapping LFS file systems to FPs 4-20 Mapping physical and virtual partitions 6-1 Marker in a histogram 9-32 Master server 2-15 Maximum number of virtual partitions on an SP 5-7 Maximum Transmit Unit (MTU) off the FDDI network 2-15 specifying in NSconfig form 2-15 Maximum virtual partition size 5-3 Maxusers kernel parameter 2-6 Media errors

Menu	FP 4-18
format command 4-38	HP 4-18
partition 4-39	Moving a one-membered mirrored
Message	partition 5-22
Auspex processor errors E-1, E-8	Moving a striped or concatenated
error E-1	Partition 5-21
kernel panics 10-11	Moving a two-membered mirrored
logging 10-11	partition 5-22
panic E-8	mt command 8-7, 8-22
recovered errors E-8	Multiple histograms, displaying 9-23
standard boot E-1	Multi-user mode
startup errors E-1	backups and restores in 8-4
text E-8	booting 3-2
Meta cache 9-12	entering monitor mode 3-4
Mirrored partition	for running SetupClient 2-29
adding a new member 5-21	for running SetupExec 2-22
attaching a member 5-9	replacing a drive in 6-6
automatic error recovery 5-13	replacing a drive in 00
changing from a concatenated	<b>N</b> 1
partition 5-19	N
definition 5-5	Name
detaching a member 4-26	
expanding a one-membered 5-18	conventions for disks and partition
moving 5-19, 5-22	tables 4-10 devices 4-2
need for backup 8-1	
	host 2-11
removing a member 5-21	NIS domain 2-15, 2-30
replacing a drive containing 6-20 unrecovered disk errors 6-10	X terminal 2-11
	Name cache 9-12
used with write cache 5-5	Naming a statistics file 9-19
with one member 6-22	Naming Exabyte 8200 tape drives 4-2, 8-6
Mirrored reads 5-5	ncheck coomand 8-17
Mirrored writes 5-5	Netinterfaces kernel parameter 2-7
Missing disk label 6-24	NetServer documentation xvii
mkdir command 4-29	NetServer serial number 2-17
Mode	NetServer throughput, improving with the
echo 10-13	Write Accelerator 7-3
multi-user 2-22, 2-29, 3-2, 3-4, 8-4	Network
single-user 3-4, 3-6, 8-4	monitoring the utilization of 9-2
Modem	server software structure 1-10
dial-in 2-19	Network interfaces
dial-out 2-19	managing 10-14
for remote diagnostics 10-12	setting memory use 2-7
Hayes-compatible 2-18	Network Processor 1-6
port 2-18	functions of 1-6
Monitoring	optional products 1-8
CPU utilization 9-2	newfs command
I/Os 9-2	for a physical partition 5-7
network utilization 9-2	for root partition 6-26, 6-28
processor utilization 9-2	initializing file systems 4-10, 4-29
mount command 4-2, 4-22, 4-30	NFS
Mounting a CD-ROM 4-5	NetServer compatibility 1-4
Mounting file systems	- •



nisstat command 1-12	Online documentation, printing xxi
reads and writes on a clone	Online help xx
partition 8-15	OpenWindows 3.0 4-16
workers 9-9	Operating system
write acceleration 7-3	installing from the CD-ROM 6-29
nfsstat command 1-12, 10-14, 10-19	SunOS 3-2
NIS	Operator errors in handling tape
client 2-15	drives 8-19
database 2-28	Optional product
domain name 2-11, 2-15, 2-30	documentation B-1
hosts database 2-28	Optional products
master server 2-4, 2-15, 2-29	100Base-T Ethernet 1-8
service 2-15, 2-30	ATM 1-8
service type 2-11	FTP on the Network Processor 1-8
slave server 2-4, 2-15	Options for tape drives
None (write cache status) 7-6	no-rewind 4-2, 8-3
nonvolatile RAM (NVRAM) 1-6	rewind 4-2
No-rewind option 4-2, 8-3	Overall Properties Window 9-28
North America customer support xx	
nosuid option in /etc/fstab 4-20	Р
Notifications E-8	•
NP statistics 9-9	Page-Down 2-10
NP Statistics screen 9-8	Page-Up 2-10
NSconfig	Panic messages 10-11
after running NSinstall 6-34	Panics 10-10, E-8
command 2-10, 2-12, 2-17	Parameter button 9-23
files modified by 2-17	Parameters menu in ax_perfhist 9-41
form 2-13	Partition Partition
form fields 2-14	
how to run 2-11	changing 4-39
NSinstall 2-2, 6-29	default disk partitions 4-7
for a spare root drive 6-31	managing 5-14
form 6-31	raw disk 4-3
form fields 6-31	scheme 4-7, 4-28
NSinstall form 6-30	size 4-7
	virtual 5-2
NSupdate 2-2, 6-29	Partition menu
NVRAM (nonvolatile RAM) 1-4, 1-6, 7-3	label option 4-40
	select option 4-39
0	Partition option in the format menu 4-39
	Partition tables
Off (write cache status) 7-6	root drive 4-8
On (write cache status) 7-6	Partitioning information in sector 0 6-24
ONC/NFS 1-4	passwd command 2-3
One-membered mirrored partition 6-22	Performance data
On-line backup 8-12	capturing and displaying in real
On-line documentation	time 9-2
printing B-7	Performance Monitor
• •	
8 · · · · · · · · · · · · · · · · · · ·	analyzing data 9-47
Online documentation	commands entered interactively 9-5
FrameViewer license B-2	FP statistics 9-3
overview B-2	how to run 9-3

NP statistics 9-3	disk label to another location 6-24
purpose of 1-12	disk sector to a new location 6-25
SP statistics 9-3	superblock to another location 6-24
virtual partition statistics 9-3	Reboot
write cache statistics 9-3	after a system crash 5-13
pkgadd(1M) 1-7	server fails to check file systems 6-24
Port	with a disk error 6-3
baud rate 2-19	reboot command 1-15, 3-2, 3-8, 3-10
console 2-18	Rebuilding the system kernel 10-9
modem 2-18	Reconfiguring a virtual partition 5-14
serial 2-18, 10-12	Reconfiguring the kernel after
Power-on	NSinstall 6-34
diagnostics 3-3	Recoverable disk errors 6-8
messages 6-3	Recovered errors E-8
Primary host name	Recovering
for a network interface 2-11, 2-14	from a damaged root disk 6-26
Primary Memory 9-47	from a drive configuration
Print Command	change 7-11
output 9-40	from disk hardware errors 7-10
Print command 9-38	from errors on a mirrored
Print Options Window 9-38 to 9-40	partition 5-13
Printer, serial 2-19	from root disk failures 6-26
Printing	from write cache errors 7-9
online documentation xxi	Reduction of disk I/Os 7-3
Printing histograms 9-38	Reinstalling the operating system from
Processor	CD-ROM 6-26
errors E-8	Relabeling a disk 6-24
FMP 1-5	Releasing an isolated file system 6-14
Host 3-3	Remapping a bad sector 6-23
Network 1-6, 9-47	Remote device location 2-25
Storage 9-47	Remote diagnostics 10-12
utilization 9-2	Remove Hist button 9-22
PROM download utilities 10-7	Removing
PROM monitor	a CD-ROM 4-6
booting from 3-4	a drive 4-31, 7-4
functions of 3-3	a failing drive 6-7
prompt 3-4, 10-11, 10-12	a histogram 9-24
Prompt, PROM monitor 3-4, 10-11, 10-12	a member from a mirrored
Properties of a histogram 9-25	partition 5-11
	a statistics file 9-37
Q	Repairing
	the root drive using fsck 6-27
Qty Hists button 9-23	Replacing a drive
Question mark key 2-10	containing multiple partitions 6-22
Quiescent file system 8-4	in a one-member mirrored
	partition 6-22
R	in multi-user mode 6-6
	in single-user mode 6-5
Raw disk device 4-3	procedure for 4-25
rc command 1-15	with mirrored partitions 6-20
Reassigning	with unmirrored partitions 6-19
	Replacing an inaccessible drive at



reboot 6-4	Sector 0 6-24
Replacing an invisible disk on a running	Serial
system 6-19	port baud rate 2-19
reporter.sh command 2-7	ports 2-18, 10-12
reporter.sh script 2-7	printer 2-19
restore command	Serial number
overview 8-2	of drives 7-4
syntax 8-3	of NetServer 2-17
Restoring a file	Server architecture 1-5
from 1.5.1-produced dump	Server host name 2-11, 2-14
tapes 8-18	Server problems 10-9
<u>-</u>	*
with damaged inode 8-17	Setting SCSI ID on a drive 4-34
Restoring status (virtual partition) 5-12	Setting the date and time on the
Restrictions	server 2-16
on concatenated and striped	SetupClient 2-27
partitions 5-7	command 2-27
on mirrored partitions 5-8	flowchart 2-27
on virtual partitions 5-7	form 2-29
Return key 2-10	form fields 2-30
Rewind option for tape drives 4-2	functions of 2-28
Right arrow key 2-10	how to run 2-29
Root	loading root and swap
damaged partition 6-26	directories 4-11
directory 2-28, 4-7, 4-11	SetupExec 2-21
file system 4-13	flowchart 2-20
installing using NSinstall 6-32	form (remote devices) 2-24
Root disk	form fields 2-25
backup 6-26, 8-11, 8-14	how to run 2-22
booting from backup 6-26	loading files to /export/exec 4-11
copying 6-27	SetupTty
damaged 6-26	form 2-19
drive label 6-27	form fields 2-19
recovering from failures 6-26	shutdown command 3-8, 3-9, 10-12
replacing 4-26	Shutting down the NetServer 3-9
Route daemon 2-11, 2-16	Silent mode of ax_perfmon 9-19
Runt partitions, combining 5-2	Single-user mode 3-4, 3-6, 6-3, 6-5, 8-4
	Site-specific configuration files 6-26, 6-28
S	Size of a clone partition 8-13
•	Size of a mirrored partition 5-8
savecore command 10-11	Slave server 2-15
Saving ax_config information to a	Slot number in an SP error message 6-8
file 10-6	Software
Saving performance statistics to a	architecture 1-10
file 9-19	configuration 2-1, 2-9
Screen captures in ax_perfhist 9-39	Software architecure of the
SCSI	NetServer 1-10
adding drives to host port 4-35	SP problems that cause file system
channel management 1-6	isolation 6-10
ID of drive 4-34	SP retries 6-8
interface error message 6-4	SP Statistics screen 9-13
port on Host Processor 4-4, 4-35	Space Bar 2-10
Port off 11050 110005501 1 1, 1 00	Spare root drive 6-31
	T

Stale file handle, during file system	on root disk 3-7
isolation 6-16	sync after reboot 3-2
Standard boot messages E-2	sync command 3-8, 3-9, 10-10
Starting ax_perfmon 9-3	Synced status (virtual partition) 5-12
Static RAM (NVRAM) 1-6	syslog.conf 10-11
Static Table Format 4-22	syslogd 10-11
Statistics displayed by ax_perfhist 9-40	System
FP 9-43	crashes 5-13, 10-10
HP 9-41	panics 10-10, E-8
NP 9-41	System cache I/O memory 3-11
SP 9-45	System cache is a memory of its
Statistics parameters in	displaying 10-5
ax_perfhist 9-40 to 9-45	System kernel, rebuilding 10-9
<u>-</u>	
Statistics screens displayed by	System Summary statistics 9-6, 9-7
ax_perfmon 9-6 to 9-15	System Summary statistics 9-3
Status, virtual partition 5-11	_
stopnfsd daemon 1-12	T
Storage Processor	
error messages 6-3, 6-8, 6-23, 6-24	Tab key 2-10
features of 1-6	Tahoe File System 4-22
how it maintains drive	Tape
information 7-4	adding a drive 4-25
number needed 9-47	blocking factor 8-7
number of virtual partitions 5-7	capacity 8-5 to 8-7
reading from a mirrored partition 5-5	cleaning drive heads 8-20
support for virtual partition	density 4-2
clones 8-12	distribution medium 2-25
warning message 6-10	drive capacities 8-5
Write Accelerator option 1-6	drive device names 4-2
writing to a mirrored partition 5-5	drive errors 8-19, 8-22
Storage subsystems 1-4	dump and restore arguments 8-7
Stripe size of a striped partition 5-8, 5-10	I/O error 8-19
Striped partition 5-4, 5-9	jam error 8-21
Striped virtual partition	LED indicator 8-22
moving 5-21	media error 8-20
Subnet mask 2-15	memory error 8-20
Summing parameters in histograms 9-35	no-rewind option 4-2, 8-3
Sun system architecture 2-22	on-line error 8-22
SUNBIN, CD or tape 2-22	read error 8-21
SunOS	rewind option 4-2
enhancements 1-12	write error 8-6, 8-19, 8-21, 8-22
error numbers E-1	TCP/IP 1-4
pointers to documentation 1-16	Temperature requirement for battery 7-5
system calls E-1	Terminal
version 2-30	console X 2-11
SunOS kernel, reconfiguring 2-5	local 2-19
Superblock 6-24	type 2-11, 6-30
Super-user privileges 3-2	Terminology xvii
Surface analysis 4-38	Throughput, NetServer 7-3
Swap file on client machine 2-30	Time required for formatting a drive 4-39
Swap space	Time required for formatting a drive 4-39 Time zone
on client machine 2-28	I IIIIC ZUIIC
CAL CHICAGO HARDEN W WU	



list C-1	larger than 2-GB 5-3
setting 2-10, 2-16	managing 5-14
Toggle field 2-9	maximum size of 5-3
Tracking hardware upgrades 10-5	mirrored 5-5, 5-9, 6-10, 8-1
ttya 2-18	numbering 5-9
ttyb 2-18, 10-12	reconfiguring 5-14
tunefs command 4-22	
	recovering disk errors 6-20
Types of statistics displayed by	restoring status 5-12
ax_perfhist 9-40	restrictions 5-7
Types of tapes for backup 8-3	running newfs on 5-14
	status 5-11
U	striped 5-4, 5-9
	synced status 5-12
UDP checksumming 2-5	types of 5-4
UFS file system format 4-5, 4-22, 7-8	Virtual Partition Manager 1-4
umount command 4-22	components of 5-1
Unattended backup, scheme for 8-10	copying data between disks 6-22
Unintialized (write cache status) 7-6	Virtual Partition Statistics screen 9-15
	VME 1-4, 4-18
UNIX	VME 1-4, 4-16 VME bus 1-6
4.2 BSD file system format 4-22	
device driver 6-25	vmunix 3-6, 4-4, 6-9, 8-20
Unmounting a CD-ROM 4-5	Volume daemon 5-11
Unreadable disk blocks 6-25	vp device driver 1-13, 5-9
Unreadable disk label 6-24	vpartab table 1-13, 5-9, 5-10, 5-14
Unreadable superblock 6-24	
Unrecovered disk errors 6-9, 6-10	W
Unrecovered file system problems 6-9	••
Unwritten data in the write cache 7-6	WarmStart 1-3
Up arrow key 2-10	Warning message from the SP 6-10
Cp arrow key 2 10	Worksheet
V	client information D-1, D-4
	configuration D-1
Version, SunOS 2-30	disk drive information D-1, D-6
Virtual partition	host information D-2
active status 5-12	virtual partition information D-1, D-7
advantages of 5-2	Write acceleration
cloning 8-12	accelerator board 7-2
concatenated 5-4	analyzing the need for 9-48
converting to a mirrored	badchecksum status 7-6
partition 5-11	dirty status 7-6
creating 5-14	disabling 7-9
damaged status 5-12	enabling 4-22, 7-8
defining 5-14	error recovery 7-9
· ·	examining statistics with
dirty status 5-12	ax_perfmon 7-8
disk errors at reboot 6-3	
error recovery 5-13	how to manage the cache 7-9
expanding 5-15	none status 7-6
files, drivers, and commands 5-9	NVRAM 7-3
for /export/root and	off status 7-6
/export/swap 4-11, 4-17	on status 7-6
information worksheet D-1, D-7	overview 7-2
mornation wormsheet B 1, B .	



X console terminal 4-16 restrictions 7-5 states and commands 7-6 X terminal Storage Processor statistics 9-14 address 2-11 uninitialized status 7-6 as system console 2-15 used with a mirrored partition 5-5 name 2-11 used with mirrored partition 5-5 use of /usr/openwin 4-16 X windows Write cache analyzing data 9-48 on-line documentation B-2 xwd command 9-38, 9-39 control data in 7-3 hit percentage 9-14 on state 7-6 Υ purging unwritten data 7-10 states and commands 7-6 ypinit command 2-4 statistics 9-3 Ζ X Zooming in on samples in a X console 2-15 histogram 9-32 to 9-34